6-2023

Novel Approach for Non-Invasive Prediction of Body Shape and Habitus

Emma Young

Novel Approach for Non-Invasive Prediction of Body Shape and Habitus

Abstract

While marker-based motion capture remains the gold standard in measuring human movement, accuracy is influenced by soft-tissue artifacts, particularly for subjects with high body mass index (BMI) where markers are not placed close to the underlying bone. Obesity influences joint loads and motion patterns, and BMI may not be sufficient to capture the distribution of a subject's weight or to differentiate differences between subjects. Subjects in need of a joint replacement are more likely to have mobility issues or pain, which prevents exercise. Obesity also increases the likelihood of needing a total joint replacement. Accurate movement data for subjects with a higher BMI is of the utmost importance because it can be used to inform treatment options for people who have received joint replacements or are waiting to receive a replacement. Currently, movement data from this subgroup of people has error introduced due to soft tissue artifacts from markered motion capture, such as Vicon. By investigating ways of measuring movement with computer vision tools, the influence of soft tissue artifacts can be investigated further in movement data. The Azure Kinect DK is a depth camera that collects point cloud data of the surface of the person and automatically calculates joint centers. The objectives of this thesis were to 1) Design a fast and accurate procedure to generate a full-body point cloud with two Azure Kinect DK cameras; 2) Create subject-specific computational representations by fitting the Skinned Multi-Person Linear (SMPL) model to Kinect point cloud data. With IRB approval, 24 subjects consented to perform T-pose , lunge, sit to stand, and walking activities while being recorded by two Azure Kinect DK depth cameras. For all activities except walking, one camera was facing the subject head on and the other recorded the right sagittal view. During walking, the subjects walked toward both cameras. Of these 24, point cloud and joint data from 16 subjects were used. This study uses exclusively data from the T-pose trial.

Without post-processing, these two views of the T-pose do not produce complete data of the surface of a person. A technique was developed to recreate missing sections of the body and create a partial point cloud. Another method of collecting point cloud data was also explored using two cameras and a turn table. A synchronous capture is taken of the front and side, the person is rotated 180 degrees, and another capture is taken. An optimization pipeline was designed to fit the SMPL model to both types of data. The process created high-quality computational representations of each subject. During data collection, extensive anthropometric measures were recorded of each subject with a tailor's tape and used to verify the accuracy of the model by comparing them to digitally recreated measures on each SMPL model. The validity of the model was quantified with a percent error calculation between subject manual and SMPL measurements. Subject manual and SMPL measurements are highly correlated with an $R^2 > 0.9$ and p-value << 0.1. Across all measurements and subjects, there is an average absolute percent error of 4.71 ± 4.09% The average absolute percent error between any measurements never exceeds 10%. The largest absolute percent error is in the ankle with 9.80 ± 6.33% , and the smallest absolute percent error is in the floor to shoulder measurement with 1.17 ± 0.76%.

This tool to create a subject-specific whole-body computational representation has a broad variety of applications. This model fitting process is ready to be deployed on the point cloud collection processes described and will be useful in creating subject-specific finite element models. The Azure Kinect DK gives the ability to perform markerless motion capture and surface point cloud collection in a clinical setting, such as a doctor's office. These body models could be generated from data like this and allow for patient classification. The SMPL model's ability to describe realistic body shapes could be used to accurately calculate the moment of inertia for many body segments. A standardized tool to measure body shape and habitus can allow for the discovery of correlations between body dimensions and movement patterns.

Document Type
Masters Thesis

Degree Name
M.S.

First Advisor
Peter J. Laz

Second Advisor
Casey A. Myers

Third Advisor
Ronnie Pavlov

Keywords
Body model, Computer vision, Optimization, Point cloud

Subject Categories
Biomechanical Engineering | Biomechanics | Biomedical | Computer Sciences | Engineering | Kinesiology | Other Computer Sciences

Novel Approach for Non-Invasive Prediction of Body Shape and Habitus

A Thesis

Presented to

the Faculty of the Daniel Felix Ritchie School of Engineering and Computer Science

University of Denver

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

by

Emma Young

June 2023

Advisor: Peter J. Laz

Author: Emma Young
Title: Novel Approach for Non-Invasive Prediction of Body Shape and Habitus
Advisor: Peter J. Laz
Degree Date: June 2023

ABSTRACT

While marker-based motion capture remains the gold standard in measuring human movement, accuracy is influenced by soft-tissue artifacts, particularly for subjects with high body mass index (BMI) where markers are not placed close to the underlying bone. Obesity influences joint loads and motion patterns, and BMI may not be sufficient to capture the distribution of a subject's weight or to differentiate differences between subjects. Subjects in need of a joint replacement are more likely to have mobility issues or pain, which prevents exercise. Obesity also increases the likelihood of needing a total joint replacement. Accurate movement data for subjects with a higher BMI is of the utmost importance because it can be used to inform treatment options for people who have received joint replacements or are waiting to receive a replacement. Currently, movement data from this subgroup of people has error introduced due to soft tissue artifacts from markered motion capture, such as Vicon. By investigating ways of measuring movement with computer vision tools, the influence of soft tissue artifacts can be investigated further in movement data. The Azure Kinect DK is a depth camera that collects point cloud data of the surface of the person and automatically calculates joint centers. The objectives of this thesis were to 1) Design a fast and accurate procedure to generate a full-body point cloud with two Azure Kinect DK cameras; 2) Create subject-specific computational representations by fitting the Skinned Multi-Person Linear (SMPL) model to Kinect point cloud data. With IRB approval, 24 subjects consented to perform T-pose , lunge, sit to stand, and walking activities while being recorded by two Azure Kinect DK depth cameras. For all activities except walking, one camera was facing the subject head on and the other recorded the right sagittal view. During walking, the subjects walked toward both cameras. Of these 24, point cloud and joint data from 16 subjects were used. This study uses exclusively data from the T-pose trial.

Without post-processing, these two views of the T-pose do not produce complete data of the surface of a person. A technique was developed to recreate missing sections of the body and create a partial point cloud. Another method of collecting point cloud data was also explored using two cameras and a turn table. A synchronous capture is taken of the front and side, the person is rotated 180 degrees, and another capture is taken. An optimization pipeline was designed to fit the SMPL model to both types of data. The process created high-quality computational representations of each subject. During data collection, extensive anthropometric measures were recorded of each subject with a tailor’s tape and used to verify the accuracy of the model by comparing them to digitally recreated measures on each SMPL model. The validity of the model was quantified with a percent error calculation between subject manual and SMPL measurements. Subject manual and SMPL measurements are highly correlated with an $R^2 > 0.9$ and p-value << 0.1. Across all measurements and subjects, there is an average absolute percent error of 4.71 ± 4.09% The average absolute percent error between any measurements never exceeds 10%. The largest absolute percent error is in the ankle with 9.80 ± 6.33% , and the smallest absolute percent error is in the floor to shoulder measurement with 1.17 ± 0.76%.

This tool to create a subject-specific whole-body computational representation has a broad variety of applications. This model fitting process is ready to be deployed on the point cloud collection processes described and will be useful in creating subject-specific finite element models. The Azure Kinect DK gives the ability to perform markerless motion capture and surface point cloud collection in a clinical setting, such as a doctor's office. These body models could be generated from data like this and allow for patient classification. The SMPL model's ability to describe realistic body shapes could be used to accurately calculate the moment of inertia for many body segments. A standardized tool to measure body shape and habitus can allow for the discovery of correlations between body dimensions and movement patterns.

ACKNOWLEDGEMENTS

Above all else, I would like to thank my advisor Dr. Peter Laz for his support and encouragement over the last three years, and for initially proposing this project in 2020.

Thank you to the other members of my committee: Dr. Casey Myers and Dr. Ronnie Pavlov. Casey has been an invaluable member of this project from the beginning and has been a great help when operating the Kinect cameras.

Thank you to all my friends and family who participated in my data collections. Without you, this would not be possible!

Thank you especially to my mother, who proofread this entire document and constantly gives me strength to continue on.

Finally, thank you Thor for everything!

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 Introduction and Motivation

Human movement is quantified using marker-based motion capture systems, such as Vicon. These systems are considered the gold standard for motion capture, yet it can be time consuming to post-process data. Additionally, a Vicon camera system can cost as much as $75k, require an entire room and skilled technicians to keep it operational, and subjects need to sometimes travel a considerable distance for a data collection. Due to the introduction of soft tissue artifacts from the usage of marker-based motion capture, it is difficult to record the locomotion for patients with higher BMI. Motion capture results have a moderate accuracy of 3-5mm and depend on how close the marker is placed to the underlying bone. These collection methods do not accurately capture the full shape of the human body. This is an important issue in biomechanics, and methods should be developed to define relationships between shape and movement patterns. Investigating this can enable novel research paths involving obesity, and implant design for obese populations. Obesity is reaching epidemic levels worldwide, recent studies have estimated almost 108 million children (5%) and 604 million adults (~12%) worldwide are obese. Obesity and increases the likelihood of needing a total joint replacement surgery [1]. The annual number of total hip replacements of people over the age of 45 doubled between 2000 and 2010 from 138,700 to 310,800. More young people are also needing joint replacements than ever before [2]. As the world becomes more overweight, medicine needs to shift its focus to designing and evaluating implant performance on the growing obese populations and their altered movement patterns [3].

The introduction of soft tissue artifacts to motion capture data is a known problem, and more pathways to collecting movement data should be explored to gain a broader perspective of the issue. Computer vision-based systems trained to identify joint centers in segments of recorded activities are one such example of an alternative system. The Microsoft Azure Kinect DK is a depth camera capable of simultaneously recording point cloud data of a scene and body tracking information of multiple people in a frame. The validity of the data from technology like this is no longer dependent on inaccurate information from markers. In a study to quantify the accuracy of the Azure Kinect DK against markered activity, walking speed and stride length were highly correlated. The average difference in maximum knee flexion angles was 2.84° [4].

The ability to generate a detailed point cloud of the surface of a subject can offer the ability to address the need to investigate body shape beyond what markered work previously allowed. Depth information from the Kinect cameras is accurate, but difficult to interpret on its own. A well described body model is needed to draw conclusions about the relationships between body habitus and movement patterns. The Skinned Multi-Person Linear (SMPL) model, developed by the Max Planck Institute for Intelligent Systems, realistically describes 3D body shapes[5] using geometry created from the Archive of Motion Capture as Surface Shapes (AMASS) [6], which is an open-source database consisting of thousands of body scans. The SMPL model is defined by 10 shape parameters (ß) which are principal component scores describing variance in several dimensions of the body. The body can be articulated with 63 pose parameters (θ), which control movement of the body in 3 dimensions for each of the model's 23 internal joints. There are three SMPL models available: Male, Female, and Neutral.

The introduction of markerless motion capture and depth data may not immediately address the issue of soft tissue artifacts when collecting human movement, but it offers an alternative to the status quo that is fast and can capture surface geometry. This depth data allows for the implementation of a realistic body model, like the SMPL model, with joint centers placed in locations where limbs can be realistically articulated.

1.2 Objectives

The objectives of this thesis were to:

1. Design a fast and accurate procedure to generate a full-body point cloud with two Azure Kinect DK cameras.
2. Create subject-specific computational representations by fitting the Skinned Multi-Person Linear (SMPL) model to Kinect point cloud data.

Previous work involved using 4 synchronized Azure Kinect depth cameras, and two laptops to gather a full-body point cloud of subjects [4]. The result created high quality representations, but data processing was time intensive and difficult. The workflow can be made more efficient if only one laptop and 2 cameras are used. Two methods were explored and implemented: a partial point cloud with reconstructions, and a full point cloud. The motivation behind collecting point cloud surface data of subject is to ultimately quantify body shape and improve understanding of relationships between habitus and movement patterns.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

In the study of biomechanics, the gold standard for measuring human movement is passive marker-based motion capture systems, such as Vicon. When collecting human movement with a Vicon camera system, the subject is fitted with retroreflective markers on segments of the body of interest. The cameras emit pulses of infrared light that reflect off these markers and back to the cameras. The movement of each marker is calculated and used to describe the subject’s movement.

When the marker is not placed close to the underlying bone or the subject has a higher body mass index (BMI), the accuracy of these systems is known to be influenced by soft-tissue artifacts. Obesity influences joint loads and motion patterns [7], [8], and BMI alone is not sufficient to capture the distribution of a person’s weight. Exploring markerless alternatives to quantify locomotion and body shape has clinical relevance, and can offer a new, streamlined way of conducting human movement trials.

This literature review will first explore methods of motion capture, with an introduction to theory and historical uses, and a heavy focus on the Azure Kinect DK and its clinical uses. The depth data the Kinect generates is also of importance: the second part of this review will delve into the world of point cloud optimization.

2.2 Markerless Motion Capture

The Azure Kinect DK is a recent development in markerless motion capture and point cloud technology. It first appeared on US markets in March 2020, and improves on previous generations of Kinect cameras. Work on this family of cameras was originally developed as a motion controller for the Xbox 360 and released in 2010. The Azure Kinect, like its predecessors, uses the Amplitude Modulated Continuous Wave (AMCW) Time-of-Flight (ToF) principle to generate depth images [9]. This principle measures distance by calculating the time a light pulse needs to travel from the camera, to an object in the scene, and back again [10]. If using more than one ToF camera, this system can be prone to error if the frequencies of the modulating light sources are not offset appropriately. Up to ten Azure Kinects can be used simultaneously if they are connected together with external sync cables. Interference between multiple cameras is avoided automatically through this syncing process if the appropriate firmware is installed, as the generated synching signal offsets the camera captures by 160µs.

The built in computer vision framework inside the Azure Kinect is capable of tracking multiple people simultaneously. Other open-source methods have also been developed to perform 3D segmentation of human point clouds with methods similar to the Kinect. These processes include 3D semantic segmentation (differentiating a person from background noise), 3D instance segmentation (the potential to identify more than one person in a frame), and 3D multi-human body-part segmentation (dividing human instances into their body parts) [11].

The Kinect camera family's ability to collect accurate [12] depth data and markerless motion capture makes it an ideal tool for quantifying human kinematics and body shape. This noninvasive, full-body, automated method of data collection allows for novel analysis of different aspects of walking, and how they are related to joint angles and kinematics of isolated body parts. This has proved particularly useful in analyzing the gait and arm swing of Parkinson's Disease (PD) patients [13]. The use of the Kinect in this study allowed researchers to analyze arm swing in healthy and PD subjects relative to the trajectory of their center of mass. Uneven and atypical arm swing patterns are an early warning sign of the development of PD. The Kinect's ability to easily capture and describe full-body movement simultaneously is superior to the data generated from wearable sensors, where movement described is limited only to parts of the body where markers are placed. In this study, a single camera was used. The portability of this motion capture system also allows human motion to be measured in scenarios outside a laboratory, like for instance in a home or a doctor's office. The Kinect camera has been proposed as a remote method for monitoring Parkinson's Disease [14]. This procedure was able to quantify statistically significant features of motor performance and easily monitor disease progression. Disease progression is typically monitored through regular doctor visits, but there is a need for more frequent assessment of patient health.

The Azure Kinect is not the only depth camera available on the market: Intel manufactures a family of cameras comparable in quality. The stereo depth family includes 5 cameras with various uses, to gather point cloud data indoors and outdoors. These cameras have been used in numerous studies as a means of collecting information about an experimental scene. The Intel RealSense technology has applications in clinical research and the measurement of health outcomes [15]. Specifically in rehabilitation, these camera systems allow for a low-cost and noninvasive way of evaluating the effectiveness of treatments. Depth cameras like the Intel RealSense are used most often to perform a 3D scan of an experimental environment. In addition to this, RealSense cameras have been used to track facial features and place facial key point locations, which can enable methods of automatically recognizing emotion. They have been used to create an automatic emotion-recognition framework to read the emotions of customers and gauge satisfaction [16].

With advancements in computer vision, traditional motion capture can be replicated with two-dimensional video-based methods to analyze human motion. A few open-source models exist, such as OpenPose [17], Tensorflow MoveNet Lightning, Tensorflow MoveNet Thunder [18], and DeepLabCut [19]. Each model is pre-trained on millions of images of individuals performing daily activities. The results of their key point predictions will vary because each is trained on a unique dataset. These models were compared in a clinical setting with an existing dataset of healthy individuals walking with synchronous motion capture. Key point locations from each were used to calculate hip and knee angles. An absolute error was recorded between the angles calculated from the Vicon synchronous motion capture and computer vision method. OpenPose and MoveNet Thunder produced the lowest error over the gait cycle, while MoveNet Lighting and DeepLabCut produced the largest error [20].

The accuracy of OpenPose has been further validated with a similar experimental setup [21]. In comparing three-dimensional motion capture from walking healthy adults to kinematics predicted with OpenPose, it was found that OpenPose was able to perform with comparable accuracy to motion capture. Mean absolute errors between motion capture and OpenPose were 0.02s for temporal gait parameters and 0.049m for step lengths.

2.3 Point Cloud Optimization Methods

The Azure Kinect's generation of point cloud data creates the need for intelligent post-processing procedures. This includes the down sampling of data, and a way of organizing and using data. Point cloud optimization is a broad topic useful in many areas of study.

Mathematical optimization is a field of study to select the best design criterion in a system. This could be as concrete as selecting the best dimensions to minimize the weight of a shipping container, or as abstract as calculating the optimal path for a robot to take to perform a task. Optimization problems can be convex or non-convex, and constrained or unconstrained. Convex problems and methods for solving their constrained versions are well understood [22] with the application and refinement of steepest descent algorithms. Methods for solving non-convex problems are more diverse and the method used needs to be chosen based on the type of problem being solved.

A point cloud is a collection of points that represent objects in 3D space using sets of Cartesian coordinates. Depth cameras, 3D scanners, and Lidar are all examples of technology that can create a point cloud. This data has applications, most notably in robotics and computer vision, where information about a scene must be efficiently inferred from a discretized representation. In other fields, point cloud registration has become a valuable tool in digital documentation and conservation of cultural heritage sites. Special point cloud optimization processes have been created in the field of architectural heritage modeling to identify different dimensional features of buildings and other manmade structures and reduce cloud density in regions of low geometric complexity [23]. This research is concerned with reducing point cloud complexity, while other studies of point cloud optimizations are concerned with recreating missing or noisy data. For example, 3D trees can be reconstructed from a single laser scan through carefully tailored point cloud optimization. After a coarse tree skeleton is extracted from a terrestrial laser scan, an iterative optimization is employed to compute the dominant directions and densities of points and estimate new skeleton points. This is repeated until all regions of missing data are filled [24].

There are certain optimization algorithms that are classically suited for point cloud data, such as the uses for Iterative Closest Point (ICP) and Coherent Point Drift (CPD) in point cloud registration and deformation. With a good initial guess, an ICP algorithm can iteratively guess translation and rotation to rigidly move a source cloud to a location near its correspondence in the target cloud [25]. ICP is improved with the use of point-to-plane and plane-to-plane metrics [26], which use surface normal to more accurately guess correspondence. Research has been conducted to uncover methods of registering point clouds in sparse-dense scenarios, where there is no direct correspondence between every point in the two clouds. The denser point cloud environment is divided into cells, and the points contained are used to create one representative point of the local surface [27].

In CPD, correspondences between two datasets are assigned point by point, and a transformation is defined to map each correspondence [28]. The most used CPD algorithms are based on motion coherence theory popularized in the late 80's [29]. CPD can be easily scaled to large point sets because point-to-point correspondences are not assumed to be one-to-one [30].

These algorithms together are used commonly to perform correspondence registration for statistical shape and intensity models. After meshing all subject bone geometries, ICP is used for rigid-body alignment, and CPD is used to deform the template mesh to each subject mesh. To account for irregular geometries, an additional non-rigid ICP algorithm is performed [31], [32].

In other methods, radial basis functions are commonly used to morph meshes and establish correspondence. A radial basis function can be thought of as a single layer neural network with numerous inputs, but one output. The value depends on the distance between the inputs and a fixed point. Mesh morphing with radial basis functions has the benefit that it is mesh independent, thus does not rely on the density of the fixed or moving cloud. The radial basis function morphs meshes by creating a deformation field from points of interest. This has applications in fields such as finite element analysis (FEA) or computational fluid dynamics (CFD), where structured meshes must be carefully adapted to different scenarios. With a radial basis network, correspondence between a target and source mesh can be quickly established. It is common for a template mesh to be morphed to a new, similar shape. This is less computationally expensive compared to the alternative of generating a new complex mesh for each new experimental scenario.

These methods have been similarly applied in the automotive design industry. Variables governing car structure design are traditionally simpler because of the computational expense of re-meshing CAD components and their connections. Radial basis function mesh morphing can be used to easily change mesh shape [33] in response to an optimization problem. The robustness of morphed meshes are verified by comparison to a baseline method of optimizing design variables to reduce intrusion into the passenger compartment of a car.

By using radial basis functions to aid in mesh transformation, the peak acceleration and maximum intrusion are reduced. In cardiovascular research, radial basis functions are used to morph meshes in scenarios where the mesh boundary must change. A moving boundary problem is common when modeling heart valves, as the motion of the fluid and heart muscle must be simultaneously monitored. Radial basis functions can restructure nodes without remeshing geometry, which leads to a computationally efficient solution [34].

2.4 Obesity and Kinematics

There are significant and well documented correlations between obesity and changes in kinematics. Obesity, most specifically a larger waist circumference, is correlated with a higher likelihood of spinal fatigue compression fracture compared to people with a smaller waist circumference [35]. This study was conducted by performing a principal component analysis on the calf, thigh, waist, and upper arm circumference using a database from the National Health and Nutrition Examination Survey (NHANES) of 5852 obese individuals. Obesity shape distributions were defined as apple and pear shaped and applied to body models with parameterized shape modifications. A regression was created to correlate segmental masses at each spinal section. Spinal load was estimated with a validated musculoskeletal model.

Obesity not only alters whole body kinematics and joint loading, but it also increases the risk of musculoskeletal injury and the potential to develop osteoarthritis (OA) in the lower limbs or disability. Excess adipose tissue drives biomechanical issues, but the whole-body effect of obesity also carries great influence. Obesity is a disease that affects musculoskeletal systems and organs. Even after an obese individual loses weight, they are still at risk of developing disability and lower extremity OA. This is because obesity triggers musculoskeletal adaptations that alter muscle power, structure, and recruitment pattern [36]. Further proof exists linking obesity to changes in biomechanical behavior. In a study to determine the gait differences between men and women and individuals with severe knee OA it was found that sex and obesity may affect biomechanics [37]. Knee and ankle kinematics were investigated in obese young individuals and compared to that of an age matched healthy cohort. There were large differences in knee and ankle function, more specifically less knee flexion, more knee ab-adduction during a gait cycle, and reduced range of motion at the ankle joint. The obese cohort exhibited a knee flexion-extension at initial contact of 3.7 ± 4.0°, a max knee ab-adduction in swing of 28.0 ± 12.2, and an ankle dorsi-plantarflexion at IC of 0.2 ± 3.2°. In the control cohort these numbers were 4.4 ± 3.2°, 10.8 ± 7.5°, and -1.4 ± 2.4°[3].

Obesity is a disease greatly affecting the adult population, but the rate of childhood obesity has also been steadily increasing in recent years. Studies have been performed to assess the impact this has on the adolescent population, but also how to most effectively halt the progression of the disease. A 13-week exercise program was designed for overweight and obese children and results from the study were compared to a control group. It was found that exercise halted the progression of gait alterations in these individuals. The problematic gait patterns of the control group continued to progress, while the exercise groups' patterns did not advance further, but they also did not improve [38].

2.5 Patient Classification

There are a variety of ways and reasons to classify an individual. In medicine, metrics like body mass index (BMI), waist circumference (WC), and a body shape index (ABSI) are used to try to predict the occurrence of chronic health conditions related to weight. ABSI is a number calculated by dividing waist circumference by an estimated regression calculated from weight and height. It is statistically independent of BMI. A generalized linear model evaluated the association of ABSI and BMI with various risk factors. Use of ABSI and BMI together in this model improved predictions of cardio-metabolic risk factors [39]. In other work, ABSI was compared to WC and BMI with meta-analysis of literature. It was found to increase the prediction of hypertension, type 2 diabetes, cardiovascular disease, and mortality risk. ABSI was better at predicting mortality but worse at predicting chronic disease [40].

The desire to classify body shape is also incredibly prevalent in the fashion industry. The objective of clothing manufacturers is to create attire that will appropriately fit people. There are ongoing projects to design dress form sizes that can accurately describe the body shape variation of the population. A Korean study designed to address this issue examined 63 anthropometric measurements from middle aged women to more efficiently classify body types. Four body types were identified and nine measurements most key in making these classifications were extracted using discriminant analysis [41]. The female body shape has been classified in similar ways in the fashion industry to try to solve a different problem: fabric utilization. A large cost of manufacturing clothing comes from the cost of fabric. A study was conducted to classify female body shapes and then correlate them with fabric usage in fitted trousers and a blouse. Three-dimensional body scans were recorded of young women and processed using principal component analysis and a K-Means ++ clustering algorithm. The clusters found body shapes of oval, circle, triangle, and rectangular. Fabric utilization was most efficient in a rectangular body type and least efficient for a circle body type [42].

Patient classification is also useful in orthopaedic medicine to predict the severity of arthritis. A famous way of classifying patients is the Walch classification. The Walch classification scale was created to distinguish different types of pathological glenoid differences in shoulder geometry and the progression of glenohumeral osteoarthritis. Classification into three main subcategories of shape and disease severity can assist in preoperative planning and potential operative difficulties [43]. A new glenoid shape was proposed, Walch B0, and identified by studying humeral head subluxation. This is an additional way to identify patients at risk of developing glenohumeral osteoarthritis and create treatment plans to slow or halt the progression of the disease. The Walch B0 glenoid appears to lead to more posterior wear of the glenoid, putting patients possibly at an increased risk of retroversion [44]

CHAPTER 3: FULL-BODY POINT CLOUD WITH TWO AZURE KINECT DK CAMERAS

3.1 Introduction

This work is a continuation of a previous project to investigate the accuracy of the Kinect's body tracking capabilities and document a process to generate full-body depth data of subjects. The accuracy of the Azure Kinect has been shown to be comparable to marker-based motion capture in dynamic activity. Walking speed and stride length between the two are highly correlated, and a minimal difference in maximum knee flexion angle has also been observed [4]. During this previous experiment, 4 cameras were used to capture a view from the front, back, and left and right sides. Using four cameras required 2 laptops set up with a remote desktop connection so one could trigger all four cameras at once. This setup worked well, but required extra, potentially unnecessary, post-processing to align all the captures into one coordinate system.

These methods explore the collection of a full-body scan using two cameras and one computer to streamline this data collection process. The point cloud and joint center data the Kinects generate are valuable, and a quicker method of obtaining them is needed so this approach can have a broader application. The objective of the work described in this chapter is to streamline a previous method of capturing human movement with the Azure Kinect DK. This enables the capture and quantification of human body shape and could be used in the future for patient classification.

3.2 Methods

Two data collection procedures were devised to create a full-body scan with two Kinect cameras. These methods do not capture dynamic movement well and are designed to capture a static T-pose. With approval of the University of Denver Institutional Review Board under Protocol: 1567600-2, 24 healthy subjects consented to perform a variety of activities in front of two Azure Kinect DK cameras including a T-pose, lunge, sit to stand, and walking. The T-pose data from 16 of these individuals is used exclusively in this work with average weight of 75.25 ± 18.89kg, height of 172.02 ± 10.51cm, and BMI of 25.14 ± 4.30 $\frac{kg}{m^2}$.

The first method of collecting a full-body scan takes two synchronous depth captures: facing the subject and facing the right sagittal plane of the subject. The two captures are stitched together into the same coordinate system using a transformation matrix created from a calibration capture. This data is missing large parts of the subject, so extensive post-processing is needed to generate data resembling a full scan. After processing, the missing left side of the person can be recreated, and the back is estimated.

The second method creates a true full-body scan with two cameras and two synchronous captures. This procedure uses a turntable with a custom calibration object on it that the subject can stand on. One synchronous capture is taken with the subject facing the master camera, then the experimenter rotates them to take another synchronous capture of the back side of the subject. During the first synchronous capture, the subordinate camera records the left sagittal plane of the subject. When they are rotated, the right sagittal plane is recorded.

3.2.1 Partial Point Cloud

The partial point cloud collection begins with two depth camera views stitched together into one coordinate system. Missing parts of the point cloud are recreated or estimated. The Kinect cameras were arranged such that their views were 90 degrees offset from each other. The subjects were asked to perform a T-pose with palms down facing the master Kinect camera, the subordinate camera recorded their right sagittal plane (Figure 3.1). The two cameras were set up to run synchronously with the use of an auxiliary cable, as described in Microsoft’s documentation. A 5-second capture was recorded from both views. The point cloud from the subordinate camera was transformed into the master camera’s coordinate system with the use of calibration objects: two floor tiles (Figure 3.2). The top corners from each view were identified and used to find the optimal translation and rotation matrix to align the current camera configuration. This was repeated for each new data collection to ensure the correct matrices were found. Figure 3.3 shows a comparison between the view from the master camera and the master and subordinate point clouds after being aligned. To preserve as much information as possible about the subject’s anatomy, the left side of the body is mirrored from the right and the back is estimated. The positioning of the subordinate camera shows only the right side of the subject (Figure 3.4), and one could reasonably assume the left side looks like the mirror image of this data. To achieve this, the master and subordinate point cloud were mean centered using the mean coordinates of the master point cloud. After centering using this view, the point cloud data from the subordinate camera were mirrored across the centerline of the body, recreating the unseen left side of the body. The trunk of the point cloud was extracted using points found between the hip joint and the armpit. Both are coordinates that were either found or calculated from the Kinect joint centers. Once the torso was isolated, it was divided into numerous slices, projected into the X-Z plane, and the back was estimated by connecting the most posterior points on the right point cloud and recreated left point cloud (Figure 3.5). The result of post-processing can be seen in Figure 3.6.

3.2.2 Full Point Cloud

A method of collecting a full-body scan was created using a large turntable and a new calibration object. Four subjects, S09(M), S11(F), S22(M), and S23(F) were recollected 10 months after their initial partial point cloud collection to generate a full-body scan that could be used in the model fitting process. The calibration object was created by cutting the base of four foam spikes at an angle and hot gluing them to the corners of an aluminum plate angled away from the center of the plate (Figure 3.7). The object was placed on a turntable and the subjects stood with feet wider than shoulder width apart on the plate. It was necessary to create a calibration object that could rotate with the subject on the turntable.

First the subject was positioned in the same way on the turntable as for the data collection for the partial point cloud (Figure 3.8). As the subject faced the master camera, the subordinate camera viewed them from the side and a synchronous capture was taken (Figure 3.9). This will be referred to as the “original” view. Then, the subject remained stationary as the experimenter rotated them 180 degrees on the turn table until they faced directly away from the master camera and the subordinate camera viewed them from the opposite side. Another synchronous capture was taken in this position (Figure 3.10). This will be referred to as the “rotated” view.

The new calibration object was designed to work similarly to the calibration object used in the partial point cloud collections where corners of a common object in the master and subordinate view were used to align the point clouds into one coordinate system. With this new calibration object, three transformation matrices were found. The first transformation matrix was used to align the original subordinate view into the original master view, creating the original point cloud. The second transformation matrix was used to align the rotated subordinate view into the rotated master view, creating the rotated point cloud. The final transformation matrix aligns the rotated point cloud into the original point cloud, creating the final complete view of the subject.

3.3 Results

The full point cloud collection captures the back of subjects, while the partial collection can only estimate this segment. An estimated 80% of the point cloud for the back is missing from the partial data collection. This was calculated by cropping out sections from the back view of the full point cloud collection that are missing in the partial point cloud collection (Figure 3.11).

In Figures 3.12, 3.13, 3.14, and 3.15 the partial point cloud collection can be directly compared to the full point cloud collection. On the left, there is the partial collection with all parts of the point cloud highlighted (front, left, right, and back section). There are clear pieces of the point cloud that are missing: the back of the legs and arms, and the upper back. On the right, the full collection is shown with the original view (front and left side) highlighted in orange, and the rotated in yellow (back and right side). There are no missing sections of the body, and this created a high quality, full-body scan of each subject. Some joint centers can be seen outside the point cloud. This is visible because the location of the joints is averaged over the entire set of frames for the capture, yet the point cloud is from only one frame. The Kinect sometimes performs poorly predicting the joint locations in the extremities, but these miscalculations have no impact on the model fitting process because the joints are rarely used, and the model fitting relies only on the point cloud.

3.4 Discussion

Though the methods described in section 3.2.1. are capable of recreating most of a person's anatomy, there are still large sections of the body that cannot be estimated or recreated. Most notably: the back of the legs, arms, and the upper back. The previous method of collecting depth data with four cameras is a desirable setup if the subject is engaged in a dynamic activity, however the subject can receive a full-body scan with two cameras if they can be rotated around. This is the idea behind the full point cloud data collection where the subject is rotated 180 degrees on a turntable. The turntable method is only appropriate if the subject is performing a stationary activity, like a T-pose. In terms of collection and processing time, as well as ease of implementation, the partial point cloud collection method is superior. Two synchronous captures are taken, and one transformation matrix is found to align the subordinate and master camera views. The other processing to add the left side and back is done automatically with prewritten code. Much of the work processing the data from the full point cloud would be difficult and time consuming for a layperson to carry out. Not only does the alignment require more transformation matrices, but it is also more complicated to create them than the partial point cloud.

In general, both methods create high quality scans of people with little effort. Using two Kinect cameras and one laptop significantly reduces the data collection setup time because a remote desktop connection does not need to be created to control all four cameras from one computer. Post-processing is relatively straightforward with the partial point cloud collection, because only one transformation matrix needs to be found to move the subordinate data into the master coordinate system. Code has already been generated to automatically recreate the left side of the subject and estimate the back. To process the data for the full point cloud collection, three transformation matrices need to be calculated. This is comparable to the old method of collecting data with four cameras, but the setup time for this new procedure is still quicker. The full point cloud collection could be made more efficient if a motorized device could spin the person and calibration object around during the 5 second capture using just one camera. This would reduce the likelihood of the subject moving during the capture and reduce the number of cameras used. To process this type of capture one would need to extract frames strategically of different angles of the body and align them in a way like the full point cloud collection that was deployed. The calibration object of the turn table method could also be improved for easier data processing. Foam spikes were used because they were the most readily available, but ideally a structure with thin shapes like the partial point cloud calibration object should be used. The foam spikes are difficult to crop out of the capture. If the capture is quick enough, the subject could be holding a triad of reflective Vicon markers or wearing a hat with an interesting and easily identifiable object on it. This method should only be used if it is certain the subject will not move considerably during the collection. Proceeding with the partial point cloud collection method is recommended because it is the most straightforward to collect and process.

It is possible some accuracy is lost with the partial point cloud data collection method. One must assume the subject is symmetric when recreating the unseen left side of the subject, so this method cannot be directly applied to asymmetric individuals (e.g.: Amputees or those with significant differences between left and right limbs). The quality of the estimations and point cloud recreations is also dependent on the positioning of the subject. If they are not facing the master camera straight-on the method of mirroring the subordinate camera's point cloud does not produce good results. In this situation, the subordinate camera does not capture a true sagittal view of the subject. Some errors were also introduced due to mistakes collecting data. Some subjects are standing with their legs too close to each other, and their point clouds do not show a clear distinction between the two legs. A "mermaid" tail is created that does not perform well in the optimization to create a subject-specific SMPL model. Additionally, a few subjects did not wear appropriate clothing for the data collection. For example, slightly reflective clothing or clothing that was not form fitting enough. Some data collections could not be used because of this issue.

The full point cloud data collection was performed months after the partial point cloud data collection, and results are greatly improved. The faults of the partial data collection were considered, and all subjects are standing with their legs an appropriate distance apart and are wearing clothing provided by the HDL. The method produces a complete scan of the person, but there are still some areas of the point cloud where there could be of issue. The subject is unlikely to move their legs or torso when they are being rotated, but it is difficult to keep their arms motionless. The changing position of the arms between the original and rotated capture is not of concern because only one subject holds their arms differently in each capture, and the shape and size of the arm is largely preserved.

3.5 Tables and Figures

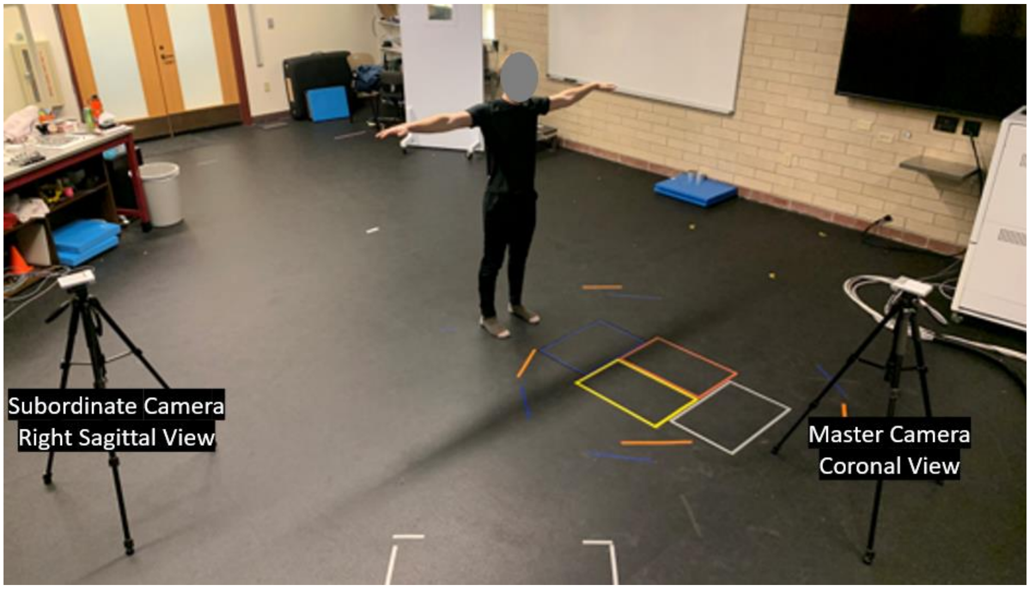


Figure 3.1 Aerial view of the partial point cloud data collection procedure.

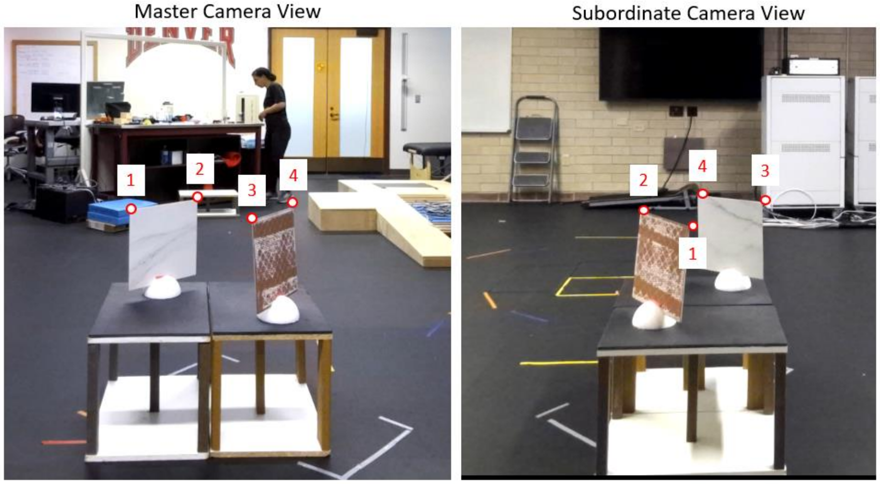


Figure 3.2 The master and subordinate view of the calibration objects used to align the two camera views.

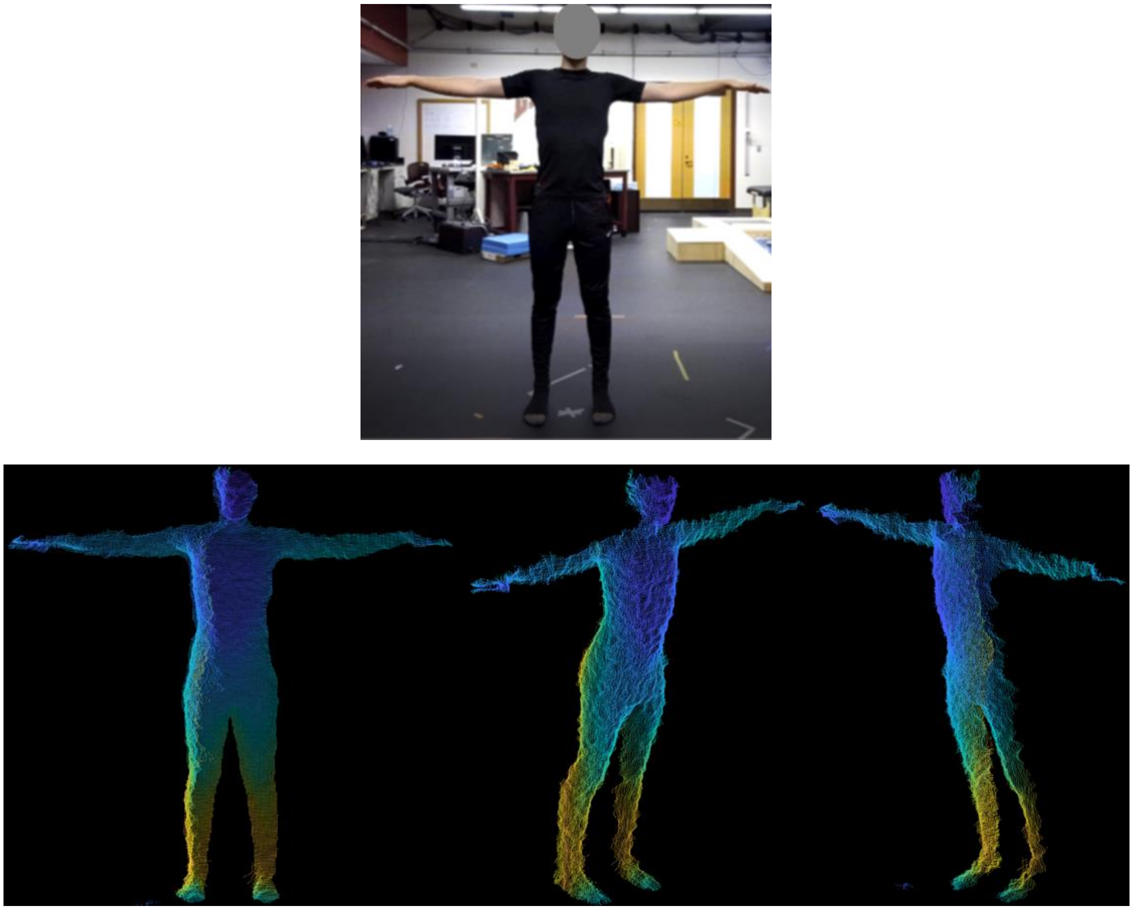


Figure 3.3 The view from the master camera of S01 and the depth image created from aligning the master and subordinate depth data.

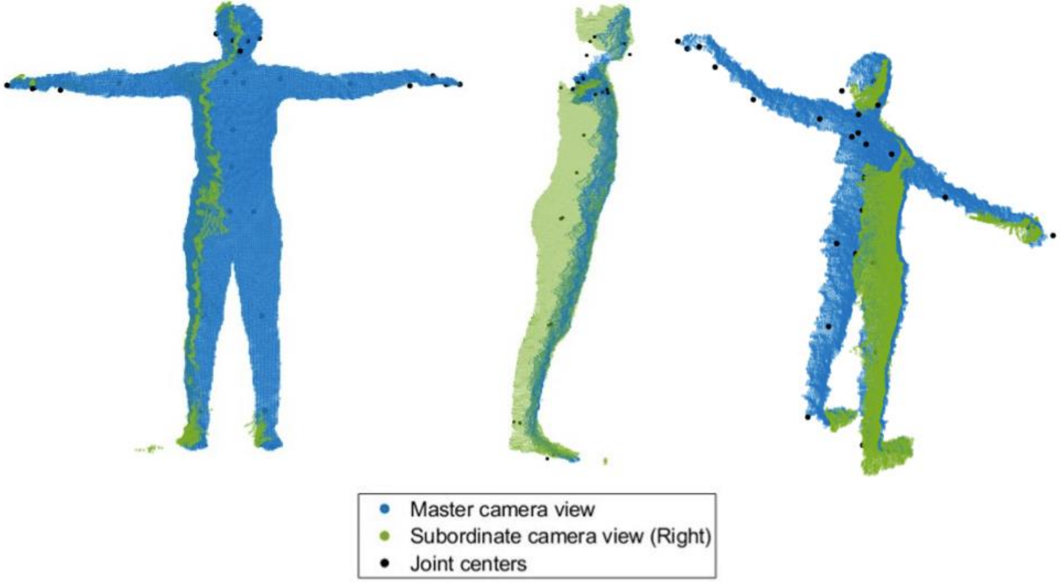


Figure 3.4 The master and subordinate camera views of S01

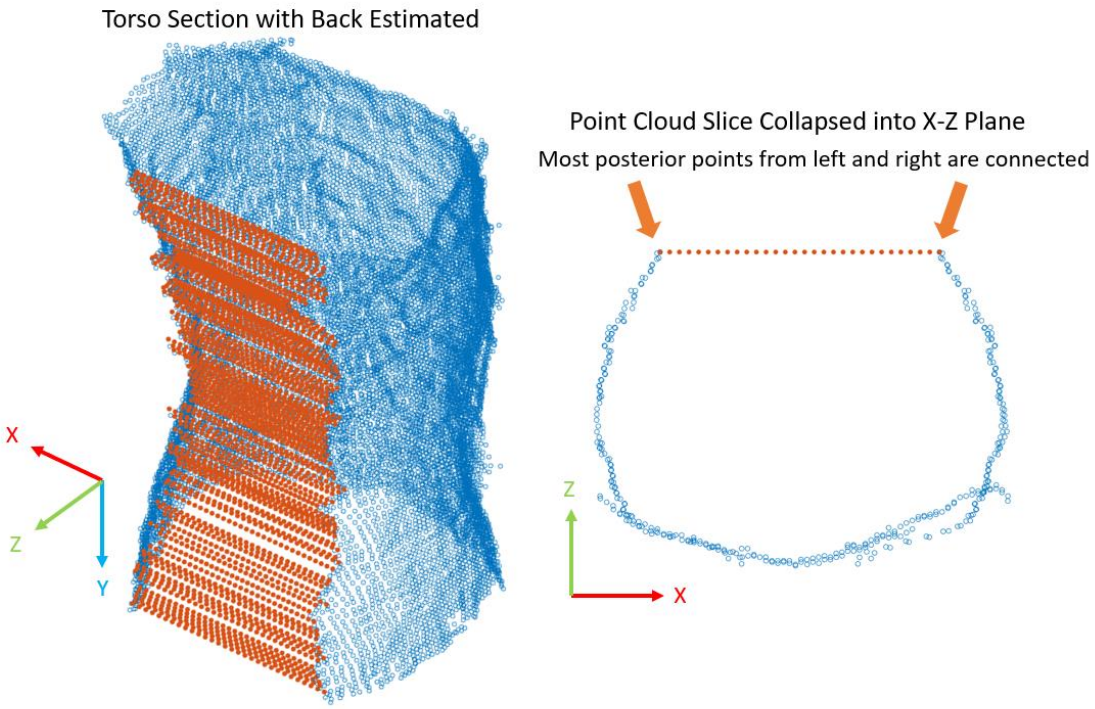

Torso Section with Back Estimated
Point Cloud Slice Collapsed into X-Z Plane
Most posterior points from left and right are connected
X
Z
Y
Z
X

Figure 3.5 (Left) The torso of S01 is shown in blue after recreating the missing left point cloud and the recreated back section is shown in orange. (Right) A slice of the back in the X-Z plane showing the method used to recreate the back.

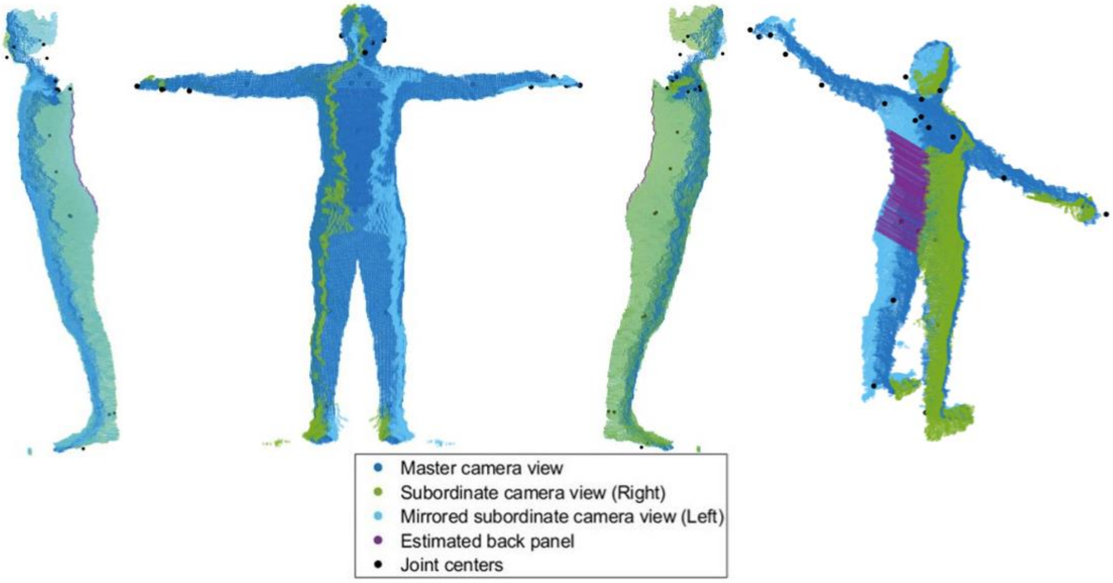

● Master camera view
● Subordinate camera view (Right)
● Mirrored subordinate camera view (Left)
● Estimated back panel
● Joint centers

Figure 3.6 The partial point cloud of S01 after post-processing.

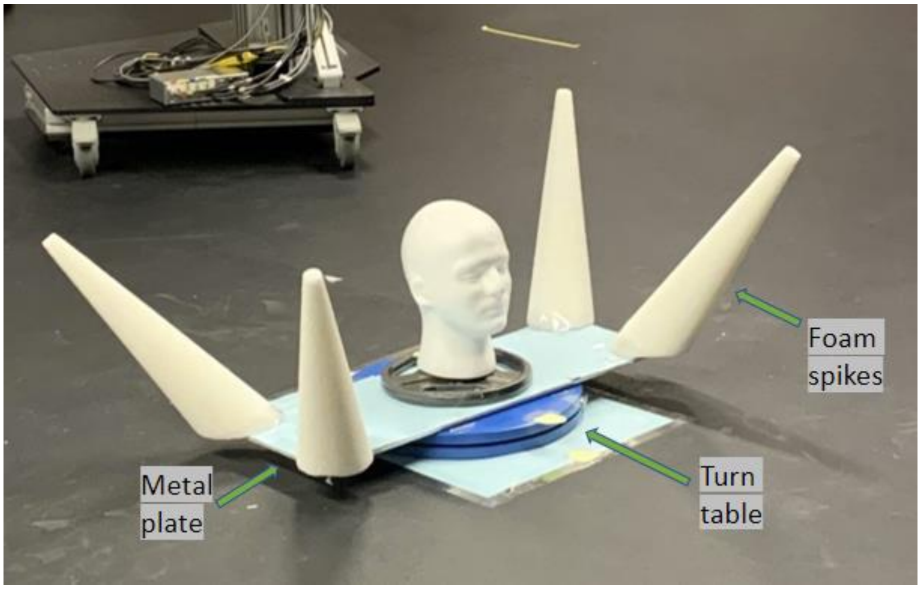


Figure 3.7 The calibration object used to align the depth captures from the original and rotated view to create a full point cloud capture.

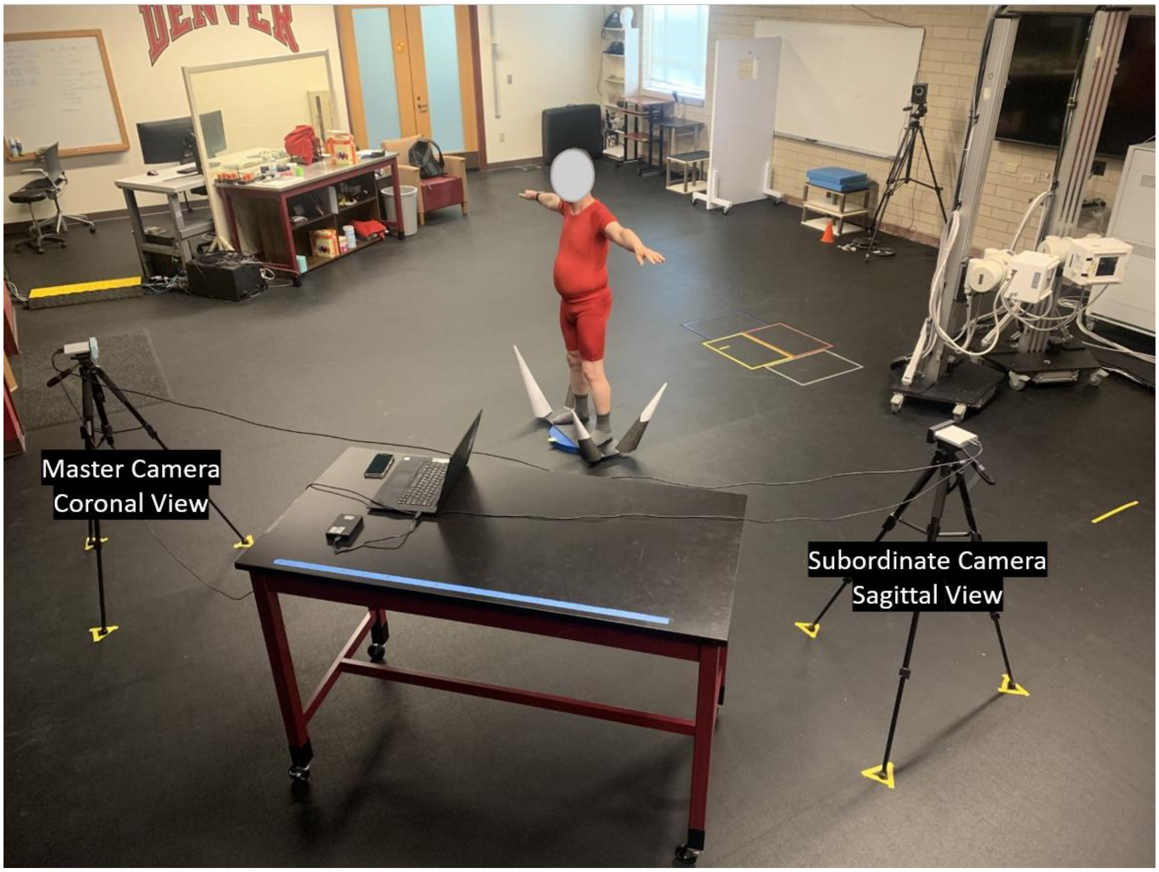


Figure 3.8 Aerial view of the full point cloud data collection procedure.

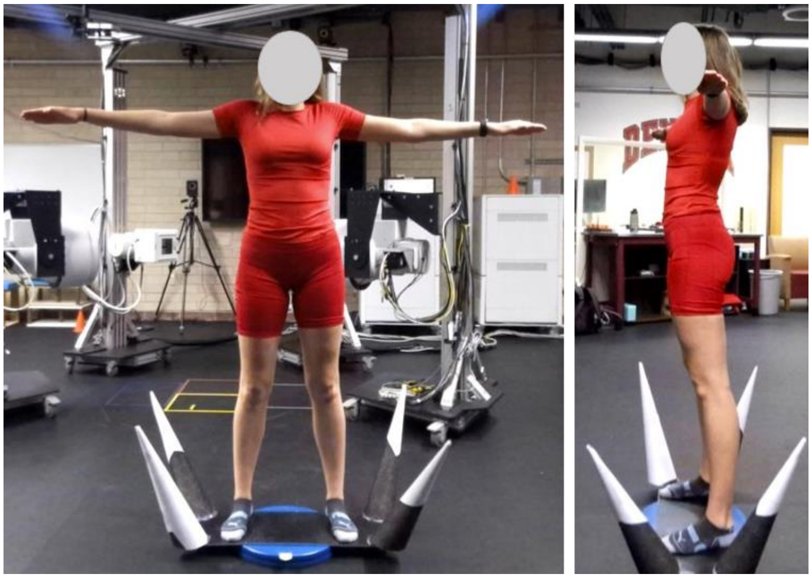


Figure 3.9 A subject standing in the original position for the full point cloud collection: facing the master camera with the subordinate camera viewing the left side.

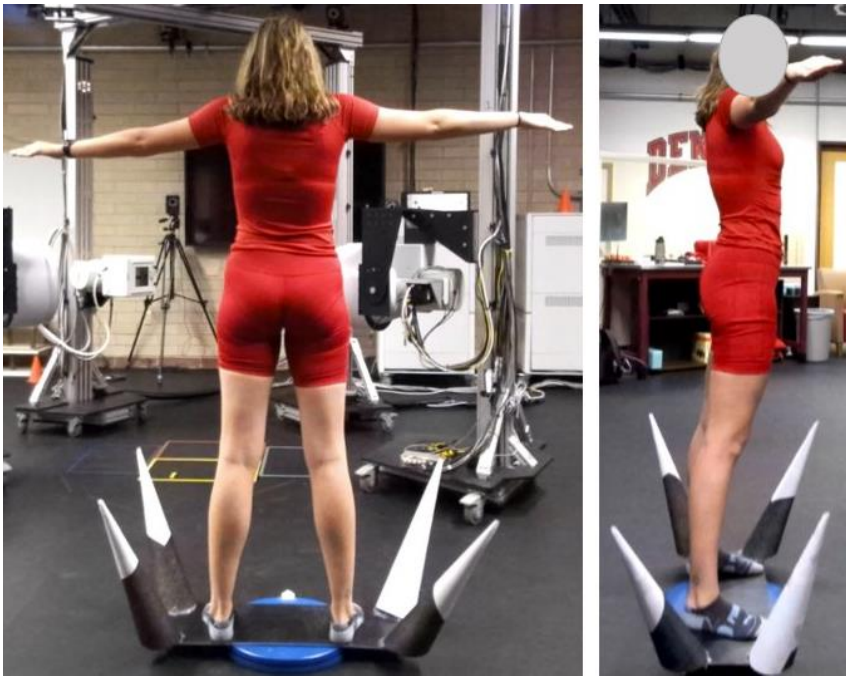


Figure 3.10 A subject standing in the rotated position for the full point cloud collection: facing away from the master camera with the subordinate camera viewing the right side.

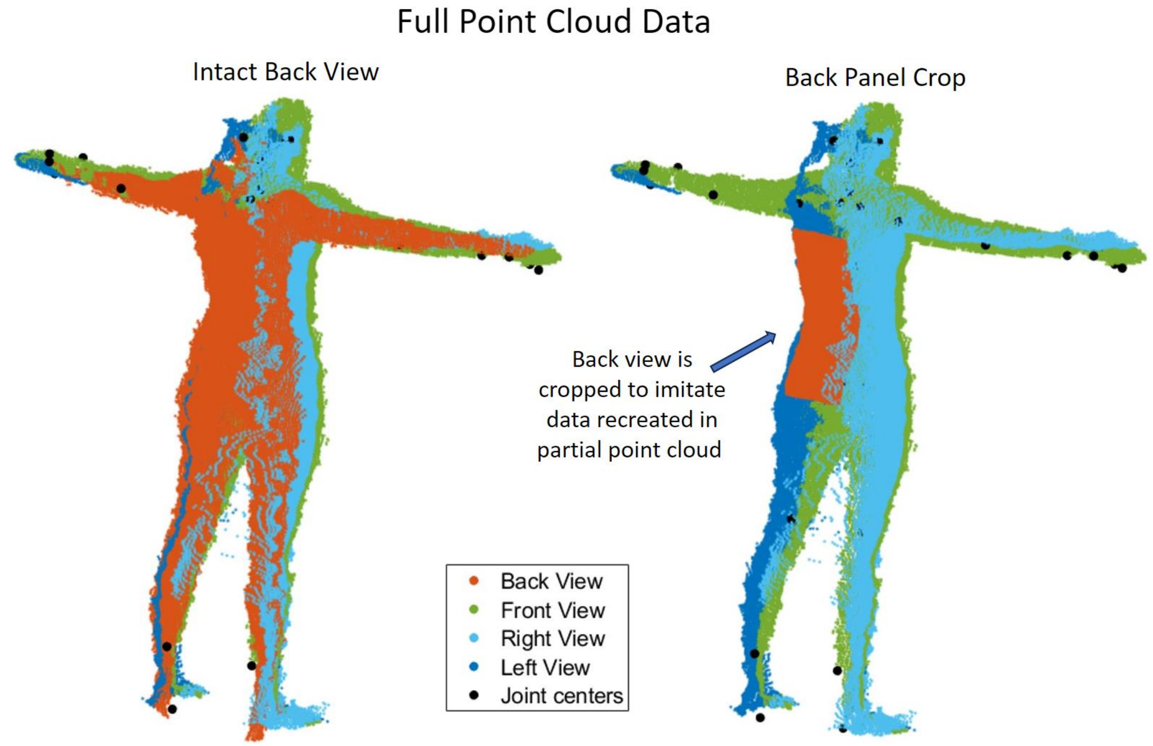


Figure 3.11 The method used to calculate the percentage of the back missing in the partial point cloud collection

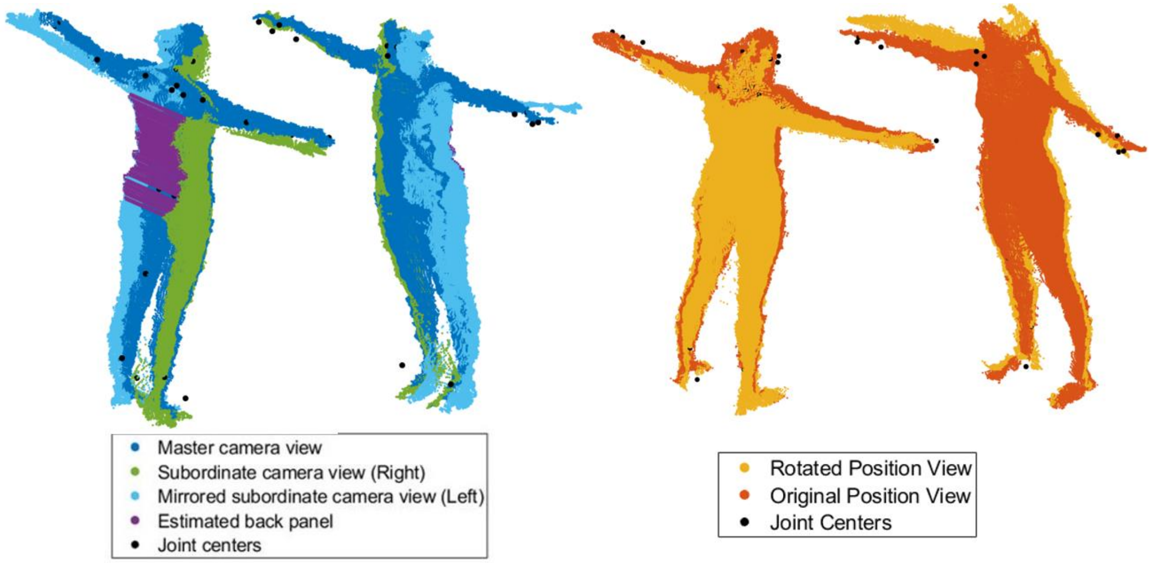


Figure 3.12 A comparison between the partial and full point cloud collection for S23

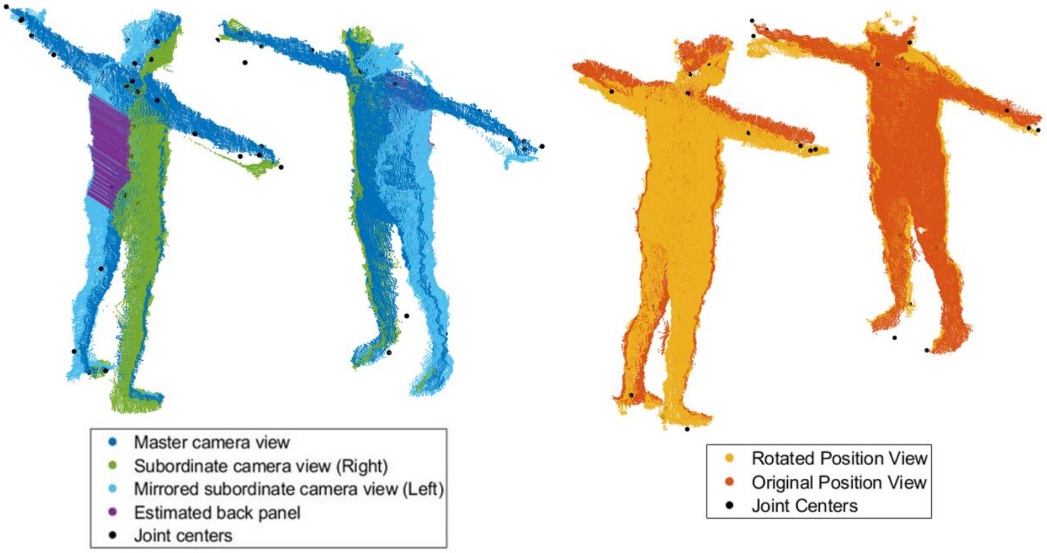


Figure 3.13 A comparison between the partial and full point cloud collection for S22

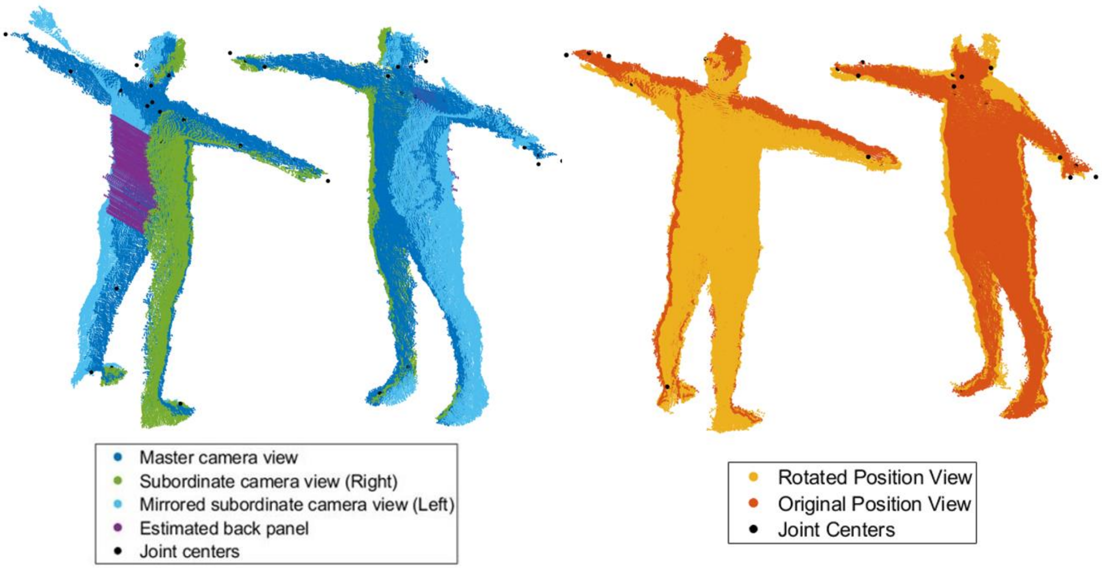


Figure 3.14 A comparison between the partial and full point cloud collection for S09

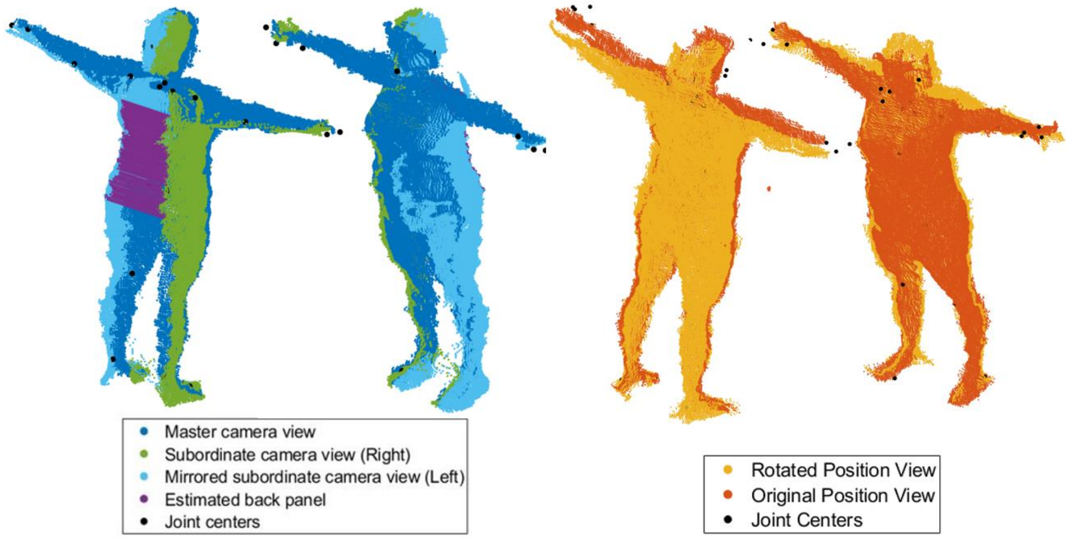


Figure 3.15 A comparison between the partial and full point cloud collection for S11

CHAPTER 4: CREATING SUBJECT-SPECIFIC COMPUTATIONAL REPRESENTATIONS WITH THE SKINNED MULTI-PERSON LINEAR MODEL

4.1 Introduction

The current gold standard for capturing human movement is passive marker-based motion capture systems, such as Vicon. These systems measure human movement with reflective markers placed on the surface of the skin. The data from this is accurate only if the marker is placed closer to the underlying bone, otherwise soft tissue artifacts are introduced. Overweight populations have been underserved in medicine [45], [46] , and this is no exception. Movement data from obese subjects is inaccurate from the shortcomings of markered motion capture, and studies show that obesity influences joint loading and movement patterns [7], [8].

There could be long lasting positive impacts for implant design and patient outcomes if movement studies were to pivot to a method of collecting data that is unbiased to the size of a person. This section describes a process to fit the SMPL model to Kinect point cloud data. Point cloud data of the surface of a person is an unbiased way of capturing movement and body shape.

The SMPL model [5] can depict realistic body shapes and movement. The model is defined by 10 shape parameters (β) which are principal component scores describing variance in certain dimensions of the body. The body is articulated with 63 pose parameters (θ), which can move the model in three dimensions at each of the 23 internal joints. There are three models available: Male, Female, and Neutral. The correlation between shape parameter value and magnitude of mesh movement was also investigated in this chapter.

4.2 Methods

The methods to create a subject-specific SMPL model are broken into four sections: rigid point cloud alignment, mesh morphing, initial SMPL model estimation, and an optimization step for final adjustments (Figure 4.1 and 4.2).

To correlate shape parameter value and magnitude of nodal movement, a Monte Carlo simulation was conducted. The neutral version of the SMPL body model [5] was used to randomly generate 1,000 people to mimic the distribution of bodily features seen in the AMASS dataset [6]. This was done by randomly generating a number between -6 and 6 for each of the parameters that control the shape of the body. This process of selecting numbers allows for visualization of the correlation between shape parameter value, and magnitude of nodal movement. More extreme nodal movement is captured in outlier shape parameter values.

Two pieces of information are stored after a person is generated: the value of their shape parameters, and the distance each node on the mesh has moved from the default mesh position (Figure 4.3). This is combined to create a correlation matrix where only the correlations between shape parameters and nodal movements are considered (Figure 4.4).

The rigid point cloud alignment step is to position the Kinect point cloud and skeleton in the coordinate system of the SMPL body model. The later optimization steps are concerned with articulating and morphing the SMPL model to best match the point cloud, so it is important to initially position the Kinect point cloud as close to the model as possible. The rigid transformation occurs in two steps: the Kabsch algorithm and the iterative closest point (ICP) algorithm. The Kabsch algorithm roughly positions the Kinect skeleton and point cloud using a rotation matrix defined by the Kinect and SMPL joints, and the ICP algorithm makes fine adjustments to the position of the Kinect point cloud and joints to better align them with the SMPL mesh.

The method used to morph the SMPL model out to the Kinect point cloud is achieved with the use of a radial basis function (RBF) algorithm originally used to fix overclosures in meshes used for finite element analysis [47], [48]. The RBF algorithm is significantly faster and outperforms similar methods, such as a coherent point drift (CPD) algorithm. Morphing one template SMPL instance to a Kinect point cloud takes ~45 minutes using CPD; however, the RBF algorithm can produce better results in only ~30 seconds (Table 4.9). It is quick to train the radial basis functions, and the process is made quicker by originally decimating the mesh and gradually bringing the mesh density back to its original size. First, 10% of the SMPL mesh and Kinect point cloud each are kept and used to train the RBF and deform the SMPL model. Then, 50% is kept and used, and then finally the full density mesh is used. The mesh reduction steps reduce the number of nearest neighbor calculations within the algorithm.

The initial SMPL model estimation aims to recover a SMPL model most closely associated with the mesh deformed with the RBF algorithm. The shape of the model is controlled by principal component scores learned from the dataset used to create the

SMPL model. The eigenvectors associated with each principal component are available within the SMPL's open-source python scripts. Quick math using this matrix and the RBF deformed SMPL model extracts a model that resembles the person represented in the Kinect scan.

The optimization for final adjustments can be broken up into two phases: an initial pose estimation, and a computationally expensive optimization using k-nearest neighbor (KNN) calculations that calculate the distance between the SMPL mesh and Kinect point cloud. The latter optimization is highly non-convex, so the initial SMPL model shape and pose estimation create a starting point in a region for the steepest descent algorithm that will create an appropriate solution after many iterations. In the initial pose estimation, an objective function is posed to minimize the distance between the Kinect and SMPL joints. The SMPL model is articulated to align more closely to the Kinect skeleton. After this initial positioning, the optimization problem minimizes the distance between the SMPL mesh and Kinect point cloud by alternating optimizing the shape and pose parameters.

4.2.1 Rigid Point Cloud Alignment

To begin the process of creating a subject-specific body model, the template body model and subject point cloud data must be aligned in the same coordinate system. This is done using a Kabsch and an iterative closest point ICP algorithm. The SMPL model being aligned is the generic, average body model, and does not represent anything about the size or shape of the Kinect point cloud. This step is focused on alignment.

A Kabsch algorithm calculates the optimal rotation matrix to minimize the root mean squared deviation between two ordered sets of points. This is an ideal algorithm to use for two sets of point clouds with the same number of points in a similar shape. The rotation matrix is found by first translating both point clouds such that their centroids are the origin of the coordinate system. This is done by mean centering both point clouds:

(1)

$$S = S_o - \bar{S}_o$$
$$K = K_o - \bar{K}_o$$

The covariance matrix, H, between the two points is calculated:

(2)

$$H = S^T K$$

If the covariance matrix does not have an inverse, singular value decomposition (SVD) can be used to calculate the optimal rotation matrix. This method used SVD to find the rotation matrix to eliminate computational problems. With SVD, the covariance matrix can be represented with three matrices:

(3)

$$H = U \Sigma V^T$$

Using this decomposition, the optimal rotation matrix R can be calculated as:

(4)

$$R = V \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & d \end{bmatrix} U^T$$

The variable d is used to correct the rotation to account for a right-handed coordinate system. It is the sign of the determinant of the transpose of the left matrix multiplied by the right matrix.

(5)

$$d = \text{sign}(det(VU^T))$$

This was used as a method of rough alignment to transform the Kinect point cloud and joints into the coordinate system of the SMPL body model before applying an ICP algorithm to make fine adjustments to the Kinect point cloud. The shoulders and hips from the Kinect body model and SMPL skeleton are selected to be the ordered pairs of points for the Kabsch algorithm (Figure 4.5). After calculating and applying the transformation matrix to the Kinect points, the same transformation matrix is applied to the Kinect point cloud (Figure 4.6).

The Kabsch algorithm aligns the two point clouds close enough for an ICP algorithm to work effectively. If the Kabsch algorithm was not used, the Kinect skeleton and point cloud would not be able to flip over, and the point clouds would be aligned with the Kinect head by the SMPL feet and the Kinect feet by the SMPL head. After applying the rotation and translation from the Kabsch algorithm, there is still a slight misalignment between the Kinect point cloud and the SMPL mesh (Figure 4.7). The ICP algorithm finds the best transformation to align the Kinect point cloud with the SMPL mesh and does not use the Kinect or SMPL joints. The algorithm is run until a correspondence convergence criterion is met. When the same nodes are found in the correspondence matrix two iterations in a row, the algorithm stops. Alignment is verified visually before proceeding.

4.2.2 Mesh Morphing

The RBF algorithm is invariant to mesh density, as it uses the location of both the source and target nodes to calculate a deformation field. The algorithm performs three key steps: source to target and target to source KNN registration, calculating and storing vector distances between all registered target and source pairs, and training a set of radial basis functions to perform mesh deformation (Figure 4.8).

The KNN registration step first finds the nearest neighbor in the source nodes for each point in the target nodes. For the partial point cloud data, this registration occurs between the point cloud and a SMPL mesh with vertices removed that correspond to incomplete point cloud data (Figure 4.9). Then, a vector is created between the source nodes and their nearest neighbor in the target nodes. This step is repeated, but by finding the nearest neighbor in the target nodes for each point in the source nodes. Both the target and source nodes are combined, along with the vectors between them. The nodal locations are used as inputs to an RBF network, which is trained to generate a deformation field that matches the calculated vectors between source and target nodes.

To reduce computational overhead, the RBF network and mesh deformation happens in three stages, each with a different level of mesh decimation. First, the Kinect point cloud and SMPL mesh are down sampled to be 10% of their original size. The RBF algorithm is implemented, and a radial basis function is trained to deform the down-sampled SMPL mesh to the down-sampled Kinect point cloud. Then, the RBF algorithm is trained again to morph the full mesh to the decimated and deformed mesh (Figure 4.10). This process is done by creating vectors between the decimated and deformed mesh and the corresponding nodes on the full mesh. The RBF algorithm can learn a new deformation field from these partial point clouds that can be quickly applied to the full point cloud.

After the full mesh is morphed, this process starts over again, only now both the SMPL mesh and Kinect point cloud are down sampled to be 50% of their original size. The RBF algorithm learns a deformation field to morph the decimated mesh, then learns another deformation field to morph the full mesh to the decimated and deformed mesh (Figure 4.11). Once the full mesh is morphed again, the process repeats one more time, but neither the SMPL mesh nor the Kinect point cloud is down sampled, and 100% of the mesh and point cloud are used (Figure 4.12). The same process can be repeated for the partial point cloud and the SMPL mesh with select vertices removed (Figure 4.13). Figures 4.14 and 4.15 show a comparison between the positioning of the SMPL mesh at each of these morphing steps.

4.2.3 Initial SMPL Model Estimation

Principal component analysis (PCA) is used as a method of performing dimensionality reduction on large datasets. After performing PCA, some accuracy is lost, but key dimensions of the data with their importance can be highlighted. The SMPL model can be manipulated using learned principal components of human anatomy. The model was created by registering a mesh with the same 6980 node topology to thousands of 3D scans of people, generating a large dataset of realistic body shapes described with 6980 nodes. Performing PCA on this dataset reveals eigenvectors and eigenvalues of the NxN covariance matrix whose entries are the covariances of all possible pairs of samples in the dataset, where N is the number of samples. The dimensionality reduction of PCA is achieved by compressing as much information as possible into these eigenvalue and eigenvector pairs. Each pair is uncorrelated, and the eigenvector with the largest eigenvalue associated with it contains the most information about the dataset. By arranging the eigenvectors in order of descending importance with their eigenvalue pair, the dimensions with the most to least variance can be extracted. If there are N variables in the dataset, there will be N pairs of eigenvectors and eigenvalues. Dimensionality is reduced by keeping only the eigenvectors with the highest eigenvalues. After preserving the most important eigenvectors, any member of the original dataset can be recreated using its principal component scores:

(4)

$$[Instance] = [PopulationMean] + [EigVect] \times [PCScore]$$

Or a new member of the population can be estimated with new principal component scores:

(5)

$$[NewInstance] = [PopulationMean] + [EigVect] \times [NewPCScore]$$

The eigenvectors from the dataset used to create the SMPL model can be found within its open-source python scripts. The shape parameters, Betas, are the eigenvalues or principal components that can be combined with the eigenvectors to create an instance in the population. The concepts of PCA allow for the creation of any person in the dataset used to create the SMPL model, but also allows for the creation of a person not in the dataset. The goal is to extract the principal component scores, or betas, associated with a new instance. This can be achieved by reposing and rearranging (6). The following equation setup is used to generate SMPL models:

(6)

$$[NewInstance]_{3*6890\times 1} = [TemplateInstance]_{3*6890\times 1} + [EigVect]_{3*6890\times 10} \times [Betas]_{10\times 1}$$

To isolate the Betas variable, the equation is manipulated by subtracting the template instance from the new instance, and multiplying by the pseudo inverse of the eigenvector matrix:

(7)

$$[NewInstance]_{3*6890\times 1} - [TemplateInstance]_{3*6890\times 1} = [EigVect]_{3*6890\times 10} \times [Betas]_{10\times 1}$$

$$[EigVect]^{+}_{10\times 3*6890} = [EigVect^{T} \times EigVect]^{-1} \times [EigVect]^{T}$$

$$[EigVect]^{+}_{10\times 6890*3}([NewInstance]_{6890*3\times 1} - [TemplateInstance]_{6890*3\times 1}) = [Betas]_{10\times 1}$$

If the new instance is being estimated using the partial point cloud data, a modified version of this equation is used where mesh vertices associated with incomplete or missing point cloud data are removed from the eigenvector, new instance, and template instance matrices:

(8)

$$[EigVect']^{+}_{10\times 4136*3}([NewInstance]_{4136*3\times 1} - [TemplateInstance]_{4136*3\times 1}) = [Betas]_{10\times 1}$$

The eigenvector and template instance matrix is known in (8), and the mesh vertices of the new instance can be estimated using the RBF mesh morphing technique described in section 4.2.2. The workflow for estimating a SMPL model from partial point cloud data is described in Figure 4.16, and in Figure 4.17 for a full point cloud.

The principal component scores associated with the new instance may be slightly inaccurate if the person is not posed close enough to the default T-pose of the template instance, or if too many vertices of the SMPL model are removed. Due to this, any entries in the Betas vector that are larger than 3 are reset back to 3, and any that are smaller than -3 are reset to -3. This step is to ensure the SMPL model that initializes the optimization represents a realistic person. The frequency of this occurring depends on how close the person is to the default SMPL pose in their scan.

4.2.4 Optimization

The first optimization is a rough pose estimation to position the SMPL instance. This uses the Kinect joint centers and the SMPL skeleton. The idea of this computationally inexpensive initial step is to articulate the SMPL model so its mesh is closer to the Kinect point cloud. The structure of these skeletons differs slightly in the torso and hips, but the position of many joints like the wrists, elbows, shoulders, knees, ankles, and toes are similar. To formulate the objective function, a registry of these similar joints is created (Figure 4.18), SR and KR. Each is a 12x3 matrix where the ith row of each matrix is the coordinate of a similar joint. For example, the first row of the SR matrix is the coordinate of the SMPL left shoulder joint, and the first row of the KR matrix is the coordinate of the Kinect left shoulder joint. The objective function minimizes the distance between similar joints on the Kinect and SMPL skeletons, essentially minimizing the distance between these two matrices. It is a function of the 63 pose parameters that control the joint angles at each of the 21joints.

(9)

$$f(\theta) = \sum_{i=1}^{12} ||KR(i) - SR(i)||_2$$

This, like all optimizations in this method, is a zeroth-order optimization problem. Only the value of the objective function is known, so the gradient must be numerically calculated at each step [49]. The gradient can be approximated by perturbing the current design point in random directions many times and recording the change in the objective function for each variable. First, a random direction in 63-dimensional space is constructed:

(10)

$$z \in R^{63},\ ||z||_2 = 1,\ z = \frac{a}{||a||_2},\ \text{where } a = randn(63,1)$$

The current design vector is perturbed a small distance, δ, in this random direction.

(11)

$$\left(\frac{f(\theta^k + \delta z) - f(\theta^k)}{\delta}\right) z \quad \text{Where } \delta = 0.0001$$

This is repeated 64 times and an average is taken to obtain a good estimation of the gradient. With every new estimation, a new random direction is selected.

(12)

$$\widetilde{\nabla} f(\theta^k) = \frac{1}{L} \sum_{i=1}^{L} \left(\frac{f(\theta^k + \delta z_i) - f(\theta^k)}{\delta}\right) z_i \quad L = 64$$

After calculating the gradient at the current design point, it is used in gradient descent to modify the joint angles to position the SMPL skeleton closer to the Kinect joint centers. There is a point where continuing the optimization gives little change in the position of the SMPL mesh. Fifteen iterations of gradient descent were arbitrarily selected because the SMPL mesh changes little when compared to one created from 50 iterations (Figures 4.19 and 4.20). If less than 10 iterations are used the pose of the SMPL model does not change enough.

(13)

$$\theta^{k+1} = \theta^k - \alpha\widetilde{\nabla}f$$

Once the SMPL pose and shape are estimated, the more computationally expensive KNN optimization can begin in a region of the design space that is more convex and a steepest descent algorithm is more likely to arrive at the correct solution.

The objective function minimizes the distance between each node on the SMPL mesh and the nearest point in the Kinect point cloud. The objective function using the KNN algorithm contains three groups of node sets, each weighted differently to obtain a desirable result. The first node set is the entire SMPL mesh, or the partial mesh if using a partial point cloud. The second node set includes only the torso, and the third node set contains only nodes of the hands and feet (Figure 4.21).

(14)

$$f = \sum_{i=1}^{N} ||SV^i - KPC_{nearest}||_2 + W_{torso}\sum_{i=1}^{T} ||SV^i_{Torso} - KPC_{nearest}||_2 +$$

$$W_{extrem}\sum_{i=1}^{E} ||SV^i_{Extrem} - KPC_{nearest}||_2$$

The node set with the entire mesh is for general shaping and posing. The node set with only the torso is weighted the highest to force the optimization to focus on minimizing the distance between the SMPL torso and the Kinect point cloud torso. The magnitude of the weight was determined experimentally. It was discovered that, without this term in the objective function, the SMPL model gets stuck in the steepest descent algorithm and the result is too large in the torso and hip area. The final node set with only the extremities is to balance the effects of the weighting on the torso. If this term is not included in the objective function, the algorithm focuses too much on fitting the torso, and the extremities are an afterthought. Including this term in the objective function ensures the arms and legs are not too short.

This part of the optimization occurs in cycles, where the gradient alternates between being calculated with the shape parameters, and the pose parameters (Figure 4.22). When the gradient is calculated with the shape parameters, an optimal step is calculated in this direction, and one iteration of gradient descent is performed. Then, when the gradient is calculated with the pose parameters, the gradient is calculated, the optimal step is calculated, and one iteration of gradient descent is performed. The gradient using the pose parameters is calculated in the same way as (12), and the gradient for the shape parameters is almost identical. The design vector, Betas, has 10 variables, so the gradient has 10 entries, and the random direction must have 10 entries:

(15)

$$z \in R^{10}, \; ||z||_2 = 1, \; z = \frac{a}{||a||_2}, \text{ where } a = randn(10, 1)$$

$$\widetilde{\nabla} f(\beta^k) = \frac{1}{L} \sum_{i=1}^{L} \left(\frac{f(\beta^k + \delta z_i) - f(\beta^k)}{\delta} \right) z_i \quad L = 11$$

$$\beta^{k+1} = \beta^k - \alpha \widetilde{\nabla} f$$

Adjusting the shape of the model moves the vertices of the mesh into a position that will be easier to articulate closer to the Kinect point cloud, and articulating the model allows the SMPL mesh to be positioned closer to the Kinect point cloud. Though the shape of the model is the desirable parameter, optimizing this alone will not arrive at the best solution. It was experimentally determined that 75 cycles of this process would occur, alternating shape and pose with each iteration. The choice of 75 cycles can be explained in Figure 4.23 where the objective function decrease is compared for multiple subjects. The objective function's journey downhill is different for each subject, but the rate of decrease has generally leveled off by 75 cycles, or 150 iterations.

To measure the accuracy of the generated SMPL model, extensive anthropometric measures of each subject were gathered with a tailor's tape prior to gathering depth data. These include measurements of the circumference of different parts of limbs and the torso (Figure 4.24), and the lengths of limbs and height (Figure 4.25). It can be inferred from these images where and how to collect most of the measurements, but a few require explanation. All measurements were taken with the subject standing in a pose as close as possible to the default SMPL T-pose: feet shoulder width apart and arms outstretched with palms down. The subjects positioned the tailor's tape for measurements of the hips, upper thigh, and chest measurements. Male subjects received just one measurement across the upper chest. Female subjects received two measurements: the "Cup Size" measurement of the fullest part of the chest, and the “Band Size” measurement of the chest underneath the breasts where the band of a bra stretches around the ribs (Figure 4.26). Measurements of the torso and legs were taken with special care to ensure the tape was parallel to the ground. This care was extended to measurements of the arms to keep the tailor’s tape perpendicular to the ground. The “Groin to Floor” measurement was collected by having the subject hold one end of the tailor’s tape at their groin, where the leg emerges from the torso. Then, they stood on an elevated platform in such a way that the other end of the tape could extend below the surface of the platform. The measurement from the groin to the surface of the platform was recorded as the “Groin to Floor” measurement. To be able to compare the subject-specific SMPL model to the subject it was created with, anthropometric measures were digitally recreated on the SMPL model (Figure 4.27). The correlation between manually collected and SMPL measurements was calculated, and a p test was performed to quantify the statistical significance of this relationship.

The methods described in this section were applied to both partial and full point cloud data to produce 16 SMPL models representing subjects collected with the partial point cloud method, and 4 SMPL models representing subjects collected with the full point cloud method.

4.3 Results

There was a positive correlation between subject manual measurements and those generated from the optimized SMPL instance fit with the partial data with an $R^2 = 0.99$ and $P << 0.01$ (Figure 4.28).

The validity of the model was further quantified with a percent error calculation between subject manual and SMPL measurements. A Bland-Altman chart is used to quantify the agreement between the manual and model-based anthropometric measurements across all measures and subjects (Figure 4.29). Bland-Altman charts are also generated to further understand errors in the legs (Figure 4.30), arms (Figure 4.31), and torso (Figure 4.32).

Across all measures, there is an average absolute percent difference of 4.71 ± 4.10% and the average absolute percent error between any two measures never exceeds 10%. The measurement with the best performance is the floor to shoulder measurement with 1.17 ± 0.76%, and the worst measurement is the ankle circumference with 9.80 ± 6.33% (Table 4.1). Two exemplary subjects with the lowest absolute percent error across all measures are highlighted in Figures 4.33 and 4.34.

The body models created from the full-body point cloud data were compared to those of the same subjects from the partial data in a variety of ways. An overlay was created between the model created from the full and partial point cloud (Figures 4.36 – 4.39). Visually, most of the meshes are close, however the principal component scores used to create them are not appropriately similar (Tables 4.3 – 4.6). The full point cloud data does not produce more accurate SMPL models despite containing more information about the subject's data, but measurements are still highly correlated (Figure 4.35) an $R^2 = 0.99$ and $P << 0.01$. The average absolute percent error between physical and SMPL measurements is 7.12 ± 5.62%. The measurement with the worst performance is the ankle with 15.40 ± 3.70%, and the best is the floor to shoulder measurement with 1.42 ± 1.35% (Table 4.7).

Though the SMPL models created with the full point cloud do not create a more accurate representation of the subject, they do perform better when inspecting the KNN error. The total distance between SMPL mesh and Kinect point cloud is normalized to the number of nodes used in the calculation: 6890 for the full point cloud and 4136 for the partial point cloud. For the 4 subjects recollected, the full point cloud results in an SMPL model with the lowest normalized KNN error (Table 4.2).

Correlations between shape parameter value and magnitude of nodal movement can be visualized on the neutral mesh to understand which areas of the body are controlled by each shape parameter (Figures 4.40 – 4.49). A heatmap has been created for each shape parameter. To read and understand these graphs fully, one must pay special attention to the colorbar legend. This shows the range in correlation across the entire mesh for a particular shape parameter. The darkest red is associated with the highest positive correlation between shape parameter and node movement, and the darkest blue is associated with the lowest correlation. Usually, the color bar indicates both positive and negative correlations. This means we can identify areas influenced by increasing the parameter (red) and areas influenced by decreasing the parameter (blue). Unfortunately, only the magnitude of mesh movement is known, not the direction.

We are also able to gauge how important each parameter is in controlling the mesh by observing how strongly it is correlated with mesh movement. The first shape parameter has the highest correlation with mesh movement, and this correlation decreases with each parameter. The maximum correlation for parameters 8, 9, and 10 all hover around 0.1.

4.4 Discussion

In general, this workflow using the partial point cloud data generated high quality computational representations of each subject. Percent error is relatively consistent across measures with a few exceptions. The model predicts measurements about the height, arms, and torso most accurately, while the predictions about leg measurements have more error. The band size measurement on women is the worst prediction.

Contributions to this error must be recognized and addressed. A discrepancy between model-predicted and manually collected measurements could exist because the manual measurements were not collected consistently, or because the SMPL model was not measured well enough. Code was designed to measure the SMPL model at the locations where subjects were measured in the lab (Figure 4.27), but there are sometimes exceptions to the conditions used and the model is not measured accurately. Additionally, the way of measuring the model is an approximation from slicing through the nodes and elements that constitute the SMPL mesh. When measuring subjects in the laboratory, special care was taken to measure them consistently, but there are likely still a few centimeters of error that was introduced from the positioning of the measurement tape.

The difference in prediction accuracy between the SMPL models fit with partial and full point cloud data is interesting (Figures 4.36 – 4.39). One would think the model would perform better when there is a more complete picture of the surface of the subject, and this is not the case. One contribution to this difference could be that the subject measurements were not recollected when the full point cloud collection was conducted 10 months after the initial partial point cloud acquisition. Some circumference measurements may differ, but in general the people should be about the same shape, and measurements of lengths of the body should not change. This explanation does not account for most of the errors. One would also expect the shape parameters to be more consistent between the full and partial point cloud groups. The observed differences in the parameters are likely a result of differences in the point cloud shape and uncertainty in the model fitting process, and likely warrants further investigation. The alignment process to stitch the full point cloud together may introduce error. While the calibration object works as intended, the points of the foam spikes are sometimes difficult to identify, although all alignments were completed with a RMSE under 10%. One final explanation for the difference in accuracy is the design of the model fitting process. The entire procedure was created to work with the partial point cloud data collection method. It is possible something within the algorithm makes using the partial data more favorable. The weighting scheme of the objective function was tailored experimentally for use with this kind of data and may not work effectively with the full point cloud data.

This work establishes a process for subject classification and future studies will investigate links to joint loading and movement patterns. Compared to BMI alone, this approach can support more realistic personalization of musculoskeletal models, particularly for representations of muscle anatomy and inertial properties of segments. Lastly, quantifying body habitus enables consideration of torso-thigh, thigh-calf and other soft tissue contacts, which are known to influence joint loading and kinematics.

4.5 Table and Figures

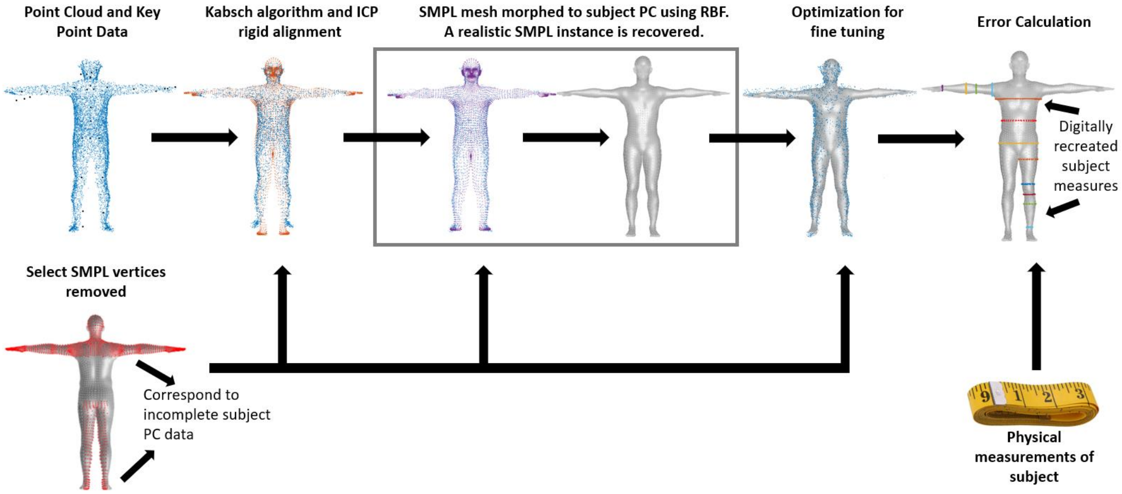

Point Cloud and Key Point Data
Kabsch algorithm and ICP rigid alignment
SMPL mesh morphed to subject PC using RBF. A realistic SMPL instance is recovered.
Optimization for fine tuning
Error Calculation
Digitally recreated subject measures
Select SMPL vertices removed
Correspond to incomplete subject PC data
Physical measurements of subject

Figure 4.1 "Reverse PCA" workflow with radial basis function (RBF) mesh deformation algorithm and partial point cloud data

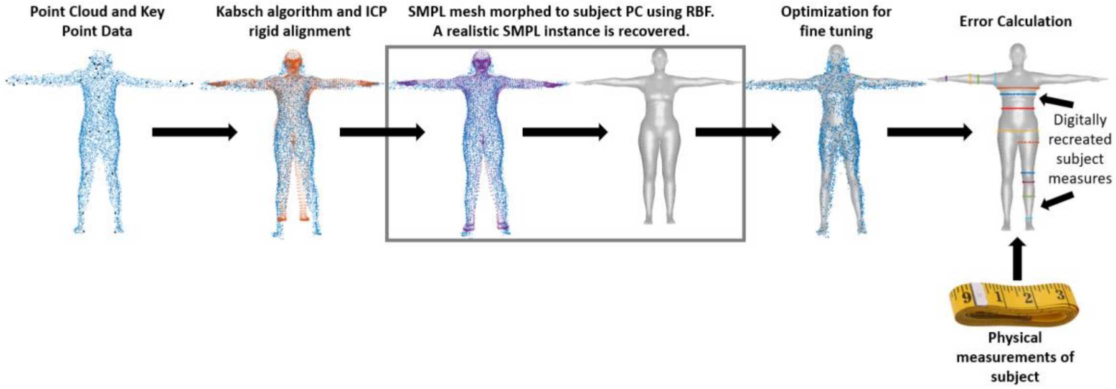

Point Cloud and Key Point Data
Kabsch algorithm and ICP rigid alignment
SMPL mesh morphed to subject PC using RBF. A realistic SMPL instance is recovered.
Optimization for fine tuning
Error Calculation
Digitally recreated subject measures
Physical measurements of subject

Figure 4.2 "Reverse PCA" workflow with radial basis function (RBF) mesh deformation algorithm and full point cloud data

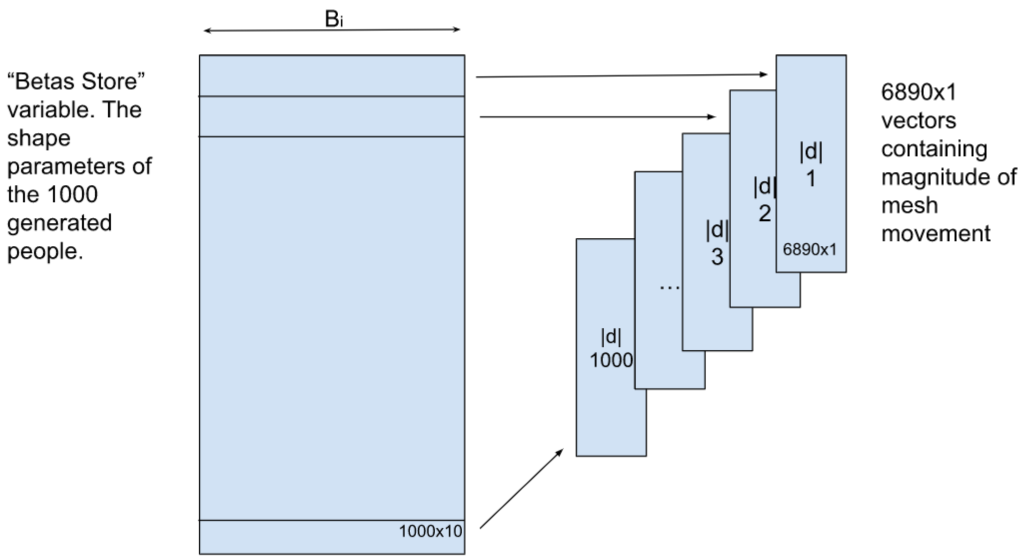

B_i
“Betas Store” variable. The shape parameters of the 1000 generated people.
1000x10
|d| 1
|d| 2
|d| 3
…
|d| 1000
6890x1
6890x1 vectors containing magnitude of mesh movement

Figure 4.3 The creation of the data used in the correlation matrix.

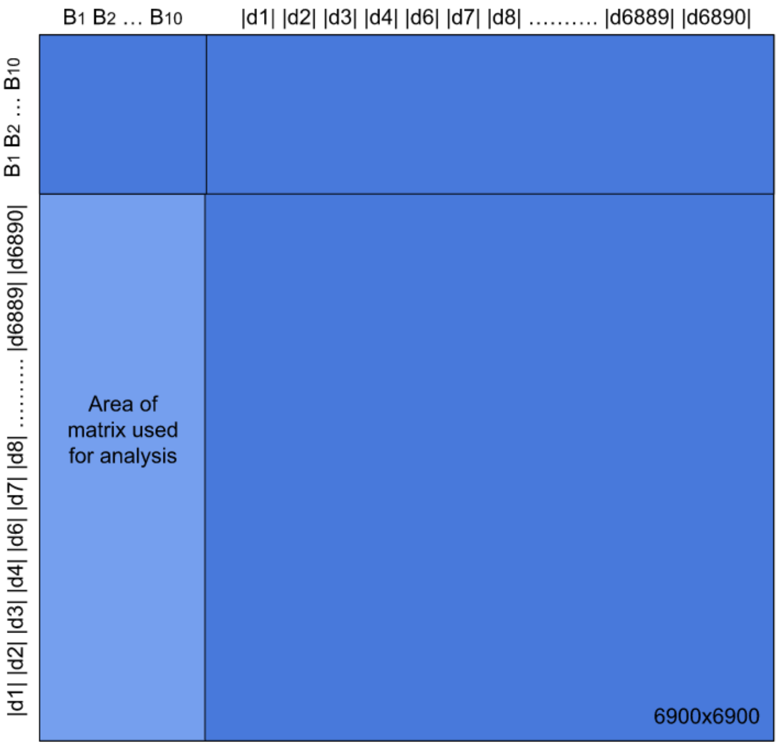

B_1 B_2 … B_{10}
|d1| |d2| |d3| |d4| |d6| |d7| |d8| ………. |d6889| |d6890|
B_1 B_2 … B_{10}
|d1| |d2| |d3| |d4| |d6| |d7| |d8| ………. |d6889| |d6890|
Area of matrix used for analysis
6900x6900

Figure 4.4 The total correlation matrix and the segments used for analysis

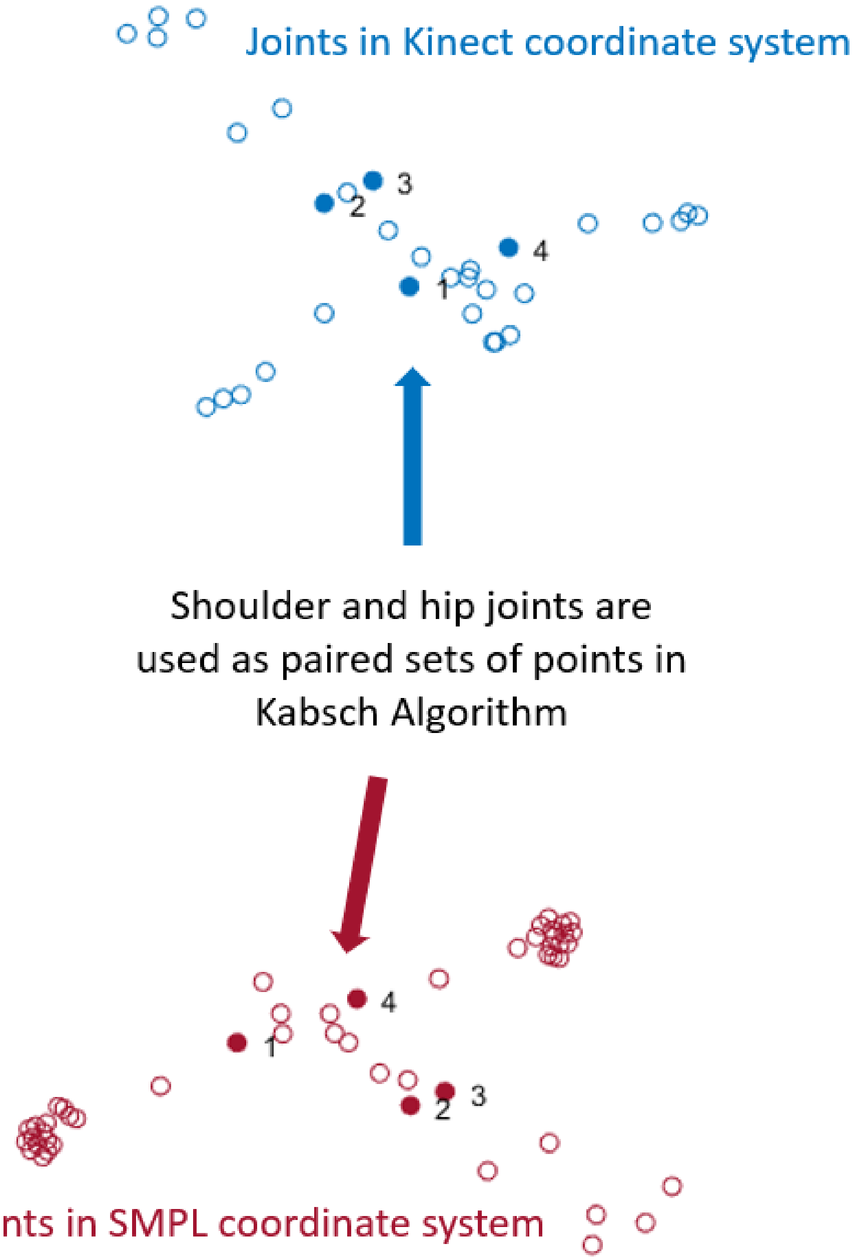

Joints in Kinect coordinate system
3
2
4
1
Shoulder and hip joints are used as paired sets of points in Kabsch Algorithm
4
1
3
2
Joints in SMPL coordinate system

Figure 4.5 The SMPL skeleton and Kinect joints before being aligned using a Kabsch algorithm.

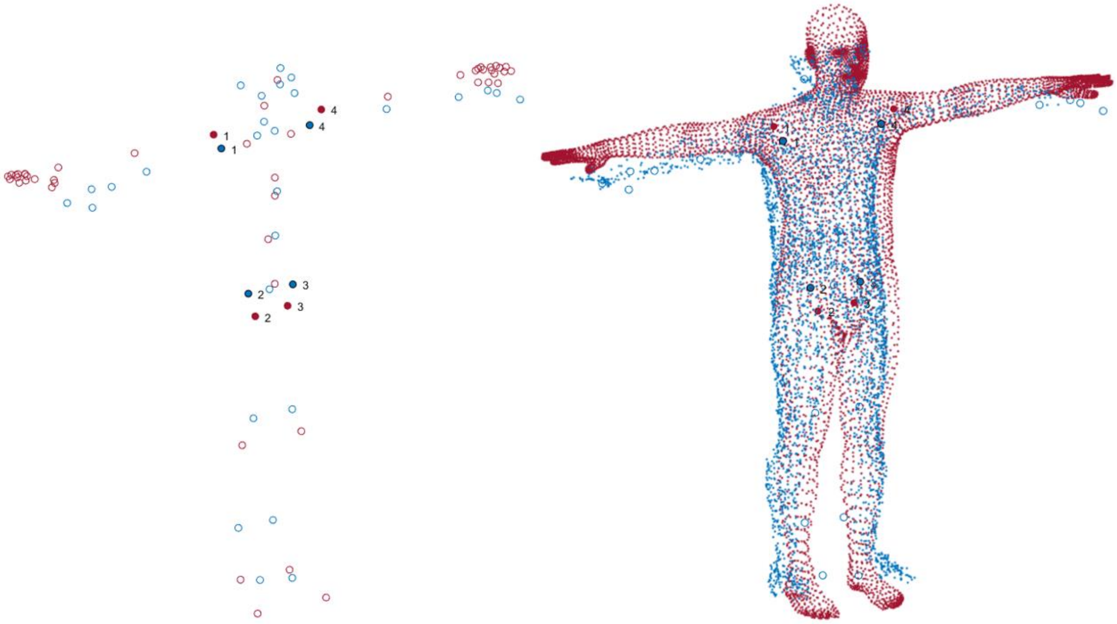


Figure 4.6 The Kinect point cloud and joints are aligned into the SMPL coordinate system with a Kabsch algorithm.

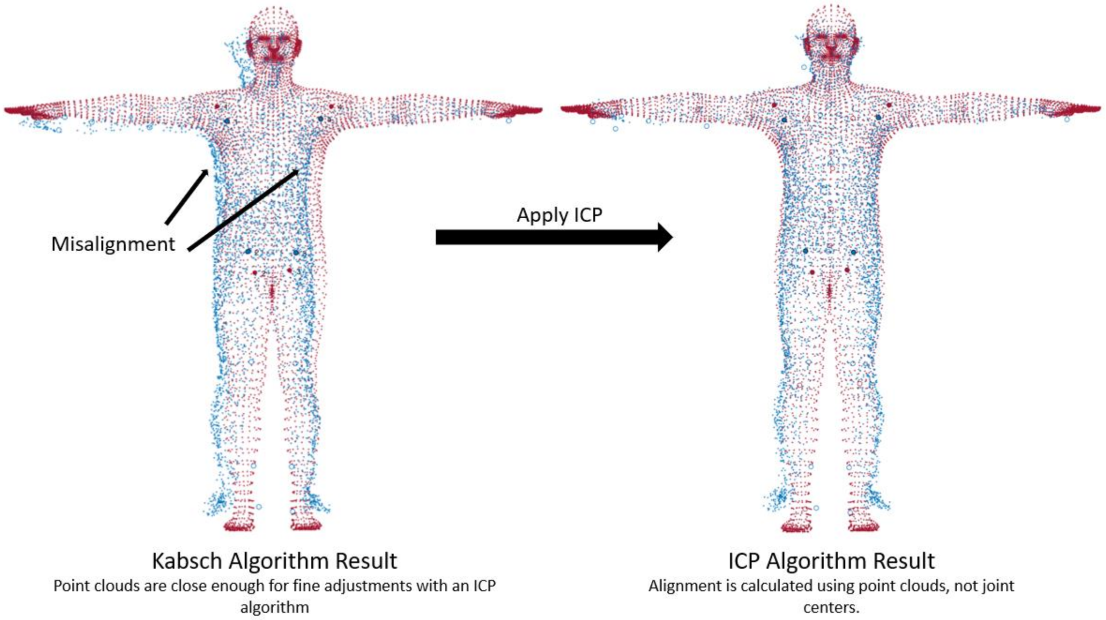

Misalignment
Apply ICP
Kabsch Algorithm Result
Point clouds are close enough for fine adjustments with an ICP algorithm
ICP Algorithm Result
Alignment is calculated using point clouds, not joint centers.

Figure 4.7 The mesh misalignment from the Kabsch algorithm is fixed with the application of ICP.

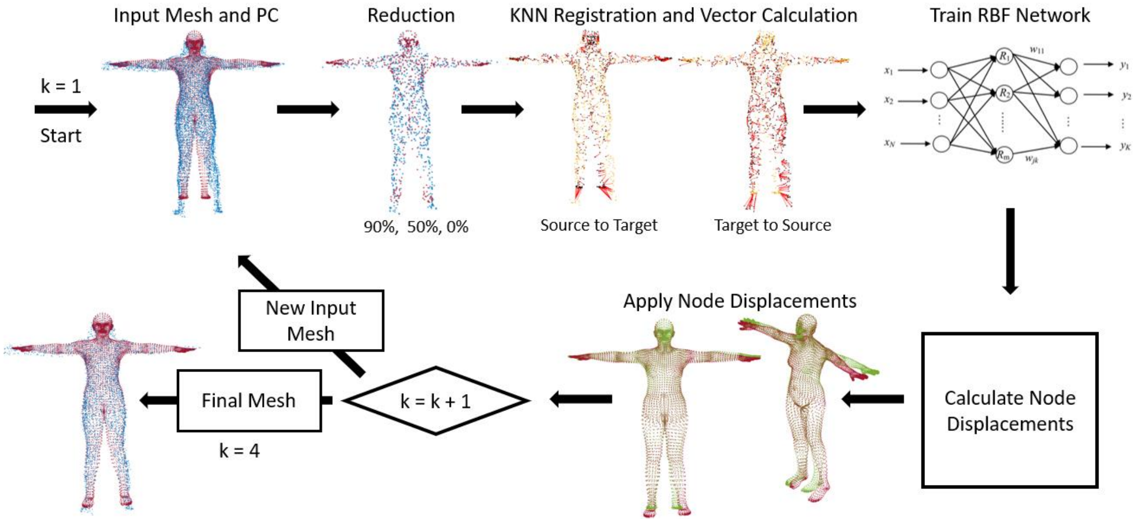


Figure 4.8 A diagram explaining the steps of the RBF mesh morphing algorithm.

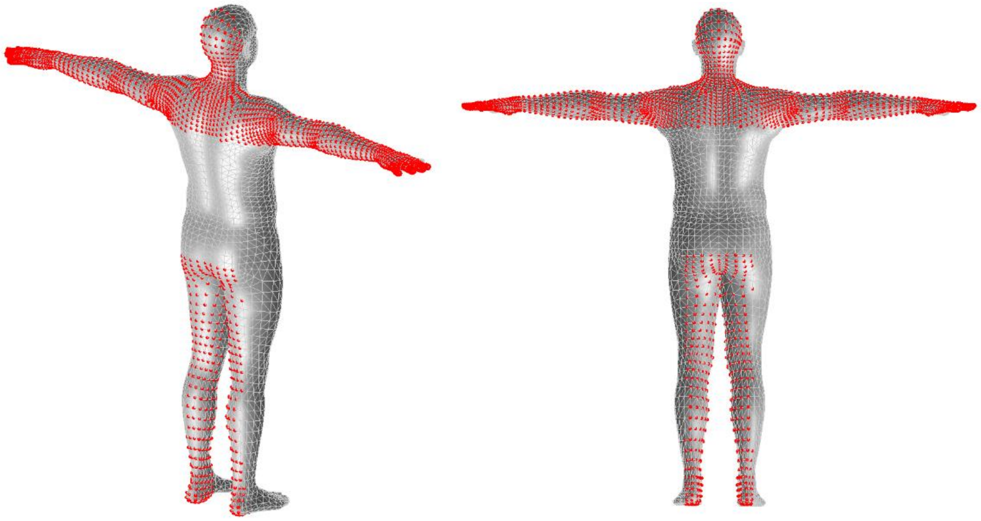


Figure 4.9 The indices removed from the SMPL mesh corresponding to missing sections of the partial point cloud.

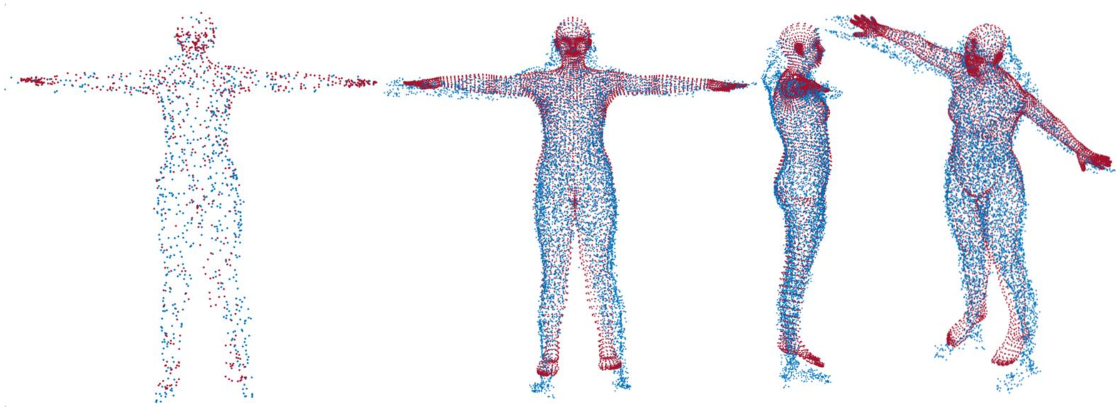


Figure 4.10 (Left) Mesh Morphed using 10% of SMPL mesh and full Kinect point cloud. (Right) Unreduced point cloud morphed to reduced point cloud location.

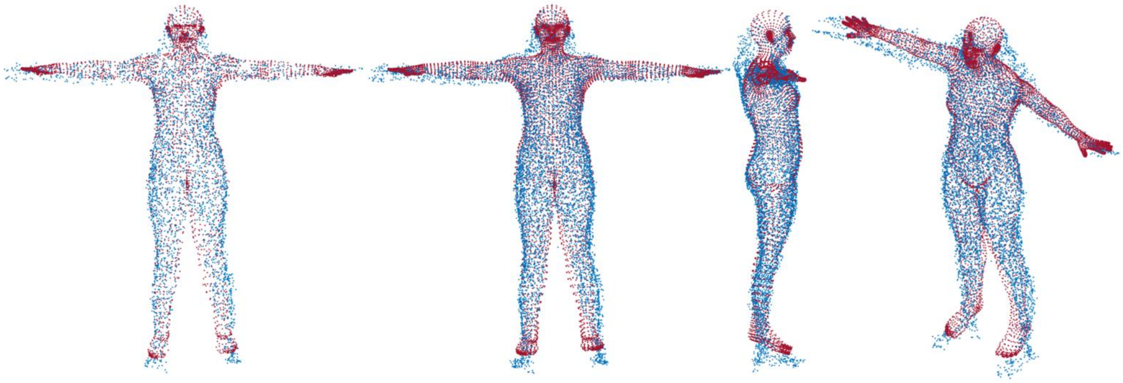


Figure 4.11 (Left) Mesh Morphed using 50% of the mesh from the 10% retrained SMPL mesh and full Kinect point cloud. (Right) Unreduced point cloud morphed to reduced point cloud location.

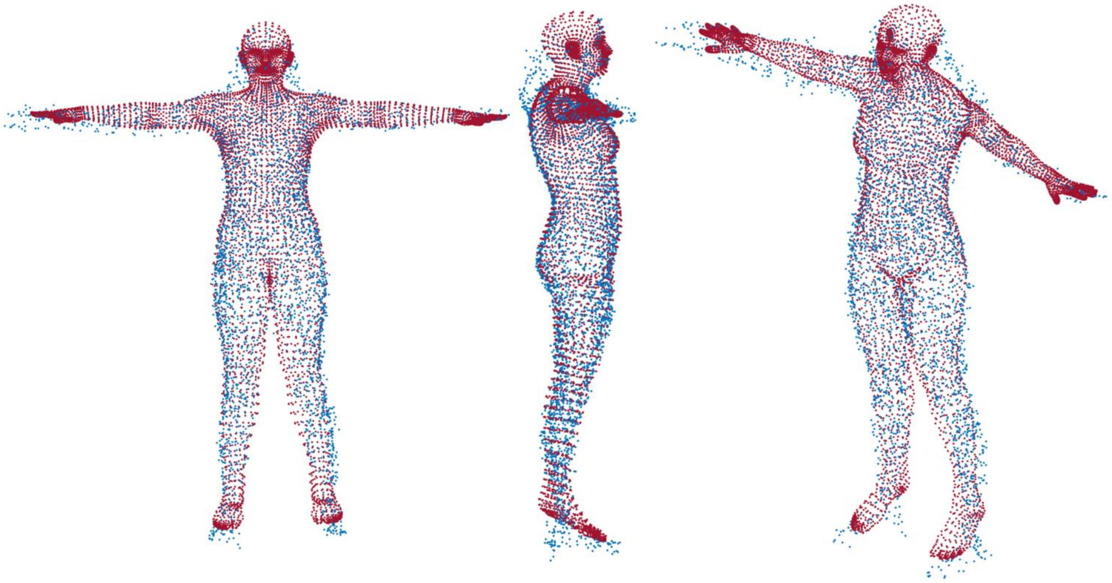


Figure 4.12 Mesh Morphed using 100% of the mesh from the 50% retrained SMPL mesh using the full point cloud.

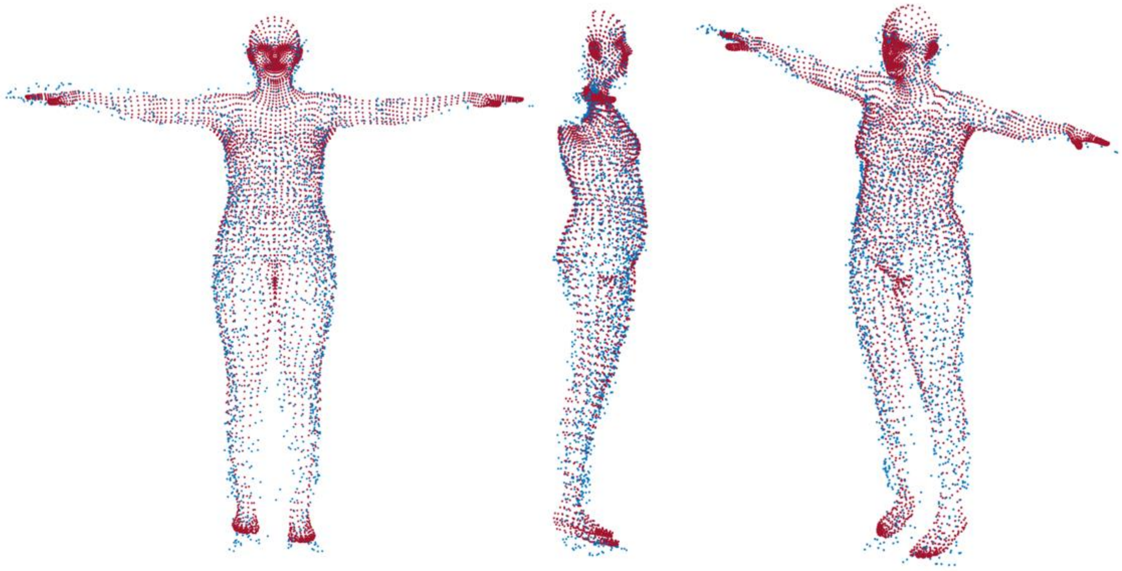


Figure 4.13 Mesh Morphed using 100% of the mesh from the 50% retrained SMPL mesh using the partial point cloud.

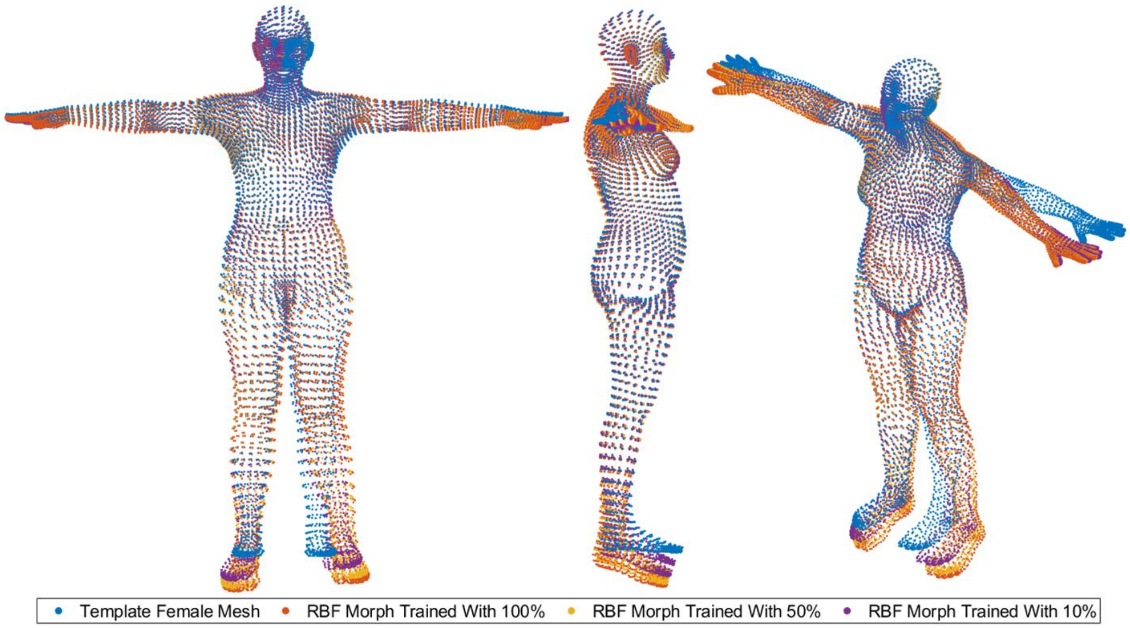

● Template Female Mesh
● RBF Morph Trained With 100%
● RBF Morph Trained With 50%
● RBF Morph Trained With 10%

Figure 4.14 A comparison between the three steps of the RBF mesh morphing algorithm using the full point cloud.

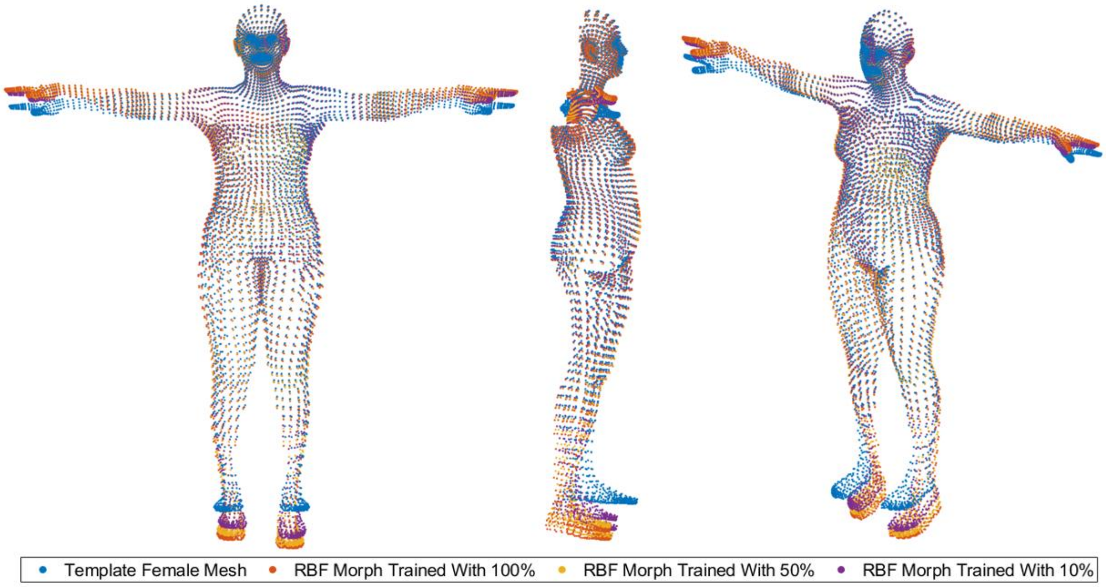

● Template Female Mesh
● RBF Morph Trained With 100%
● RBF Morph Trained With 50%
● RBF Morph Trained With 10%

Figure 4.15 A comparison between the three steps of the RBF mesh morphing algorithm using the partial point cloud.

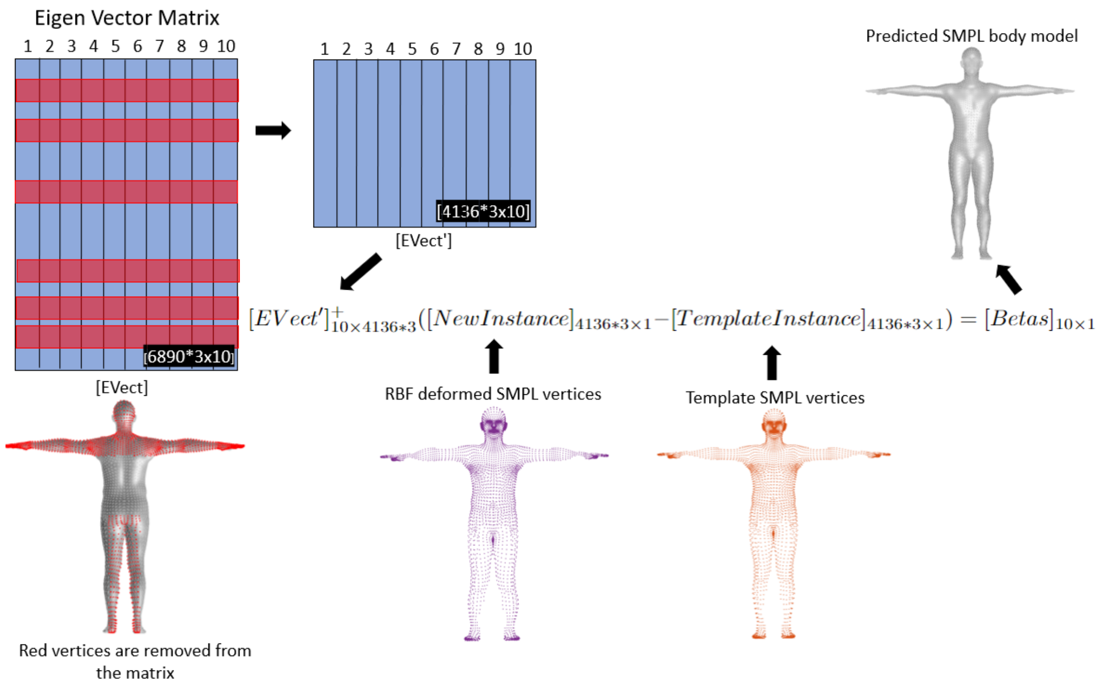

Eigen Vector Matrix
1 2 3 4 5 6 7 8 9 10
[6890*3x10]
[EVect]
1 2 3 4 5 6 7 8 9 10
[4136*3x10]
[EVect']
Predicted SMPL body model
$[EVect']^{+}_{10\times4136*3}([NewInstance]_{4136*3\times1}-[TemplateInstance]_{4136*3\times1})=[Betas]_{10\times1}$
RBF deformed SMPL vertices
Template SMPL vertices
Red vertices are removed from the matrix

Figure 4.16 "Reverse PCA" explanation with sparse point cloud data

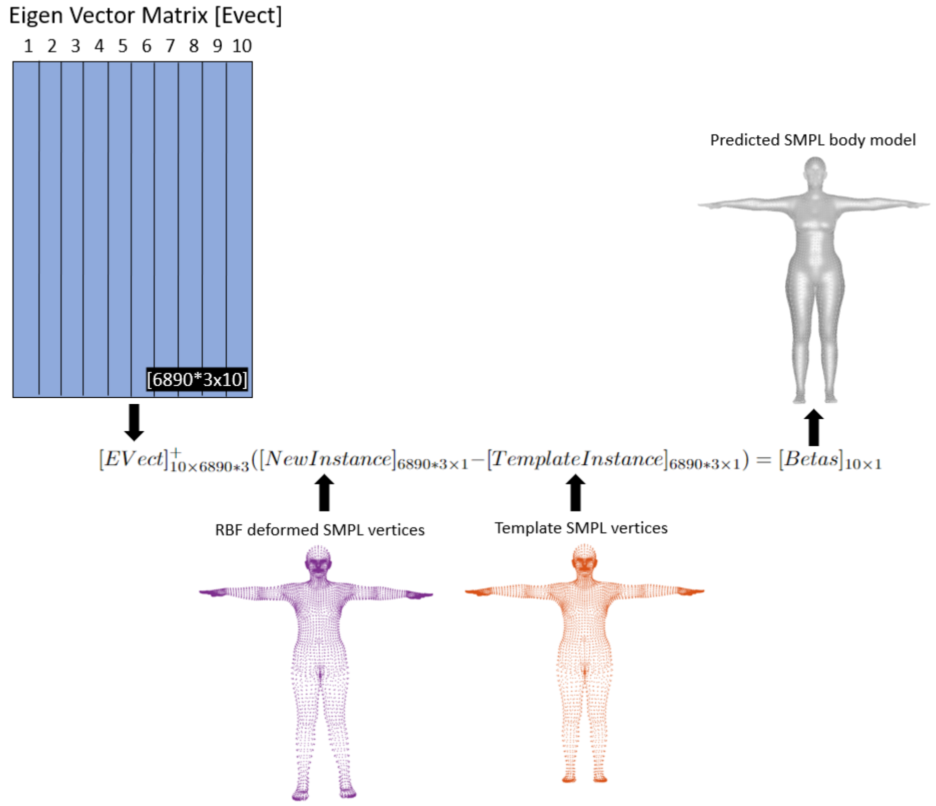
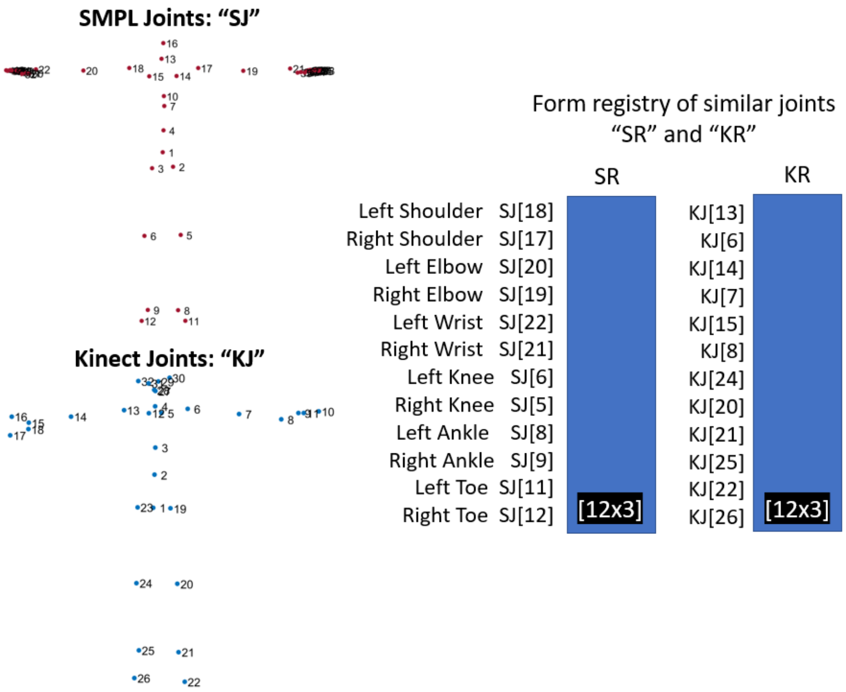

Eigen Vector Matrix [Evect]
1 2 3 4 5 6 7 8 9 10
[6890*3x10]
Predicted SMPL body model
$[EVect]^{+}_{10\times6890*3}([NewInstance]_{6890*3\times1}-[TemplateInstance]_{6890*3\times1})=[Betas]_{10\times1}$
RBF deformed SMPL vertices
Template SMPL vertices

Figure 4.17 "Reverse PCA" explanation with full point cloud data



	SR	KR
Left Shoulder	SJ[18]	KJ[13]
Right Shoulder	SJ[17]	KJ[6]
Left Elbow	SJ[20]	KJ[14]
Right Elbow	SJ[19]	KJ[7]
Left Wrist	SJ[22]	KJ[15]
Right Wrist	SJ[21]	KJ[8]
Left Knee	SJ[6]	KJ[24]
Right Knee	SJ[5]	KJ[20]
Left Ankle	SJ[8]	KJ[21]
Right Ankle	SJ[9]	KJ[25]
Left Toe	SJ[11]	KJ[22]
Right Toe	SJ[12]	KJ[26]

Figure 4.18 The registry of similar joints used in the initial pose optimization.

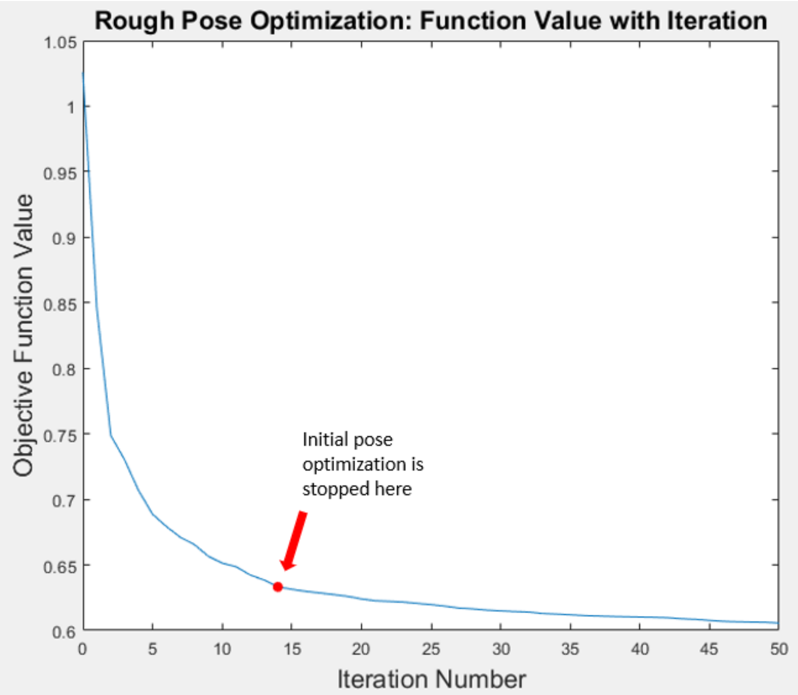


Figure 4.19 Objective function value of the initial pose optimization and iteration. The optimization stopping point is highlighted in red.

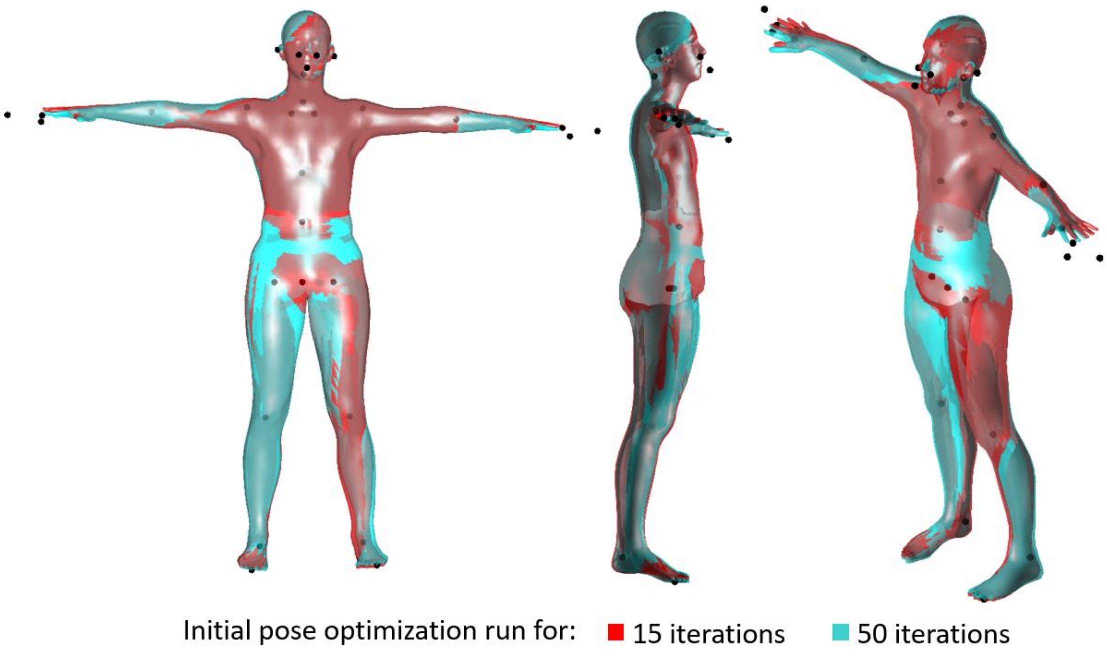


Figure 4.20 The SMPL models created from 50 and 15 iterations of initial pose optimization.

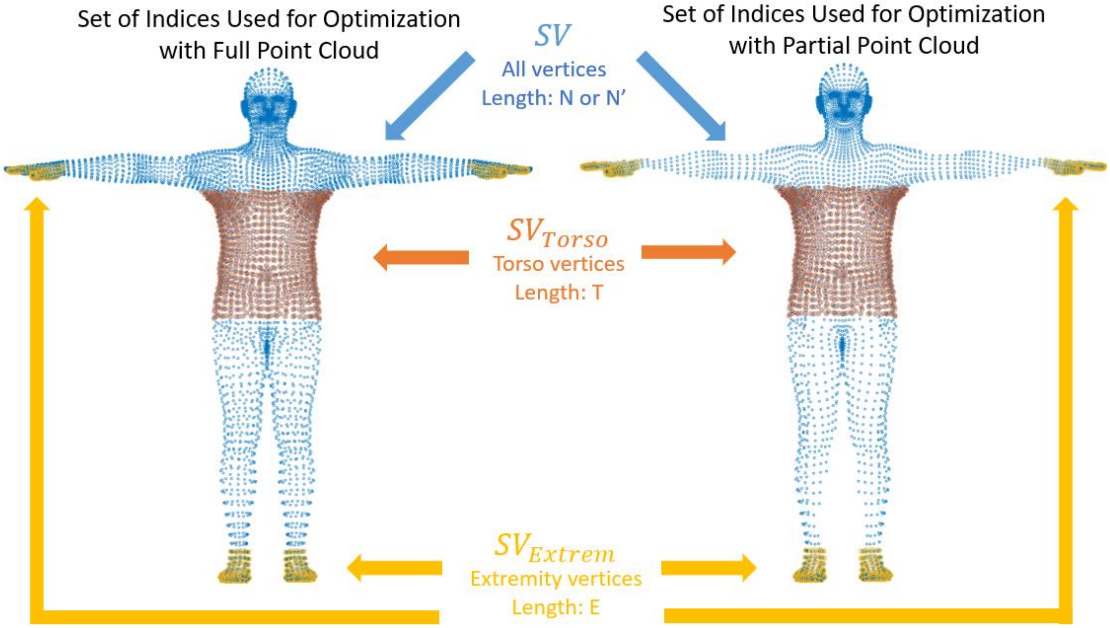

Set of Indices Used for Optimization with Full Point Cloud
SV
All vertices
Length: N or N'
Set of Indices Used for Optimization with Partial Point Cloud
SV_{Torso}
Torso vertices
Length: T
SV_{Extrem}
Extremity vertices
Length: E

Figure 4.21 The node sets used in the KNN objective function for the full and partial point cloud

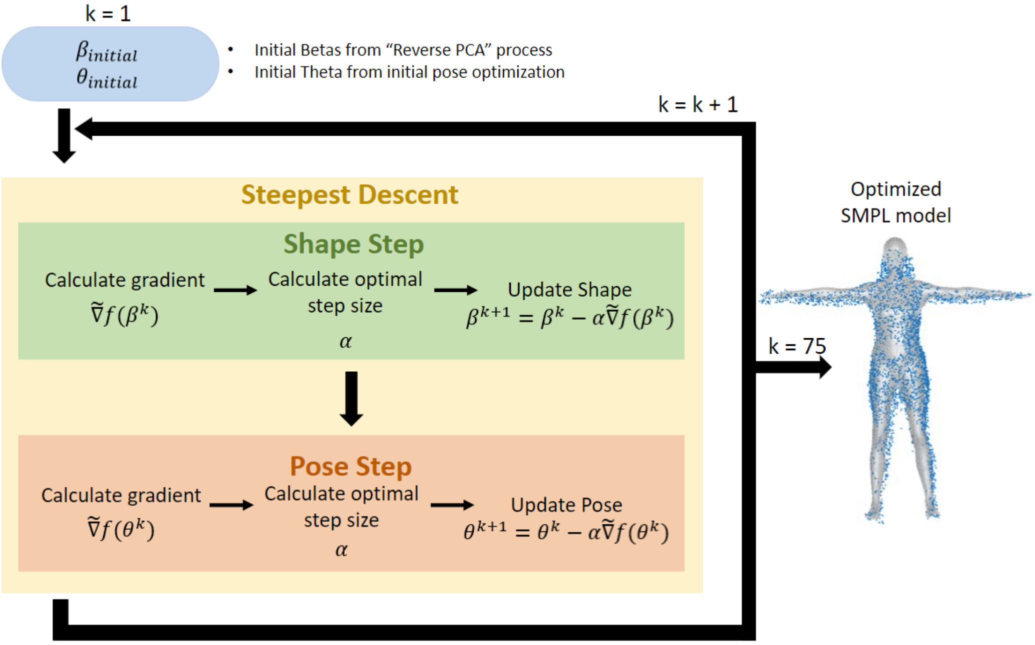

k = 1
$\beta_{initial}$
$\theta_{initial}$
Initial Betas from "Reverse PCA" process
Initial Theta from initial pose optimization
k = k + 1
Steepest Descent
Shape Step
Calculate gradient
$\tilde{\nabla} f(\beta^k)$
Calculate optimal step size
α
Update Shape
$\beta^{k+1} = \beta^k - \alpha\tilde{\nabla} f(\beta^k)$
Pose Step
Calculate gradient
$\tilde{\nabla} f(\theta^k)$
Calculate optimal step size
α
Update Pose
$\theta^{k+1} = \theta^k - \alpha\tilde{\nabla} f(\theta^k)$
k = 75
Optimized SMPL model

Figure 4.22 The KNN optimization workflow

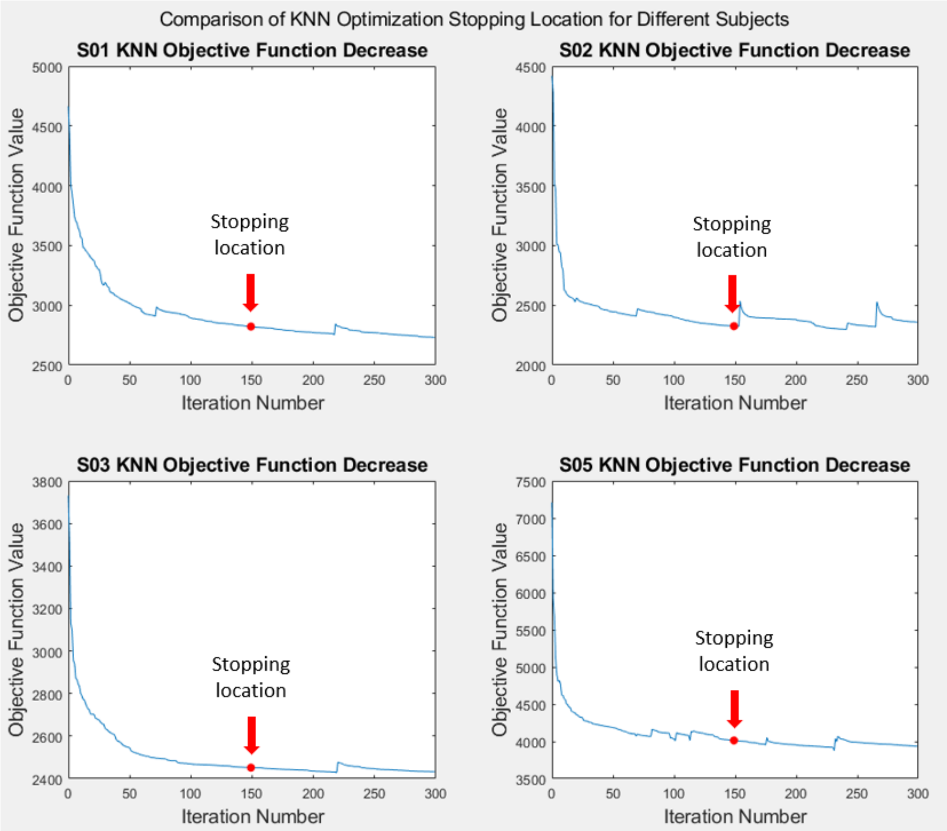


Figure 4.23 Comparison of KNN optimization stopping locations for different subjects

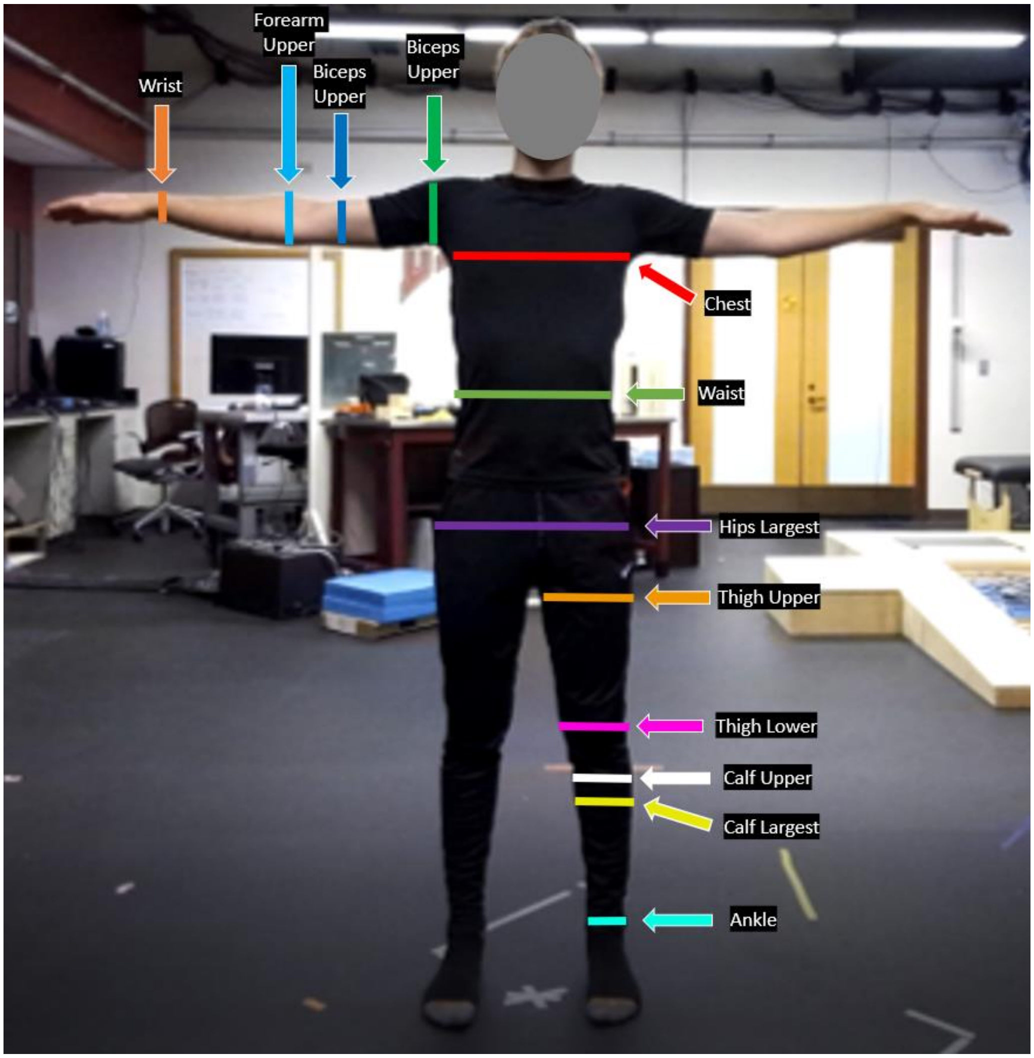


Figure 4.24 The circumference measurements taken of each subject

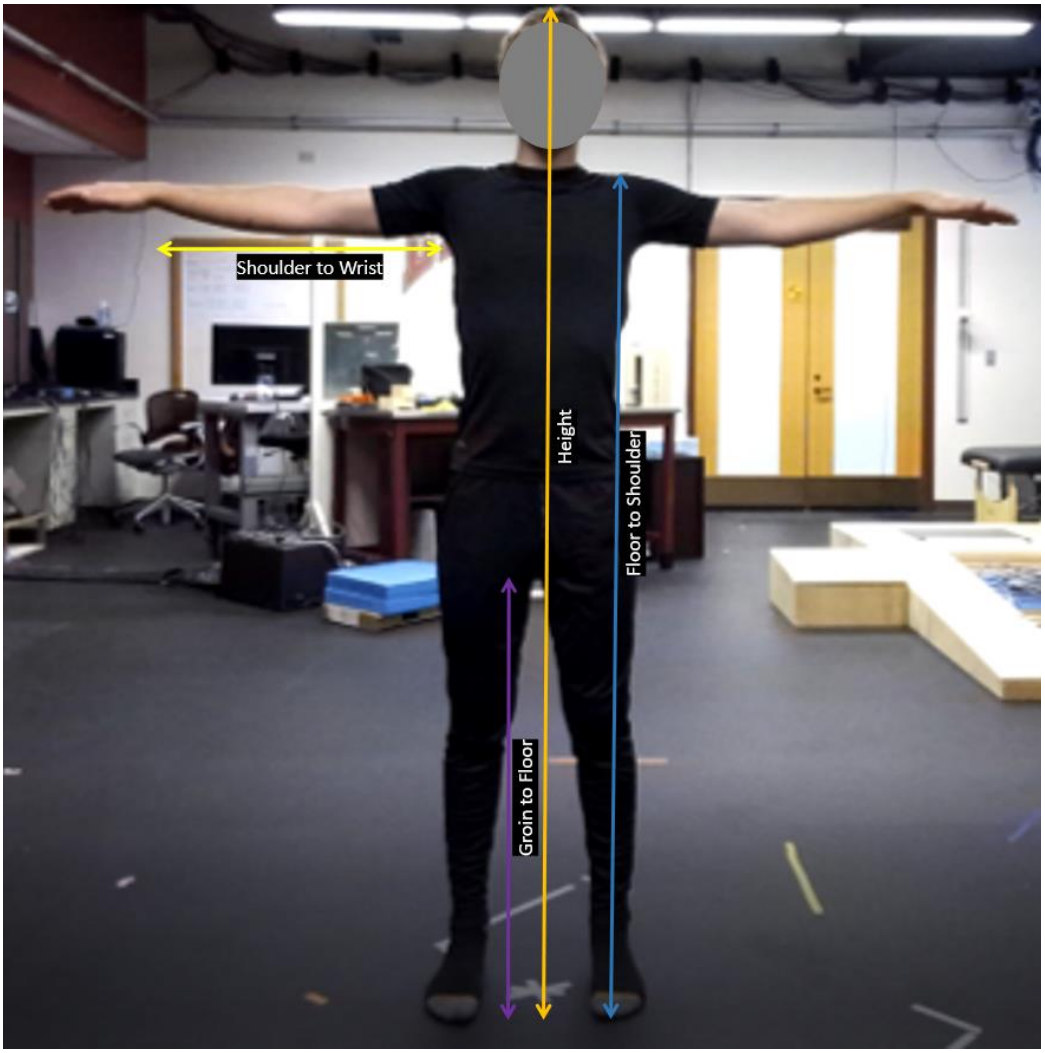


Figure 4.25 The length measurements taken of each subject

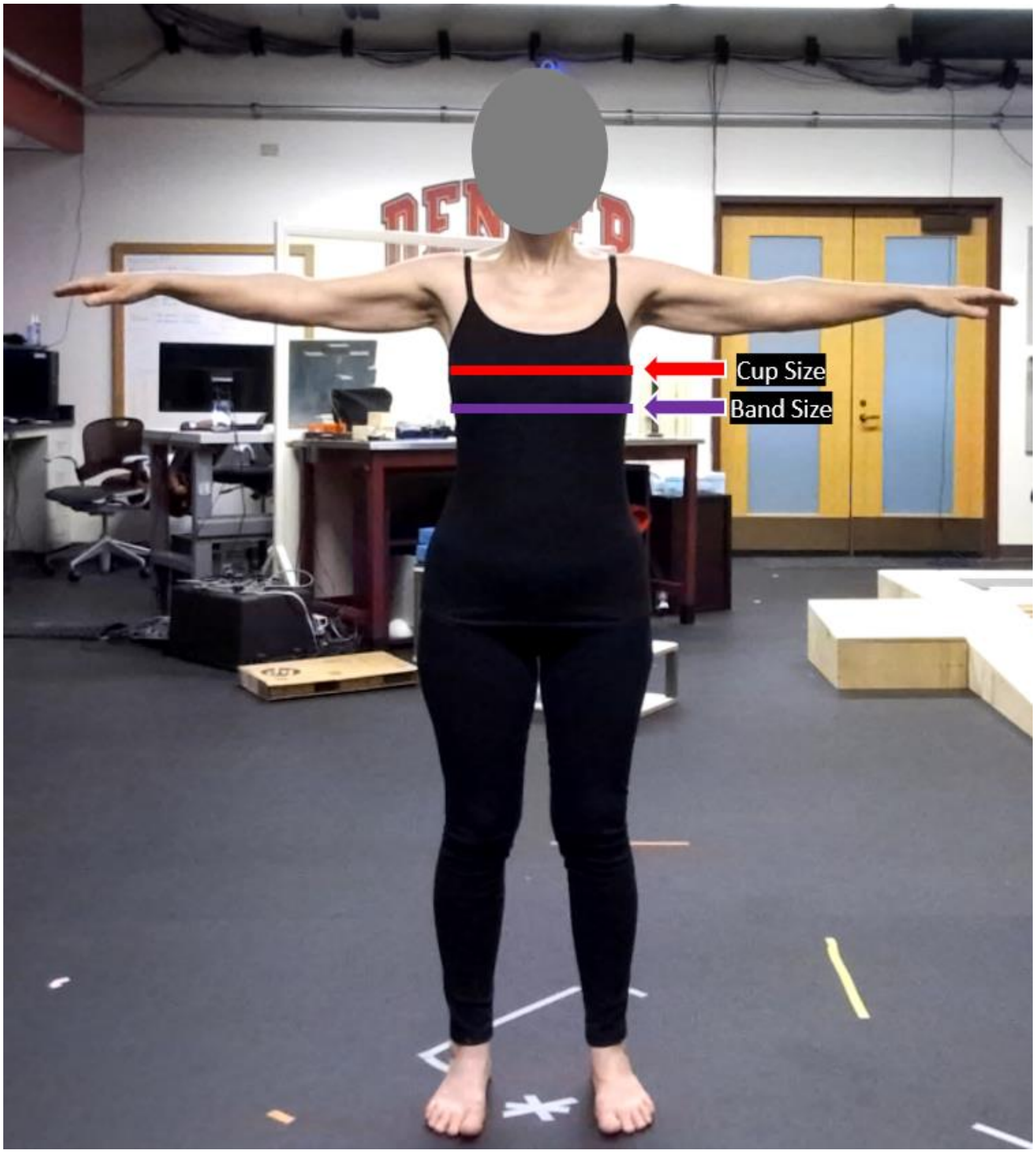


Figure 4.26 The alternative chest measurements taken of female subjects

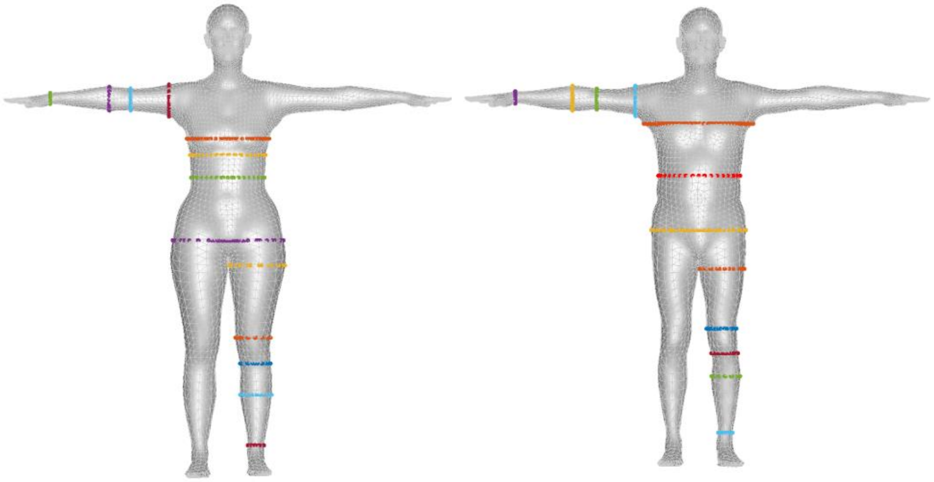


Figure 4.27 An example of accurate measurements measured from subject-specific SMPL models

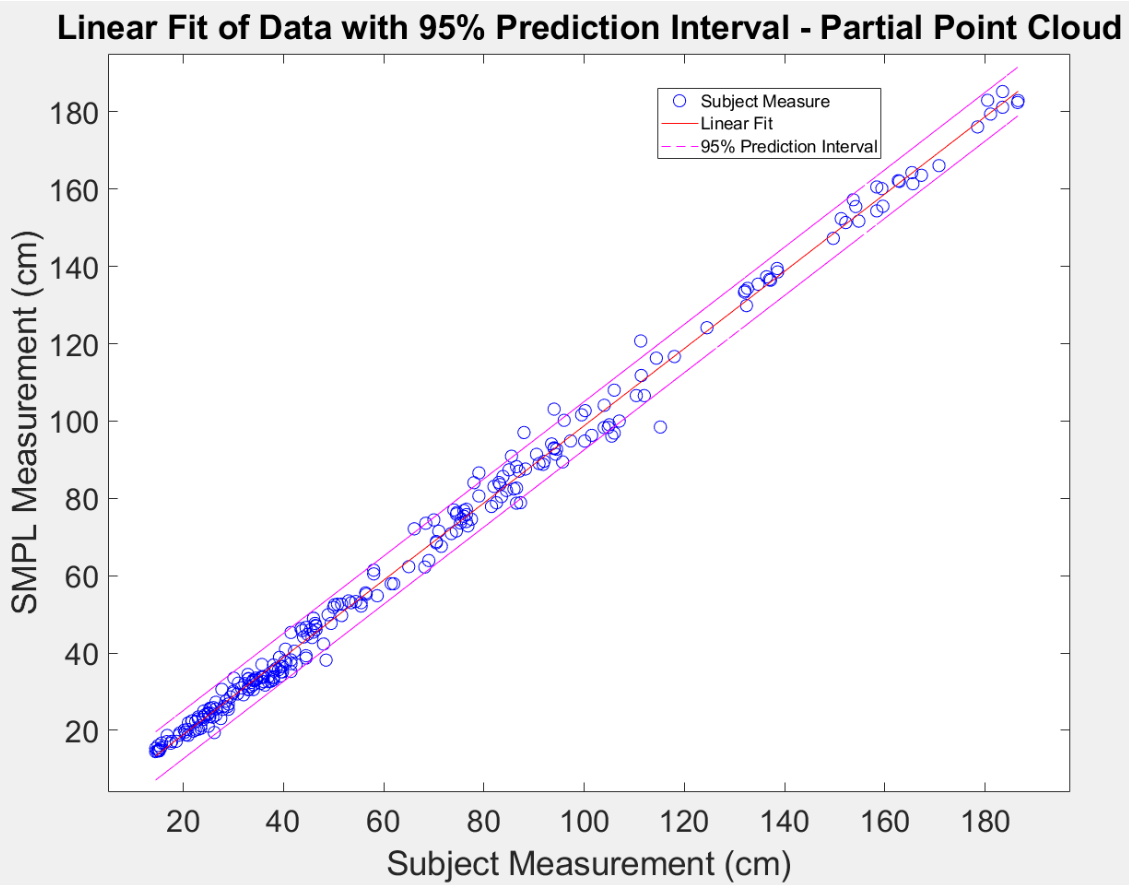


Figure 4.28 Model-predicted and manual anthropometric measures compared for all measures and subjects using partial point cloud data

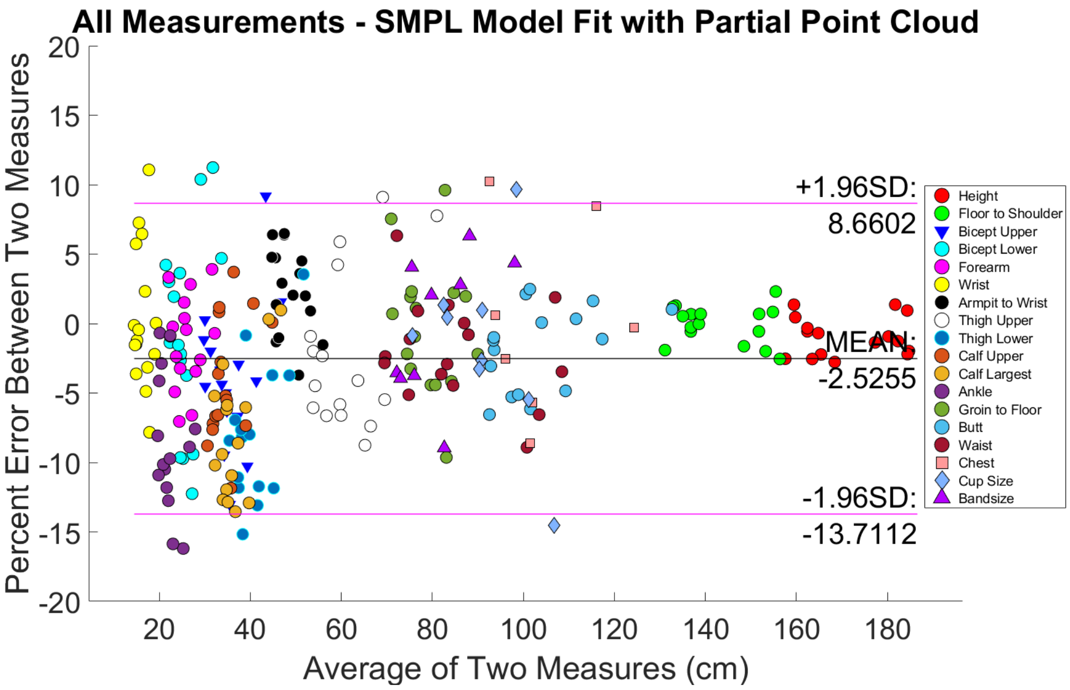


Figure 4.29 Bland-Altman diagram of all model-predicted and physical anthropometric measures

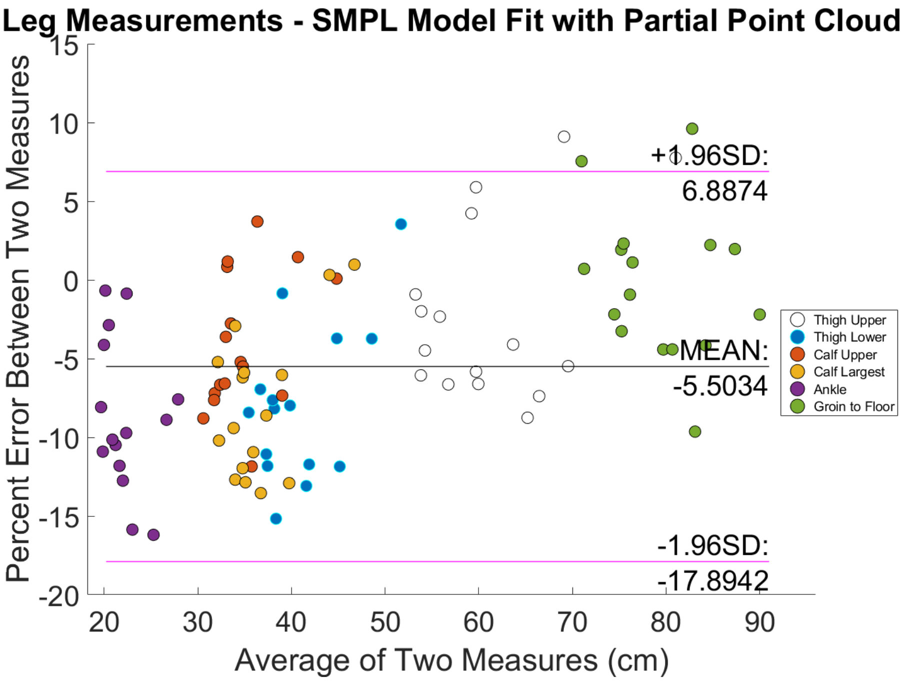


Figure 4.30 Bland-Altman diagram of leg measurements for model-predicted and physical anthropometric measures

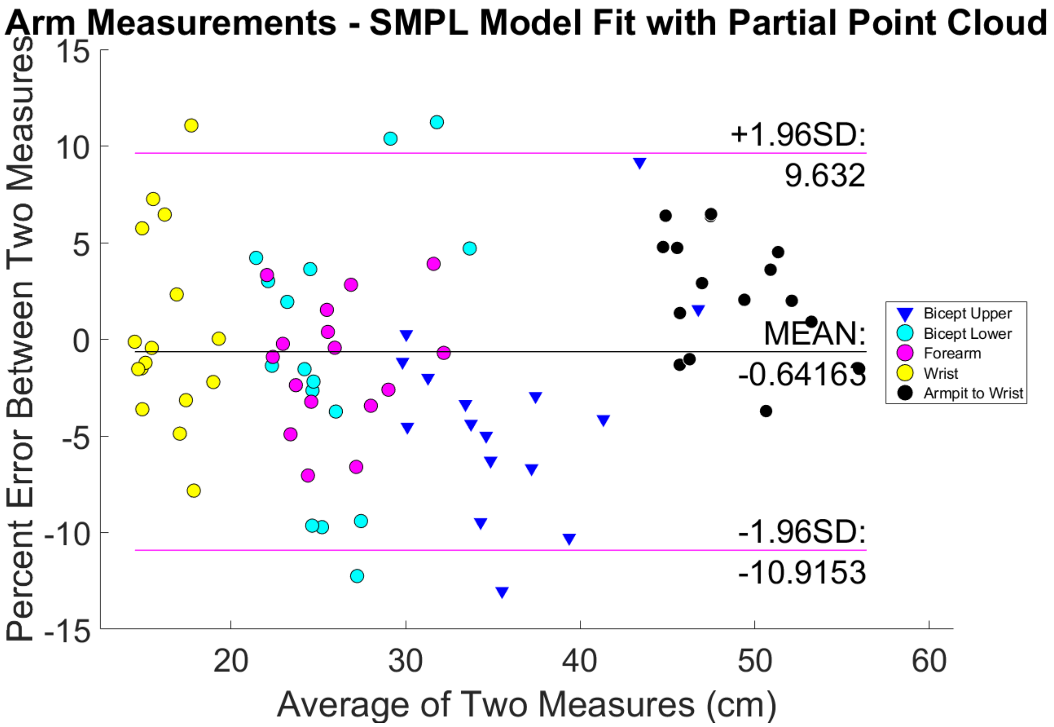


Figure 4.31 Bland-Altman diagram of arm measurements for model-predicted and physical anthropometric measures

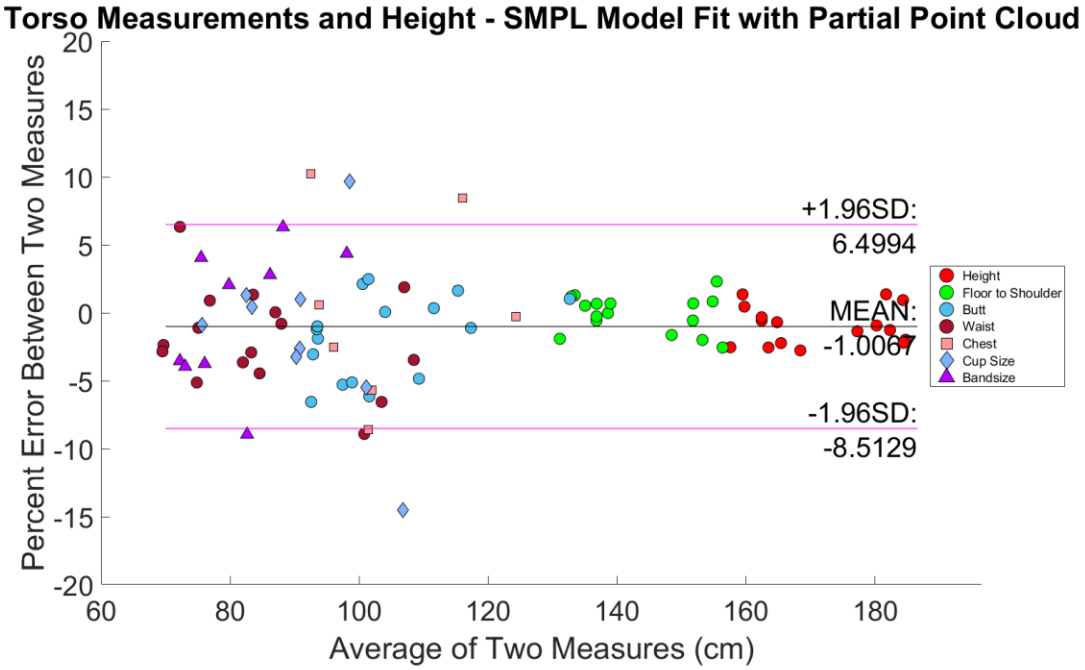


Figure 4.32 Bland-Altman diagram of torso and height measurements for model-predicted and physical anthropometric measures

Table 4.1 The average absolute percent error for every measure – partial point cloud

	Mean	SD
Floor To Shoulder	1.17	0.76
Height	1.47	0.80
Butt	2.74	2.15
Forearm	2.78	2.11
Waist	3.29	2.46
Armpit To Wrist	3.35	2.00
Groin To Floor	3.66	2.88
Wrist	3.71	3.20
Cup Size	4.35	4.81
Band Size	4.41	2.06
Calf Upper	5.03	3.28
Bicep Upper	5.26	4.09
Thigh Upper	5.47	3.65
Chest	5.61	2.35
Bicep Lower	5.73	3.93
Calf Largest	8.17	4.36
Thigh Lower	9.19	5.12
Ankle	9.80	6.33

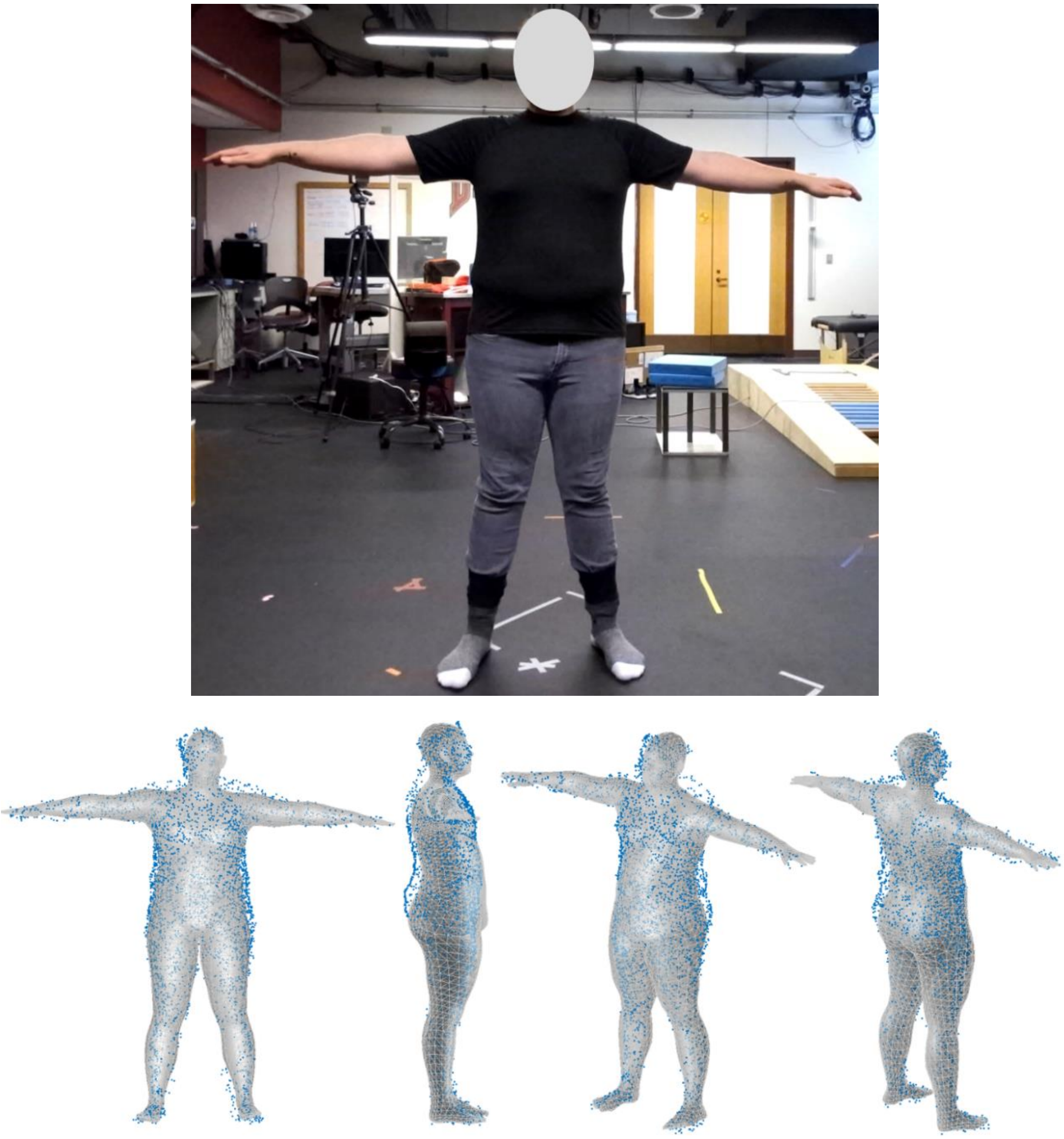


Figure 4.33 Exemplary subject S05 with one of the the lowest total absolute percent errors between SMPL and anthropometrc measures

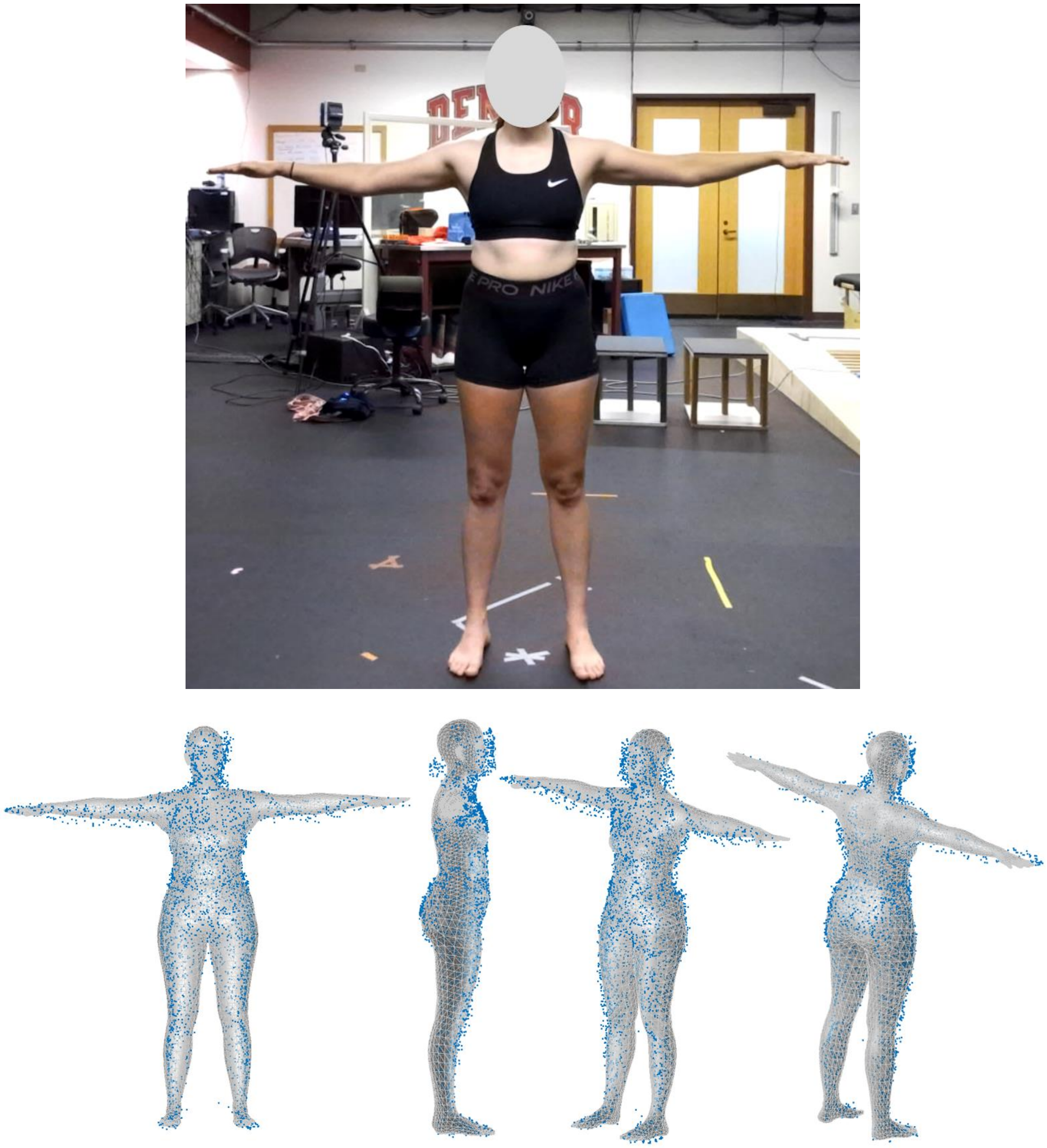


Figure 4.34 Exemplary subject S07 with one of the lowest total absolute percent errors between SMPL and anthropometrc measures

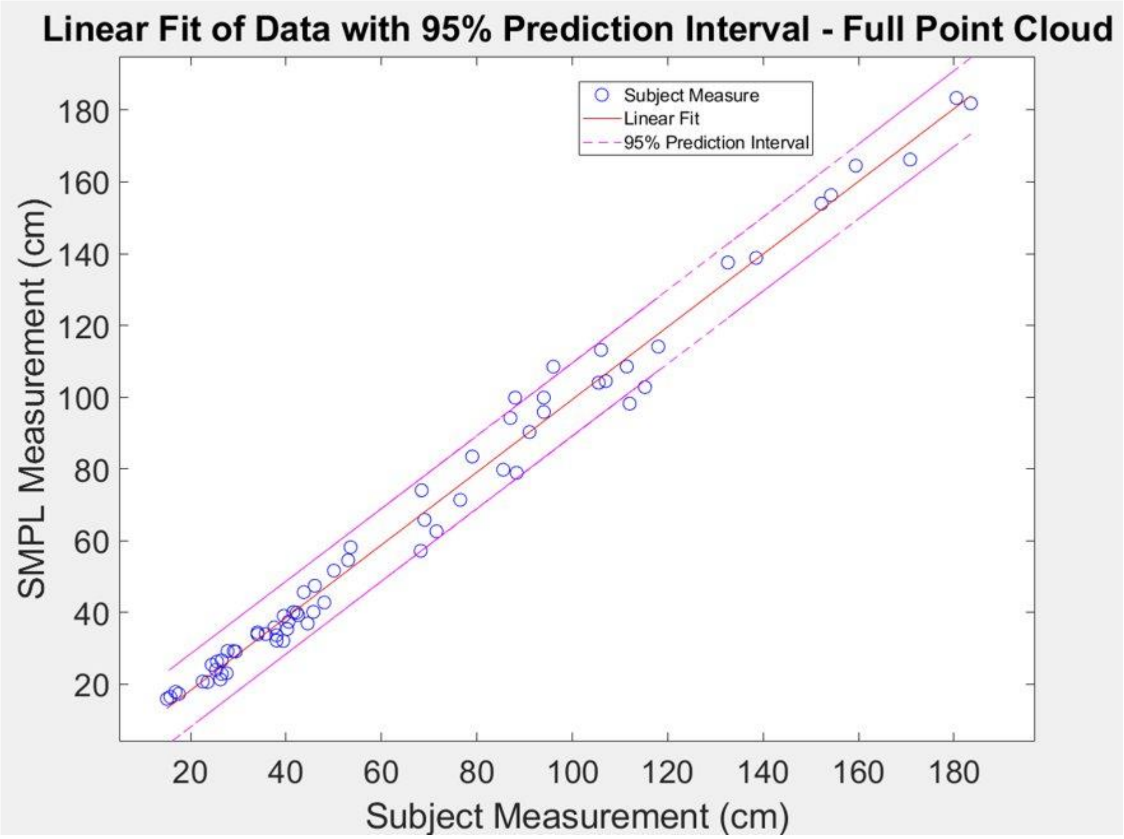


Figure 4.35 Model-predicted and manual anthropometric measures compared for all measures and subjects using full point cloud data

Table 4.2 The average KNN error between SMPL mesh and Kinect point cloud for models fit with partial and full point clouds

	Average KNN Error – Full (cm)	Average KNN Error – Partial (cm)
S09	0.0211	0.0470
S11	0.0232	0.0443
S22	0.0227	0.0401
S23	0.0214	0.0495

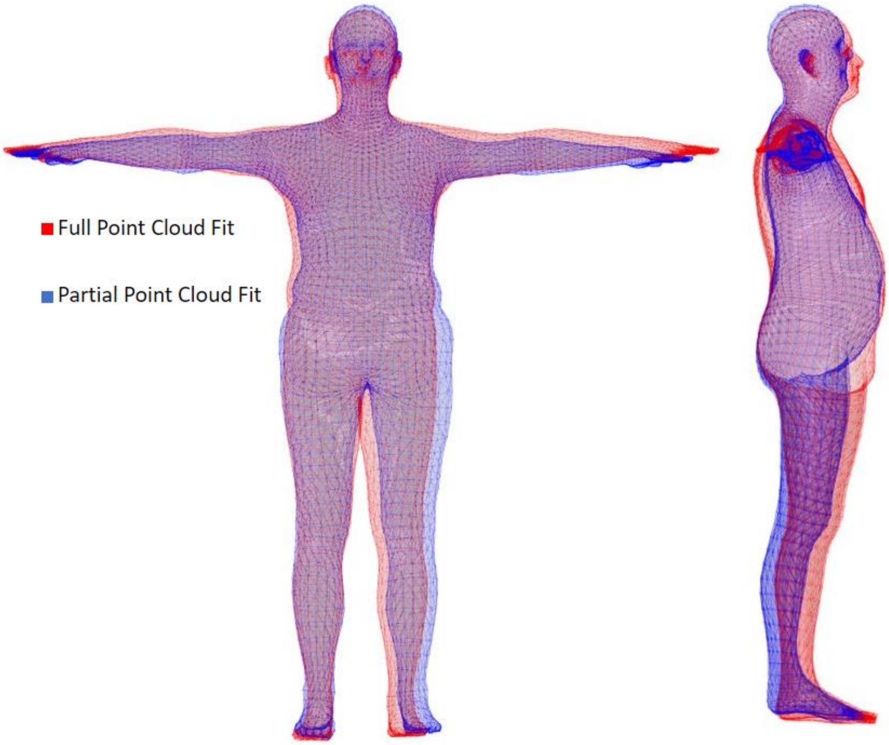


Figure 4.36 A comparison between S09's SMPL model created with the full and partial point cloud

Table 4.3 The comparison between the shape parameters found from fitting with the partial and full point cloud for S09

	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
Partial	0.46	1.45	3.40	-1.66	3.05	1.79	0.02	12.95	5.69	2.97
Full	-0.19	-0.34	1.96	-2.28	-0.15	1.62	-0.03	-1.15	0.35	2.09

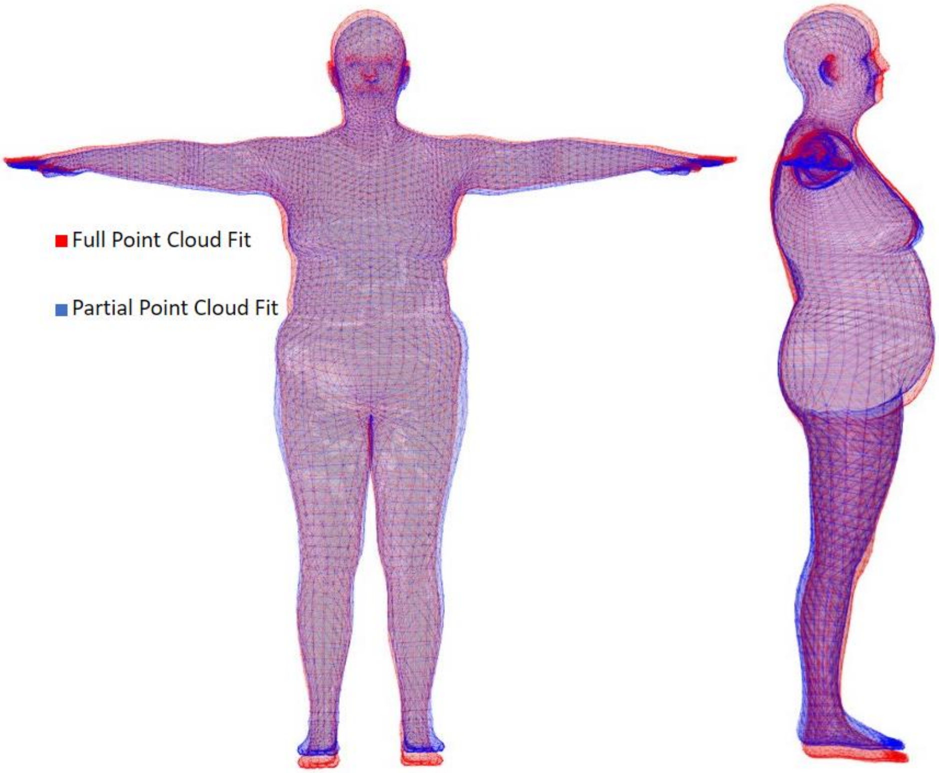


Figure 4.37 A comparison between S11's SMPL model created with the full and partial point cloud

Table 4.4 The comparison between the shape parameters found from fitting with the partial and full point cloud for S11

	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
Partial	-0.82	-1.56	2.19	1.72	-0.65	-2.46	-0.39	2.62	-1.87	1.06
Full	-0.21	-1.44	2.54	0.49	1.51	-1.53	1.87	-1.56	0.60	2.02

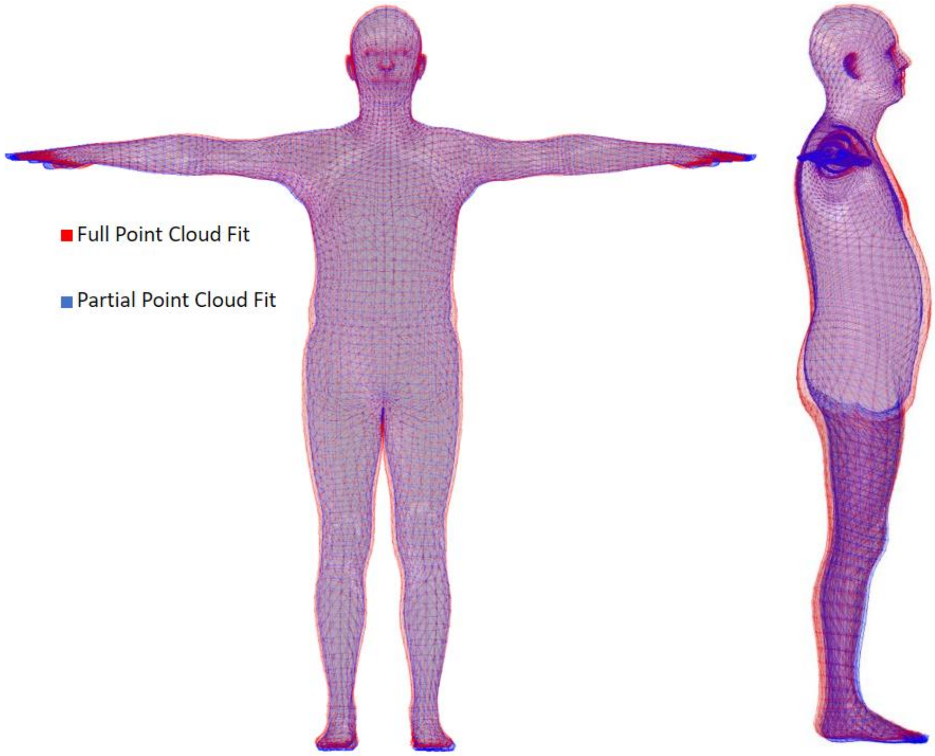


Figure 4.38 A comparison between S22's SMPL model created with the full and partial point cloud

Table 4.5 The comparison between the shape parameters found from fitting with the partial and full point cloud for S22

	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
Partial	1.61	-0.30	0.84	-0.79	-1.01	-2.60	-0.21	0.98	2.84	1.81
Full	1.80	-0.40	2.47	-0.17	0.41	-0.25	1.02	0.50	2.60	-0.34

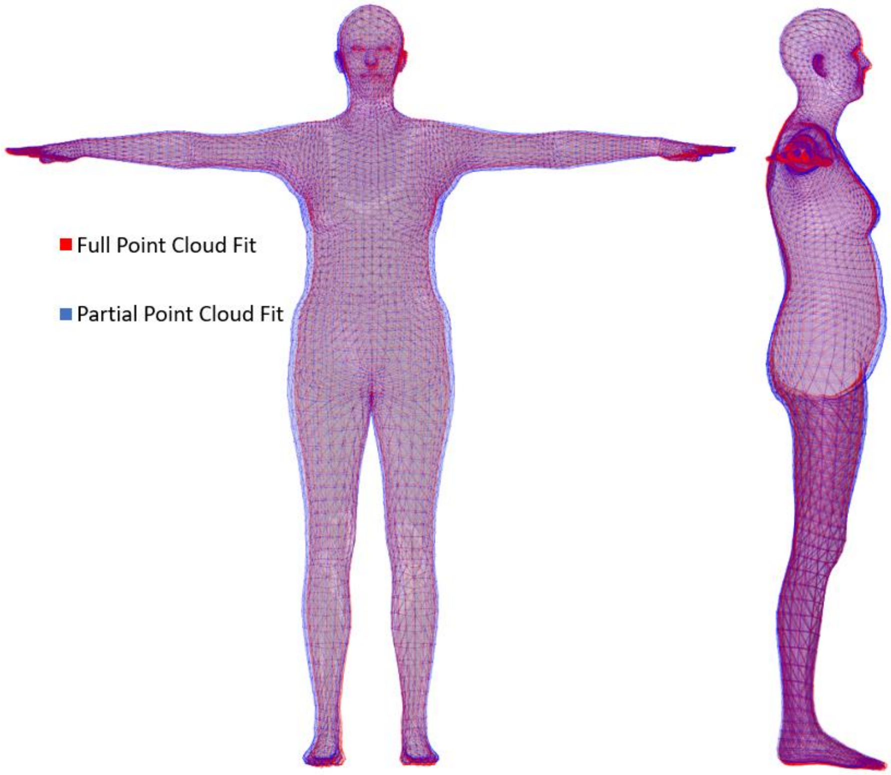


Figure 4.39 A comparison between S23’s SMPL model created with the full and partial point cloud

Table 4.6 The comparison between the shape parameters found from fitting with the partial and full point cloud for S23

	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
Partial	2.21	1.10	1.73	1.16	-0.09	-1.90	1.54	1.14	1.36	2.32
Full	2.11	1.73	1.65	1.71	1.38	-2.36	1.34	0.91	-2.12	1.09

Table 4.7 The average absolute percent error compared for only subjects collected with the partial and full method

Partial Point Cloud				**Full Point Cloud**		
	Mean	SD			Mean	SD
Floor to Shoulder	0.85	0.31		Floor to Shoulder	1.42	1.35
Height	1.46	0.96		Forearm	2.25	2.19
Butt	1.82	2.04		Height	2.27	1.10
Forearm	1.83	1.12		Armpit to Wrist	3.68	1.94
Armpit to Wrist	3.94	2.34		Bicep Lower	4.19	6.78

Waist	4.08	4.34		Wrist	5.21	2.20
Bicep Upper	5.18	3.43		Waist	5.79	4.05
Band Size	5.33	1.38		Groin to Floor	5.91	2.65
Thigh Upper	5.63	3.42		Bicep Upper	6.53	4.43
Groin to Floor	5.65	3.50		Butt	6.69	4.89
Bicep Lower	6.98	5.14		Cup Size	6.75	6.40
Calf Upper	7.37	3.37		Calf Upper	8.54	6.37
Wrist	7.41	2.63		Chest	9.16	4.21
Chest	9.42	1.17		Band Size	9.97	4.56
Thigh Lower	9.93	4.23		Thigh Upper	11.92	6.35
Calf Largest	11.34	3.55		Thigh Lower	12.13	6.81
Cup Size	12.10	3.46		Calf Largest	12.60	6.13
Ankle	13.16	10.54		Ankle	15.39	3.69

Table 4.8 The average distance between SMPL meshes fit with partial and full point cloud data

	S09	S11	S22	S23
Average distance (cm)	3.54	1.93	1.33	0.910

Table 4.9 Computational time of model fitting for S01 in seconds

Rigid alignment	Morphing	Initial Model Estimation	Pose Optimization	KNN Optimization	Final time
4	29	0.01	3	264	300

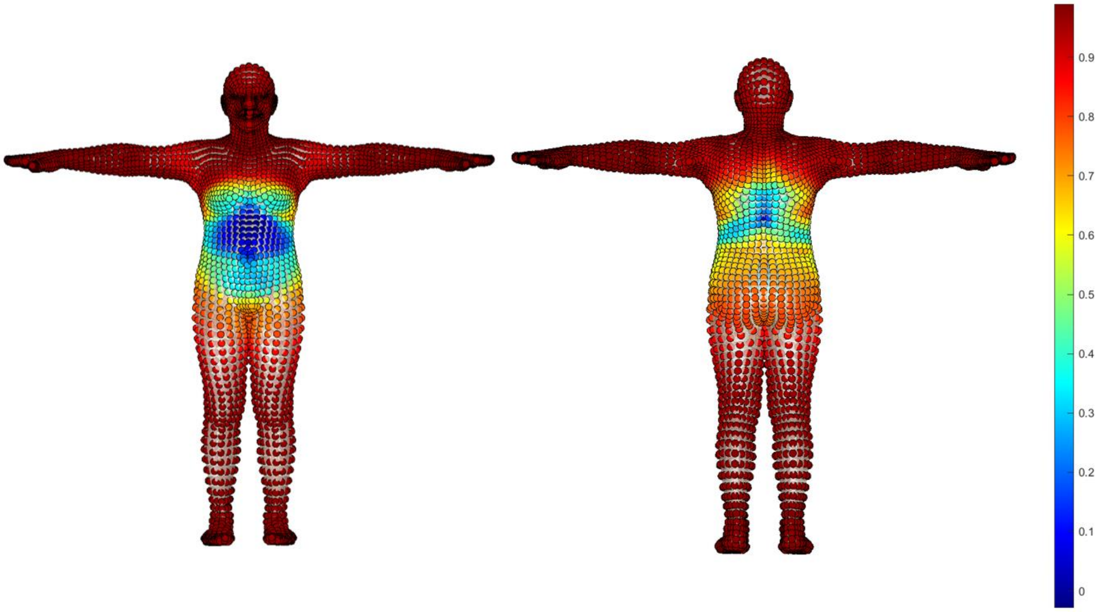


Figure 4.40 Correlation between 1st shape parameter and magnitude of node movement

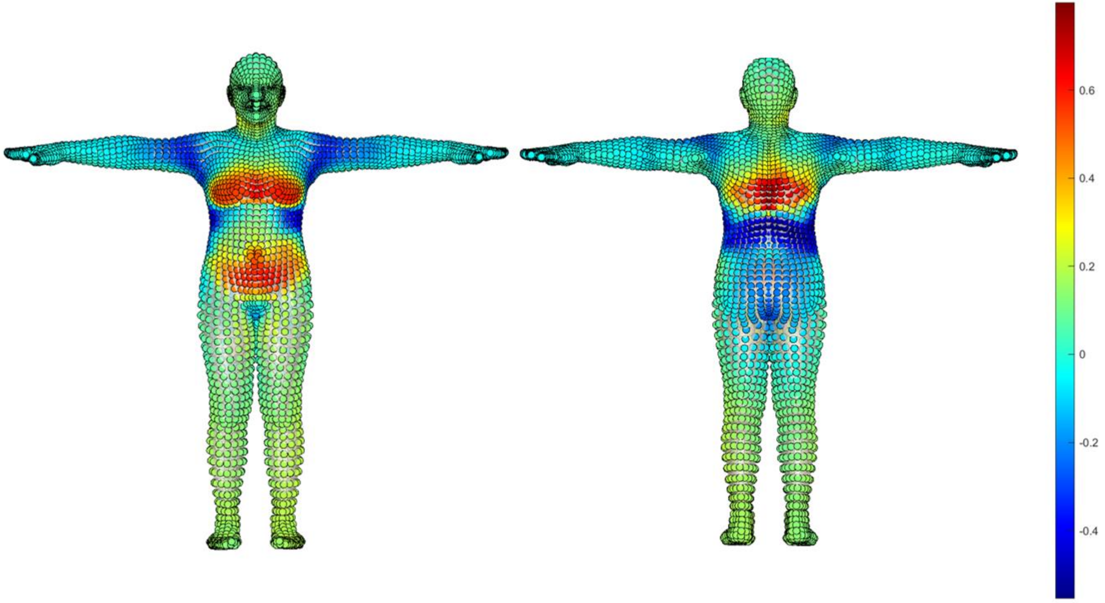


Figure 4.41 Correlation between 2nd shape parameter and magnitude of node movement

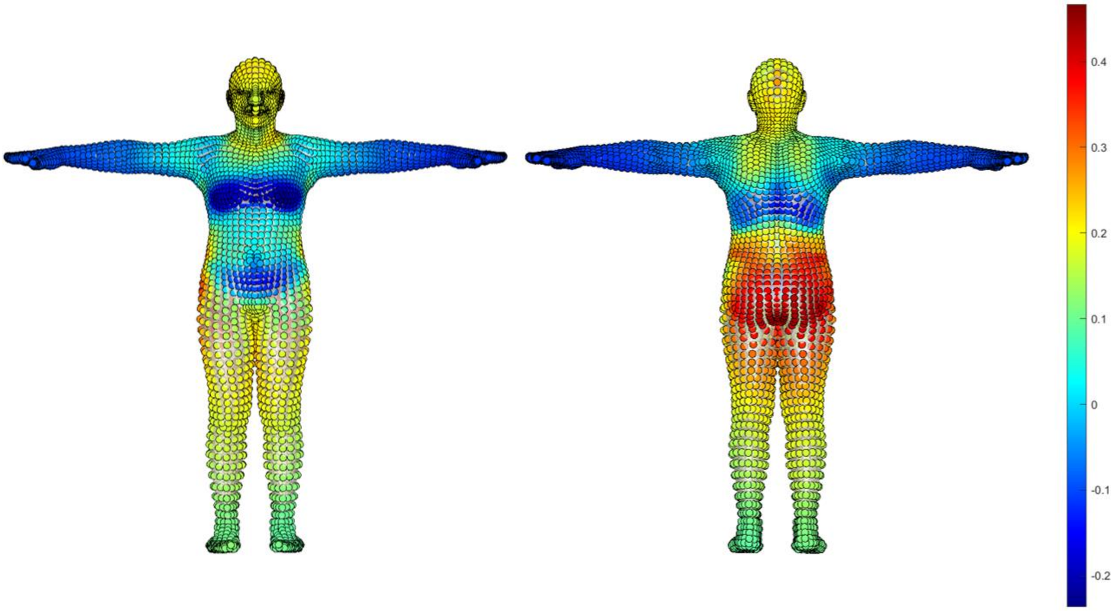


Figure 4.42 Correlation between 3rd shape parameter and magnitude of node movement

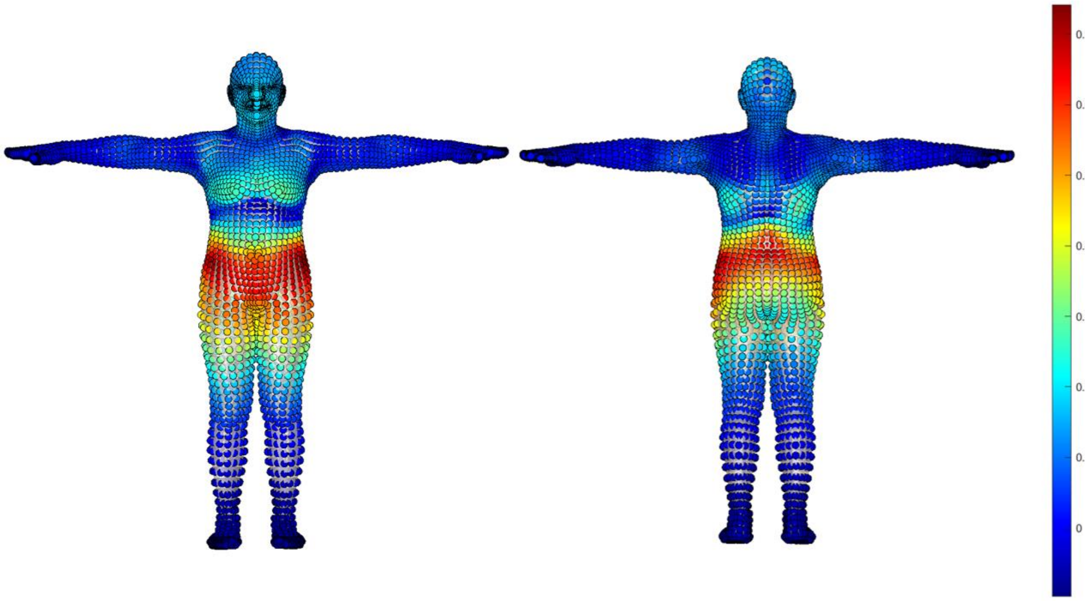


Figure 4.43 Correlation between 4th shape parameter and magnitude of node movement

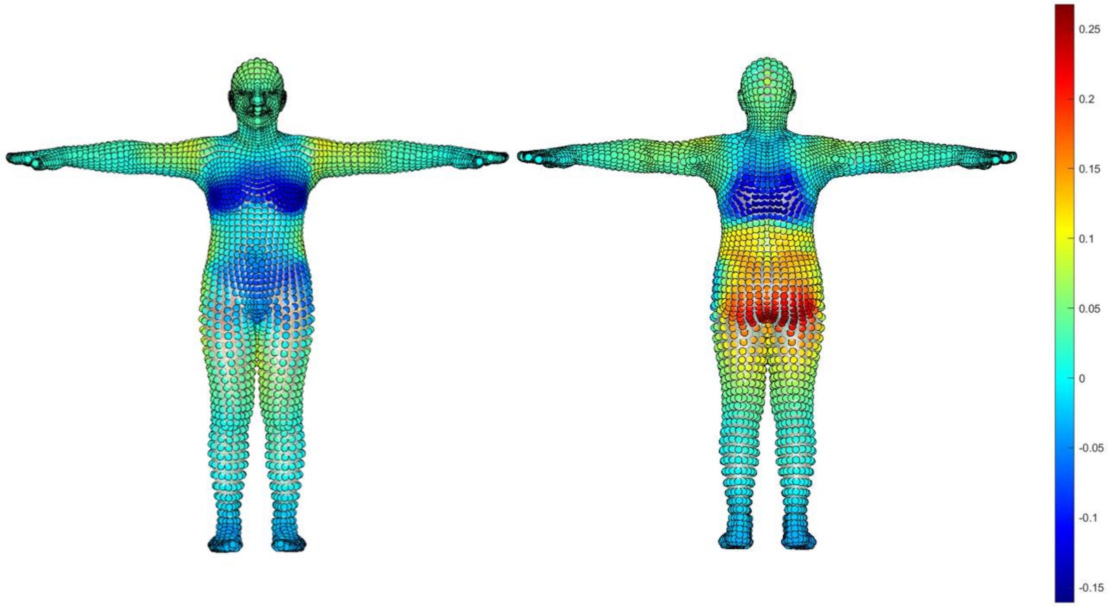


Figure 4.44 Correlation between 5th shape parameter and magnitude of node movement

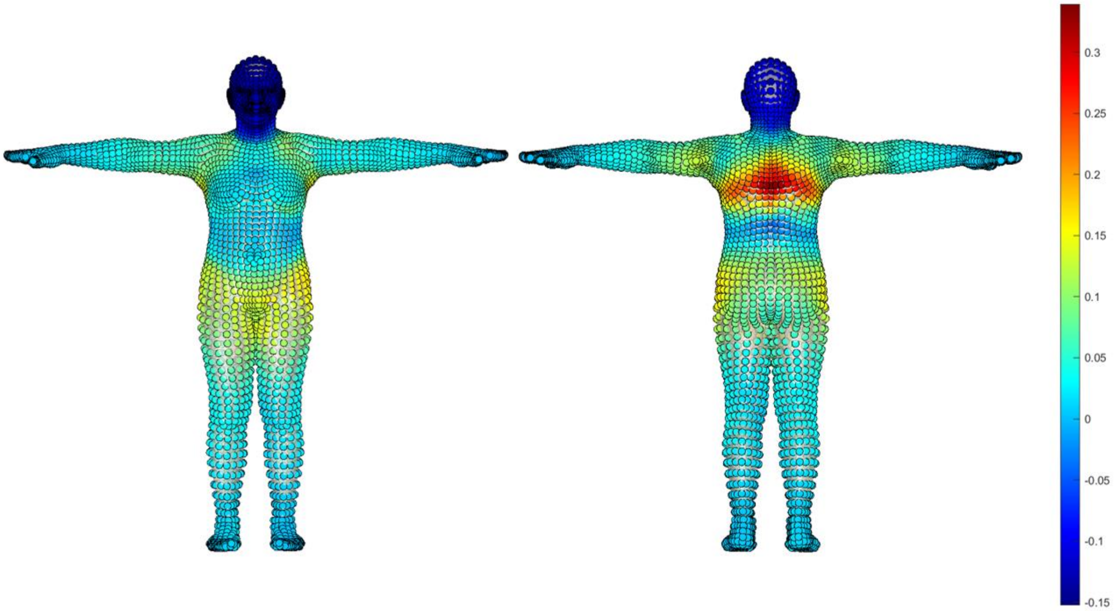


Figure 4.45 Correlation between 6th shape parameter and magnitude of node movement

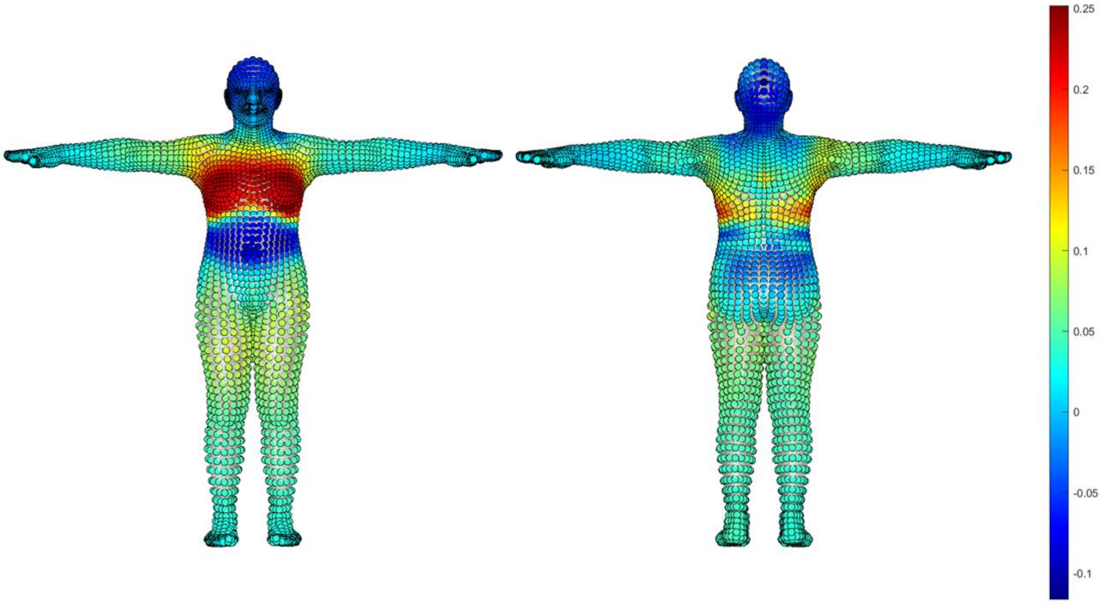


Figure 4.46 Correlation between 7th shape parameter and magnitude of node movement

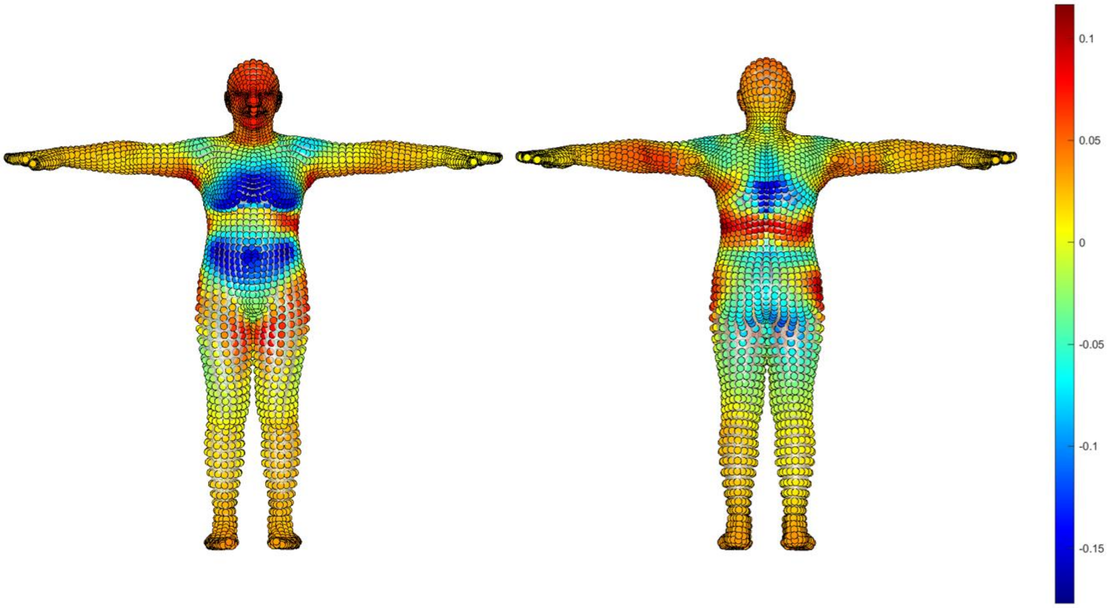


Figure 4.47 Correlation between 8th shape parameter and magnitude of node movement

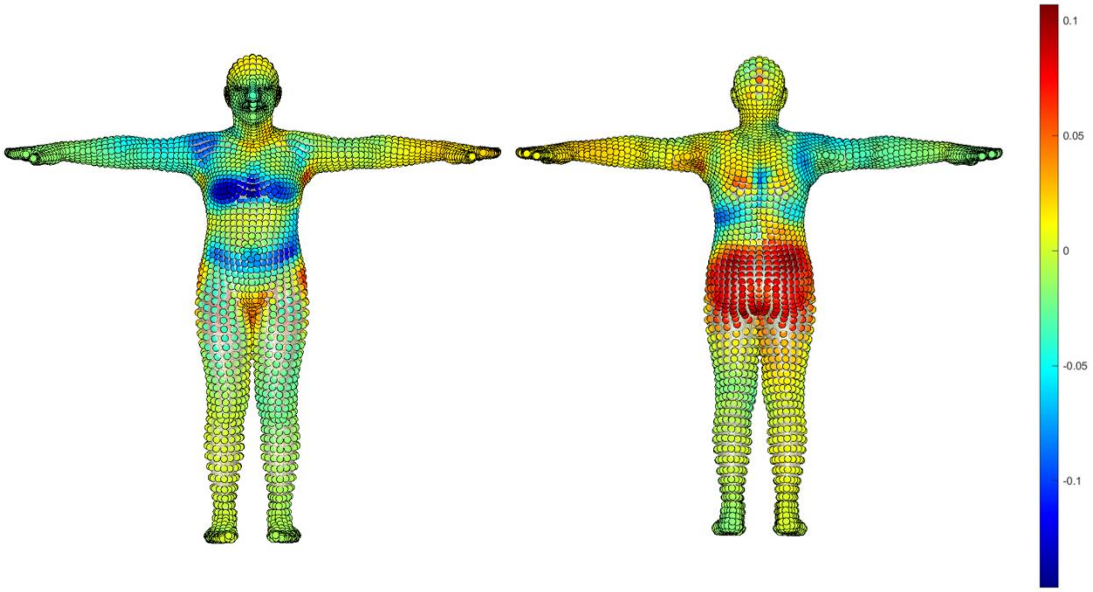


Figure 4.48 Correlation between 9th shape parameter and magnitude of node movement

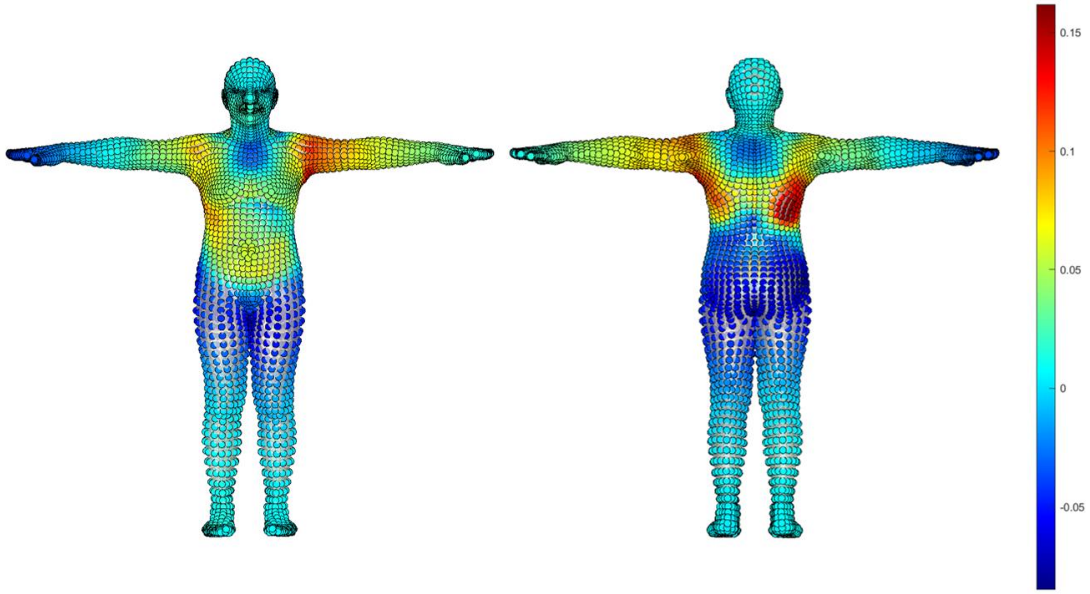


Figure 4.49 Correlation between 10th shape parameter and magnitude of node movement

CHAPTER 5: CONCLUSIONS AND RECOMMENDATIONS

This method was able to create accurate representations of human body shape. While there are variations in the accuracy of some areas of the model, the average absolute percent error between physical and SMPL model measurements is always under 10%. Across a broad variety of body types, this method can recreate surface body shape accurately. The optimization is robust, and able to perform effectively on a sparse point cloud of only 6,890 points. Every aspect of the alignment, initial SMPL model estimation, and optimization is automated by code and requires no human intervention. The data collection methods explored are straightforward and could be conducted and processed by someone with little expertise.

There is a large variety in the size of errors across body parts, and there does not seem to be a consistent region of the body where the optimization struggles. Code was created to automatically measure the SMPL models in the way the subjects were measured in real life, but there still exist variations to the way people were measured in the lab. Though care was taken to measure everyone the same, there are certain measurements that allow extra error to creep in, and it is difficult to recreate the same measurement technique on all subjects. The code to measure the SMPL model will have issue accurately measuring the body segments if the standard it is compared against changes slightly with each subject. For measurements like calf largest, the SMPL measurement was taken by finding the largest part of the calf by looking from a side profile view. This view was hard to obtain and scrutinize when taking physical measurements because it is difficult to crouch on the ground and see it. This could also be another reason why the ankle measurement is not as accurate. The lower thigh measurement was collected at a location ~5in from the center of the knee and the placement for the tape was not as standardized as it was for other measurements. This made it difficult to recreate the location of the measurement in the SMPL model. The larger differences between the SMPL and physical measurements are driven largely by an inability to measure the SMPL model in the exact location the subject was measured in the lab.

The optimization process could be improved by investigating global optimization methods because the objective function is not convex. These methods may lead to longer computational times and may render the initial SMPL model estimation useless. The workflow to process one subject is already quick, global optimization could be explored but not pursued if the computational time is not comparable to this method that already produces accurate body models. Future work should focus on refining the optimization further. It would be useful to pick more appropriate weights for the torso and extremities. The current weights work well enough, but more effort should be spent in finding optimal numbers. Currently, the extremity weight must be included in the objective function to keep the arms and legs from shrinking as the optimization focuses too heavily on the weight of the torso. This sometimes leaves the arms and legs unnaturally stretched. Additionally, the thighs could be weighted in a similar way to the torso to improve the accuracy of the leg measurements. Adjusting weights or adding a more complex weighting scheme for different areas of the body would likely not add any computational time and only improve the model fitting results further.

Additional improvements could be made by exploring ways of imposing constraints on the shape parameters. Sometimes the optimization can “run-away” and start pursuing unrealistic parameters. To prevent this, the optimization sets any shape parameters larger than 3 to 3, and any smaller than -3 to -3. This works well for individuals that have “regular” features, but performance suffers for people who have outlier features, for example, being tall or overweight. If the regular optimization is not working well on certain individuals, the constraints are lifted, and the performance usually improves. The optimization should be tailored to release these constraints when it is appropriate, or perhaps the constraints are not as crucial as originally believed. This is a weakness of the algorithm because it is at the discretion of the experimenter when to make the optimization unconstrained, and the results of both need to be known to make a decision about which results to keep. There are some features someone can possess that are undeniably outliers. It is easy to recognize the optimization should be switched to unconstrained for these cases. There are shape parameters of the SMPL model that describe these variations, but people can also be outliers in other combinations of shape parameters that may not be immediately apparent, and it would be difficult to know whether to unconstrain the optimization.

A clear stopping criterion should be defined that can discern between a correctly fit model and an unconverged model. Each subject's point cloud is down sampled to the same number, and the male, female, and neutral SMPL mesh all have the same number of nodes. The objective function is a summation of the same number of KNN distances for each subject, so there exists some number the function can reach where the mesh and point cloud are close enough to each other, and the algorithm can be considered converged.

It would be advantageous to verify the accuracy of this body scan method in a setting outside the HDL, in a more clinical or casual setting. The point cloud data collection and processing procedures explored are not complicated and could be carried out easily by a layperson. The process created to fit the SMPL model to the Kinect point cloud is also completely automated. Some subjects were collected wearing skin-tight spandex attire, while others were wearing tighter street clothes like skinny jeans. It is not clear if the selection of clothing impacted the results of the model fit because there is a mix of people wearing street clothes and spandex in the final 16 individuals that data analysis was performed on. The subject that created the best SMPL model was wearing skinny jeans and a tight T-shirt, and the second-best model was created by a person wearing a sports bra and spandex shorts. The model fitting process appears to work fine regardless of what type of form fitting clothing the person wears. However, the scan taken of the person may not capture the person's true surface geometry. For example, if the person is wearing extremely tight skinny jeans and a shirt that is too big, this will create a surface scan of the person where they are stuffed into pants and not accurately capture their torso region. At its current accuracy level, this method can work on people wearing street clothes if these clothes fit appropriately and do not significantly alter the surface geometry of the person.

Future work with this model fitting process could involve exploring the relationship between shape parameters, the distribution of weight, and how this affects movement patterns. It could be possible to identify a parameter, or combination of parameters, that indicates a subject is more likely to be at risk of unusual movement patterns that could cause problematic wear in implants.

This is only the beginning for the use of the SMPL model in biomechanics studies at DU. This thesis allows for the creation of high-quality computational representations. The next step is to animate the body model to motion-capture data. This optimization process finds the principal component scores that describe the shape of the SMPL model, and these will not change during a dynamic trial. Only the pose parameters would need to be optimized to animate a model in a dynamic activity, which is a much easier problem than this one. Significant dynamic data was obtained of each subject performing a variety of activities with the partial point cloud collection method. This data includes three trials of each of the following: lunge, sit to stand from a high chair, sit to stand from a low chair, self-selected walking, and fast walking. The transformation matrices to align the two point clouds were already found for aligning the views for each T-pose, so post-processing would just involve cropping the point cloud to the activity. The shape parameters of 16 subjects are already known, so the pose parameters for each frame of motion capture would need to be found to animate their SMPL model to the capture. Code to optimize the SMPL pose already exists because it was used in the cyclic KNN optimization. A person would just need to take the prewritten steepest descent function, pass in a frame of the point cloud capture and let the optimization converge, then repeat with the next frame using the previous pose as a starting location for the optimization. This would be a good starting point for animating the model, but it is not clear yet if this method would create a smooth representation of the activity. If it does not, a person could take a select few frames, optimize the pose in these, and then interpolate the joint locations between the frames. With the data collected from this study, it could be difficult to quantify the accuracy of the SMPL body's movements. A new study could be designed to capture dynamic activity while a subject wears some form of markered motion capture. Stride length, walking speed, and knee flexion angle calculated from SMPL joint centers could all be compared against Vicon data. The accuracy of the position of certain locations on the SMPL mesh through an activity could be compared to strategically placed markers in similar locations on the subject.

BIBLIOGRAPHY

[1] S. Harms *et al.*, "Obesity increases the likelihood of total joint replacement surgery among younger adults," *Int Orthop*, vol. 31, no. 1, pp. 23–26, Feb. 2007, doi: 10.1007/s00264-006-0130-y.

[2] M. L. Wolford *et al.*, "Hospitalization for total hip replacement among inpatients aged 45 and over: United States, 2000-2010.," *NCHS Data Brief*, no. 186, pp. 1–8, Feb. 2015.

[3] P. Capodaglio *et al.*, "Effect of obesity on knee and ankle biomechanics during walking," *Sensors (Basel)*, vol. 21, no. 21, Oct. 2021.

[4] A. Eustace, "Development of a Clinical Marker-less Motion Capture System for Patient Monitoring," University of Denver, Denver, 2020.

[5] M. Loper *et al.*, "SMPL," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 1–16, Nov. 2015.

[6] N. Mahmood *et al.*, "AMASS: Archive of motion capture as surface shapes," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, Oct. 2019.

[7] Z. F. Lerner *et al.*, "Effects of obesity on lower extremity muscle function during walking at two speeds," *Gait Posture*, vol. 39, no. 3, pp. 978–984, Mar. 2014.

[8] S. P. Messier *et al.*, "Influences of alignment and obesity on knee joint loading in osteoarthritic gait," *Osteoarthritis Cartilage*, vol. 22, no. 7, pp. 912–917, Jul. 2014.

[9] "Azure Kinect DK hardware specifications," *Microsoft*, Sep. 02, 2022.

[10] R. Whyte *et al.*, "Application of lidar techniques to time-of-flight range imaging," *Appl. Opt.*, vol. 54, no. 33, pp. 9654–9664, Nov. 2015.

[11] A. Takmaz *et al.*, "3D segmentation of humans in point clouds with synthetic data," Dec. 2022.

[12] M. Tölgyessy *et al.*, "Skeleton tracking accuracy and precision evaluation of Kinect V1, Kinect V2, and the Azure Kinect," *Appl. Sci. (Basel)*, vol. 11, no. 12, p. 5756, Jun. 2021.

[13] C. Ferraris *et al.*, “Evaluation of arm swing features and asymmetry during gait in Parkinson’s disease using the azure Kinect sensor,” *Sensors (Basel),* vol. 22, no. 16, p. 6282, Aug. 2022.

[14] G. Amprimo *et al.*, “Assessment tasks and virtual exergames for remote monitoring of Parkinson’s disease: An integrated approach based on Azure Kinect,” *Sensors (Basel),* vol. 22, no. 21, p. 8173, Oct. 2022.

[15] F. L. Siena *et al.*, “Utilising the Intel RealSense camera for measuring health outcomes in clinical research,” *J. Med. Syst.*, vol. 42, no. 3, p. 53, Feb. 2018.

[16] J. Purnama *et al.*, “The Intel realsense depth-camera performance for real-time customer satisfaction analysis using facial expression detection,” *J. Phys. Conf. Ser.*, vol. 1175, p. 12076, Mar. 2019.

[17] Z. Cao *et al.*, “OpenPose: Realtime multi-person 2D pose estimation using Part Affinity Fields,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 172–186, Jan. 2021.

[18] Martín~Abadi *et al.*, “ TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems.” 2015. [Online]. Available: https://www.tensorflow.org/

[19] A. Mathis *et al.*, “DeepLabCut: markerless pose estimation of user-defined body parts with deep learning,” *Nat. Neurosci.*, vol. 21, no. 9, pp. 1281–1289, Sep. 2018.

[20] E. P. Washabaugh *et al.*, “Comparing the accuracy of open-source pose estimation methods for measuring gait kinematics,” *Gait Posture*, vol. 97, pp. 188–195, Sep. 2022.

[21] J. Stenum *et al.*, “Two-dimensional video-based analysis of human gait using pose estimation,” *PLoS Comput. Biol.*, vol. 17, no. 4, p. e1008935, Apr. 2021.

[22] P. Venkataraman, *Applied optimization with MATLAB programming*. Nashville, TN: John Wiley & Sons, 2002.

[23] R.-G. P. *et al.*, “Point cloud optimization based on 3D geometric features for architectural heritage modelling,” 2021.

[24] J. Mei *et al.*, “3D tree modeling from incomplete point clouds via optimization and L1-MST,” *Geogr. Inf. Syst.*, vol. 31, no. 5, pp. 999–1021, May 2017.

[25] P. J. Besl *et al.*, “A method for registration of 3-D shapes,” *IEEE Trans Pattern Anal Mach Intell*, vol. 14, no. 2, pp. 239–256, Feb. 1992, doi: 10.1109/34.121791.

[26] A. Makovetskii *et al.*, “Affine registration of point clouds based on point-to-plane approach,” *Procedia Eng.*, vol. 201, pp. 322–330, 2017.

[27] M. Lamine Tazir *et al.*, “CICP: Cluster Iterative Closest Point for sparse–dense point cloud registration,” *Rob. Auton. Syst.*, vol. 108, pp. 66–86, Oct. 2018.

[28] A. Myronenko *et al.*, “Point set registration: coherent point drift,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 12, pp. 2262–2275, Dec. 2010.

[29] A. L. Yuille *et al.*, “A mathematical analysis of the motion coherence theory,” *Int. J. Comput. Vis.*, vol. 3, no. 2, pp. 155–175, Jun. 1989.

[30] O. Hirose, “A Bayesian formulation of coherent point drift,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 7, pp. 2269–2286, Jul. 2021.

[31] I. Sintini, “ Statistical Shape and Intensity Modeling of the Shoulder,” *ProQuest Dissertations Publishing*, 2017.

[32] W. S. Burton 2nd *et al.*, “Assessment of scapular morphology and bone quality with statistical models,” *Comput. Methods Biomech. Biomed. Engin.*, vol. 22, no. 4, pp. 341–351, Mar. 2019.

[33] X. Du *et al.*, “A radial-basis function mesh morphing and Bayesian optimization framework for vehicle crashworthiness design,” *Struct. Multidiscipl. Optim.*, vol. 66, no. 3, Mar. 2023.

[34] F. Xu *et al.*, “Numerical simulations of flow patterns in the human left ventricle model with a novel dynamic mesh morphing approach based on radial basis function,” *Comput. Biol. Med.*, vol. 130, no. 104184, p. 104184, Mar. 2021.

[35] F. Ghezelbash *et al.*, “Obesity and obesity shape markedly influence spine biomechanics: A subject-specific risk assessment model,” *Ann. Biomed. Eng.*, vol. 45, no. 10, pp. 2373–2382, Oct. 2017.

[36] L. M. Bollinger, “Potential contributions of skeletal muscle contractile dysfunction to altered biomechanics in obesity,” *Gait Posture*, vol. 56, pp. 100–107, Jul. 2017.

[37] K. L. Paterson *et al.*, “The influence of sex and obesity on gait biomechanics in people with severe knee osteoarthritis scheduled for arthroplasty,” *Clin. Biomech. (Bristol, Avon)*, vol. 49, pp. 72–77, Nov. 2017.

[38] P. Molina-Garcia *et al.*, “Effects of integrative neuromuscular training on the gait biomechanics of children with overweight and obesity,” *Scand. J. Med. Sci. Sports*, vol. 32, no. 7, pp. 1119–1130, Jul. 2022.

[39] S. Bertoli *et al.*, “Association of Body Shape Index (ABSI) with cardio-metabolic risk factors: A cross-sectional study of 6081 Caucasian adults,” *PLoS One*, vol. 12, no. 9, p. e0185013, Sep. 2017.

[40] M. Ji *et al.*, “Effectiveness of A Body Shape Index (ABSI) in predicting chronic diseases and mortality: a systematic review and meta-analysis,” *Obes. Rev.*, vol. 19, no. 5, pp. 737–759, May 2018.

[41] M. Yu *et al.*, “Body shape classification of Korean middle-aged women using 3D anthropometry,” *Fashion Text.*, vol. 7, no. 1, Dec. 2020.

[42] T. Naveed *et al.*, “Female body shape classifications and their significant impact on fabric utilization,” *Fiber. Polym.*, vol. 19, no. 12, pp. 2642–2656, Dec. 2018.

[43] K. V Vo *et al.*,, “Classifications in brief: Walch classification of primary glenohumeral osteoarthritis,” *Clin. Orthop. Relat. Res.*, vol. 475, no. 9, pp. 2335–2340, Sep. 2017.

[44] P. Domos *et al.*, “Walch B0 glenoid: pre-osteoarthritic posterior subluxation of the humeral head,” *J. Shoulder Elbow Surg.*, vol. 27, no. 1, pp. 181–188, Jan. 2018.

[45] N. Eyal, “Denial of treatment to obese patients-the wrong policy on personal responsibility for health,” *Int. J. Health Policy Manag.*, vol. 1, no. 2, pp. 107–110, Aug. 2013.

[46] S. M. Phelan *et al.*, “Impact of weight bias and stigma on quality of care and outcomes for patients with obesity,” *Obes. Rev.*, vol. 16, no. 4, pp. 319–326, Apr. 2015.

[47] T. E. Andreassen *et al.*, “An automated process for 2D and 3D finite element overclosure and gap adjustment using radial basis function networks,” Sep. 2022.

[48] J. Zhang *et al.*, “Point-cloud registration using adaptive radial basis functions,” *Comput Methods Biomech Biomed Engin*, vol. 21, no. 7, pp. 498–502, May 2018, doi: 10.1080/10255842.2018.1484914.

[49] S. Liu *et al.*, “A Primer on Zeroth-Order Optimization in Signal Processing and Machine Learning: Principals, Recent Advances, and Applications,” *IEEE Signal Process Mag*, vol. 37, no. 5, pp. 43–54, Sep. 2020, doi: 10.1109/MSP.2020.3003837.

APPENDIX A. SUBJECT-SPECIFIC SMPL MODELS GENERATED FROM PARTIAL POINT CLOUD DATA

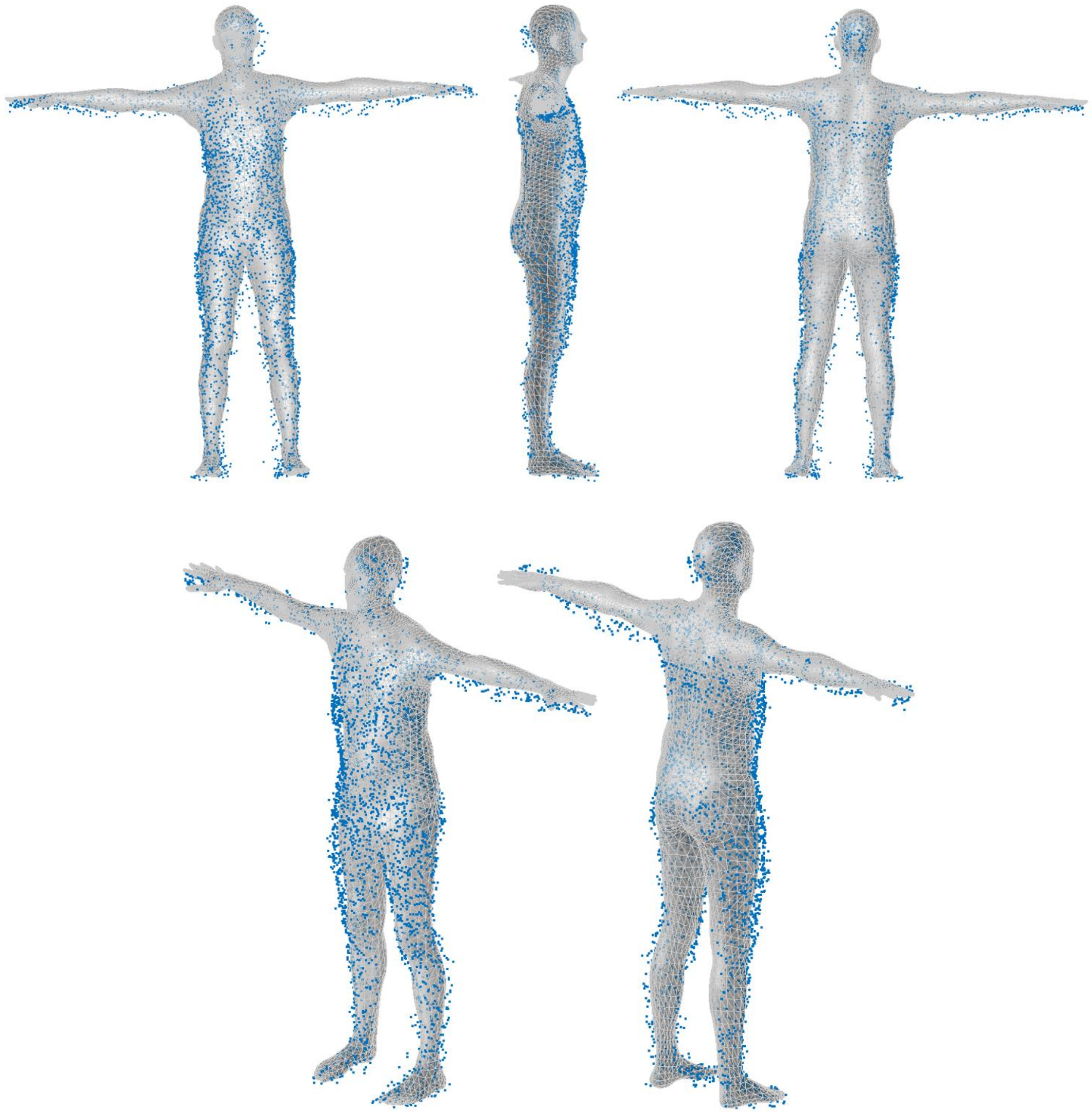


Figure A.1. The SMPL Model generated using partial point cloud data from S01

Table A.1 The shape parameters to generate the S01 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
0.13	1.29	1.86	-0.73	1.89	-2.90	-0.16	1.42	1.33	0.46

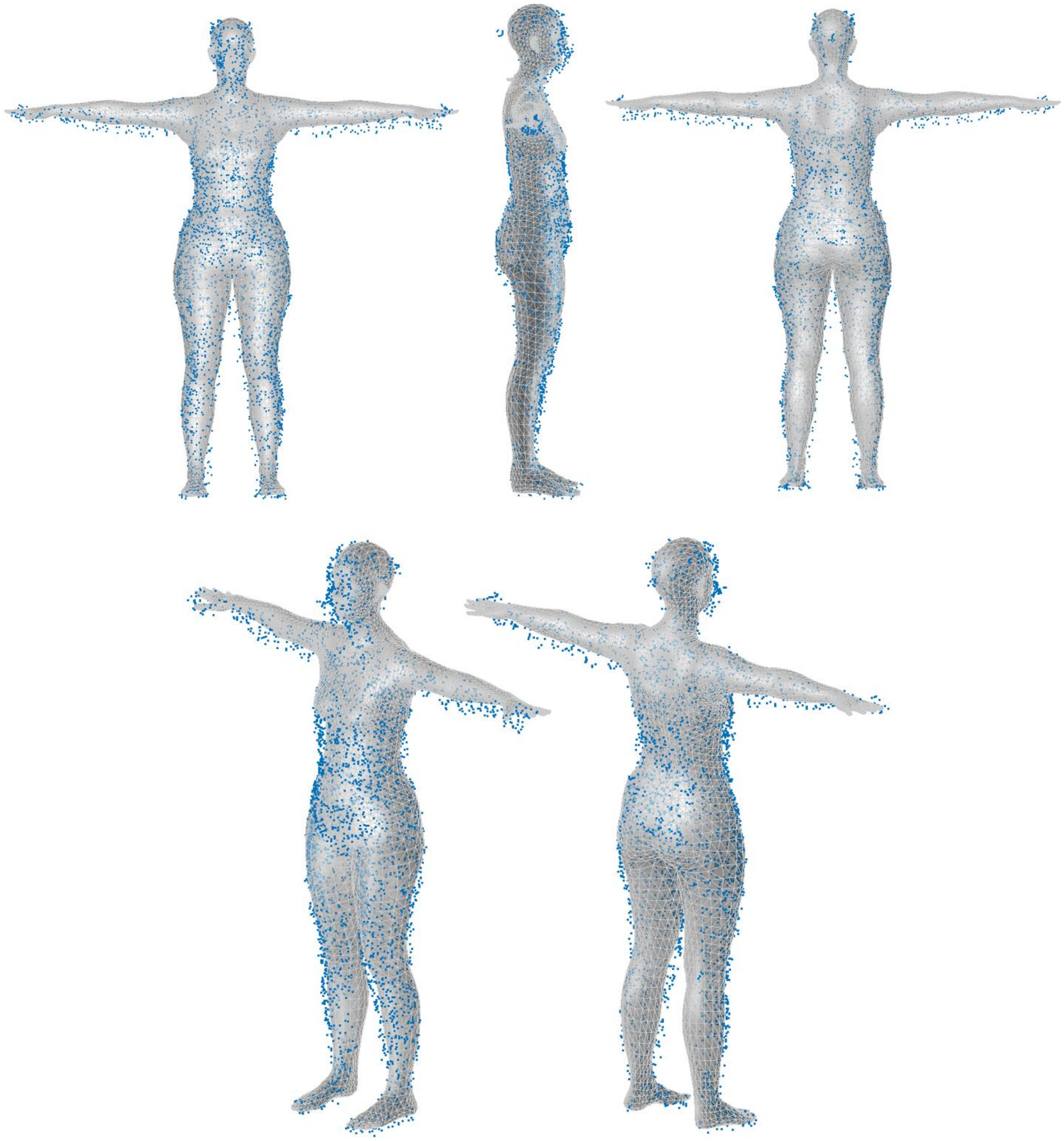


Figure A.2. The SMPL Model generated using partial point cloud data from S02

Table A.2 The shape parameters to generate the S02 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-0.64	1.01	2.45	0.07	-0.55	-2.30	-2.13	1.86	1.10	2.36

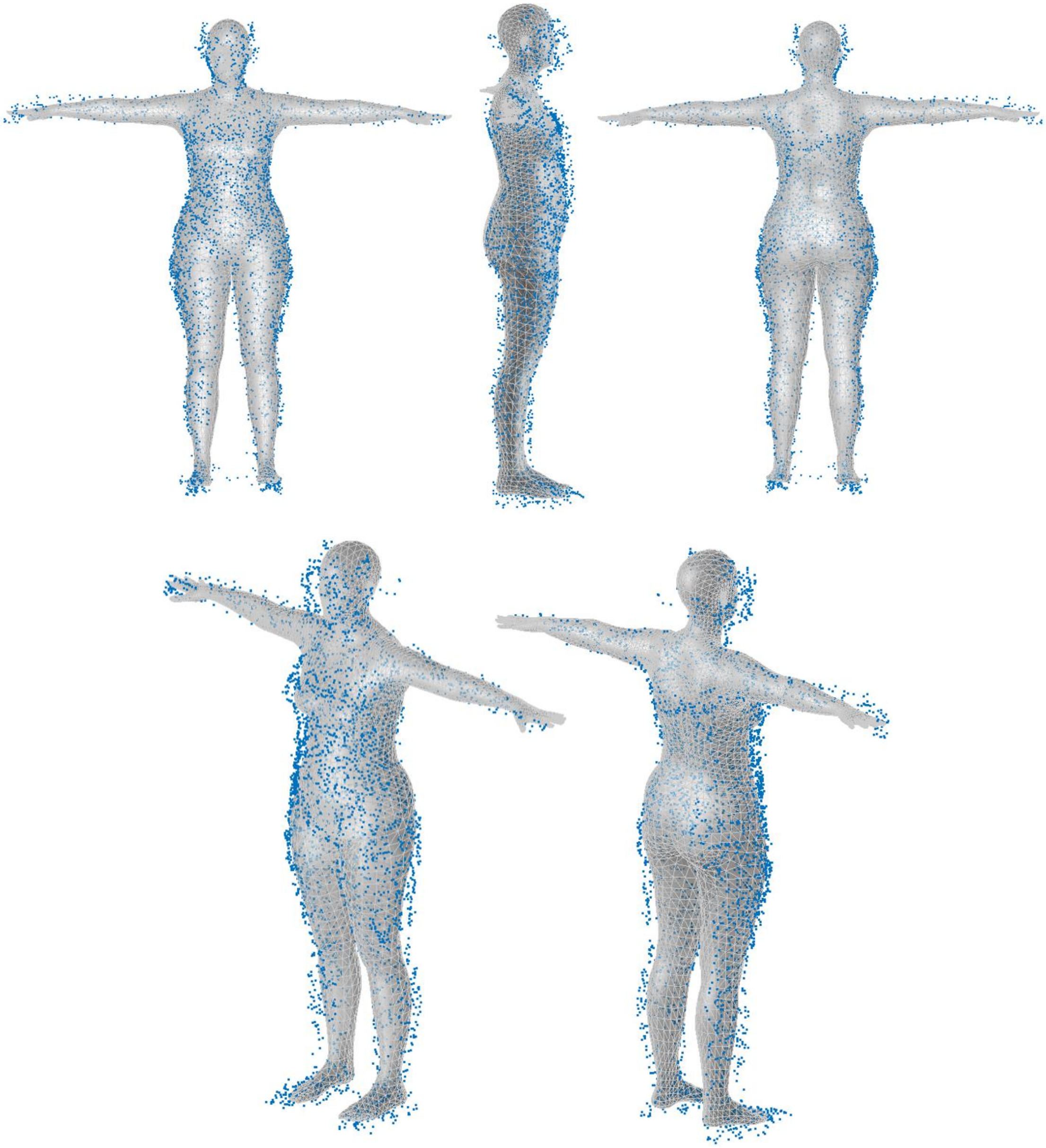


Figure A.3. The SMPL Model generated using partial point cloud data from S03

Table A.3 The shape parameters to generate the S03 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-0.79	0.33	2.06	1.97	-0.82	-2.79	-1.58	1.14	0.93	2.08

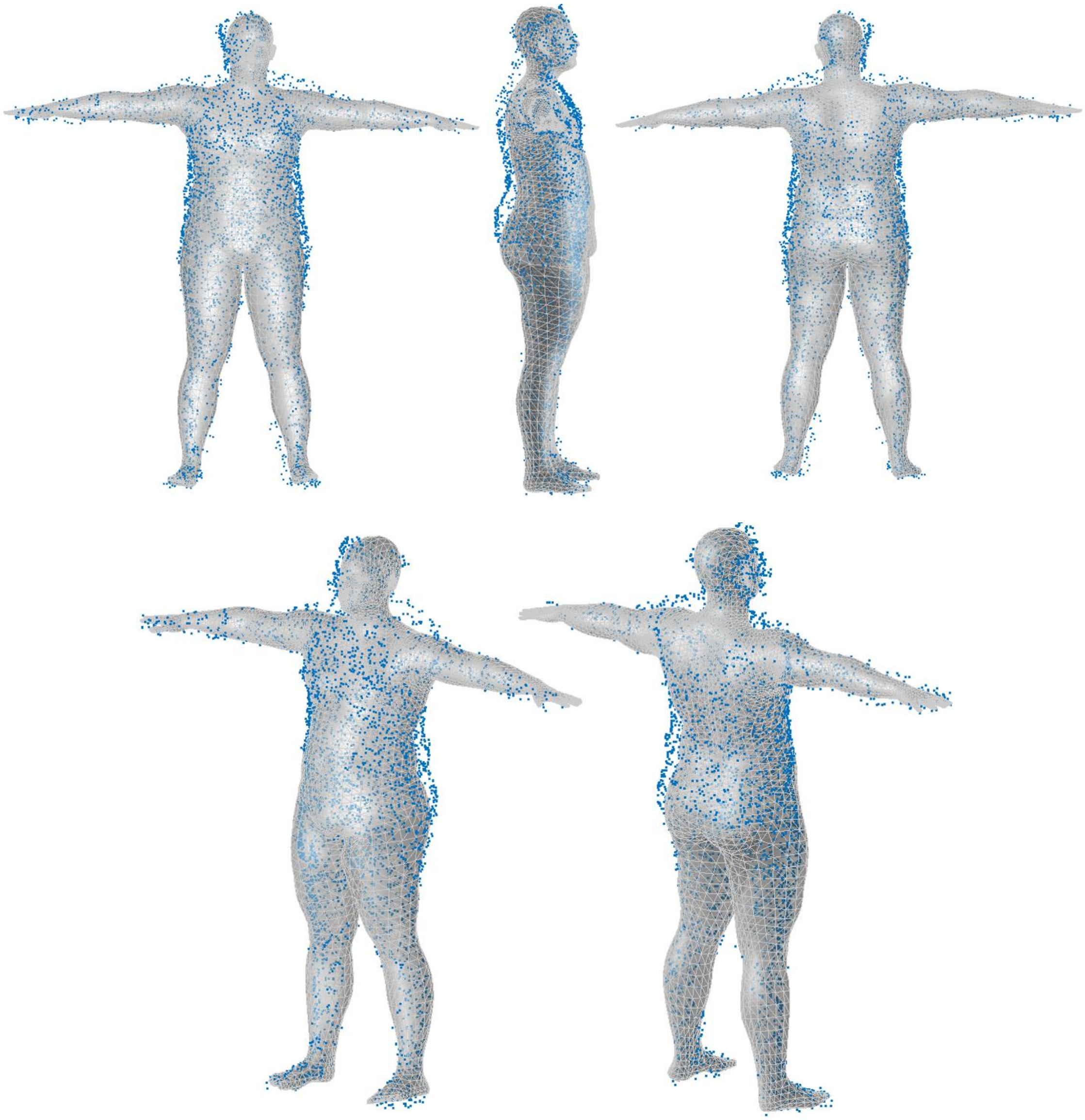


Figure A.4. The SMPL Model generated using partial point cloud data from S05

Table A.4 The shape parameters to generate the S05 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-0.65	-0.51	0.92	0.19	3.26	8.02	-3.10	0.74	4.60	3.41

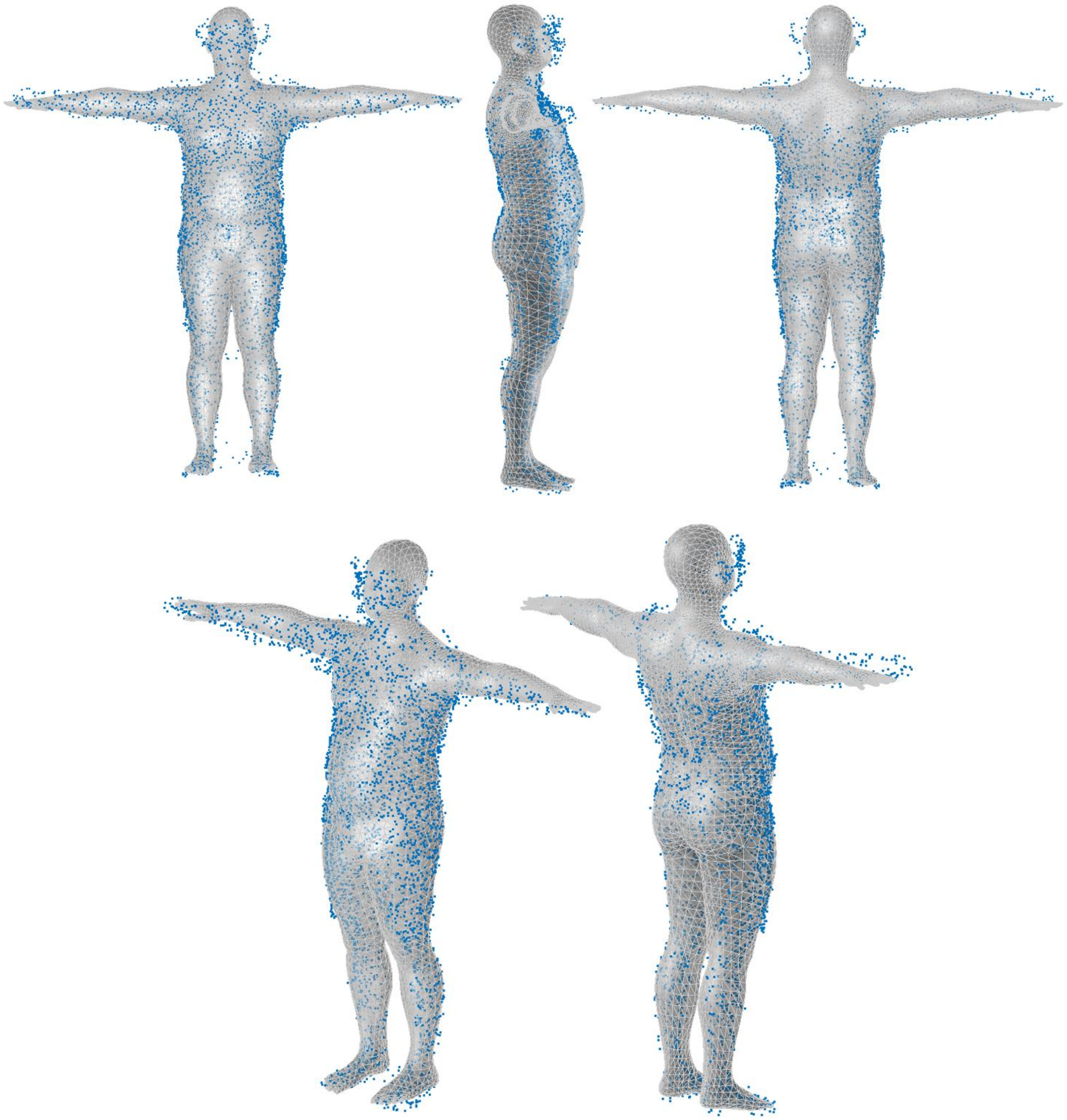


Figure A.5. The SMPL Model generated using partial point cloud data from S06

Table A.5 The shape parameters to generate the S06 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-0.20	-1.22	2.68	0.96	0.41	2.17	-1.47	0.69	-1.62	1.40

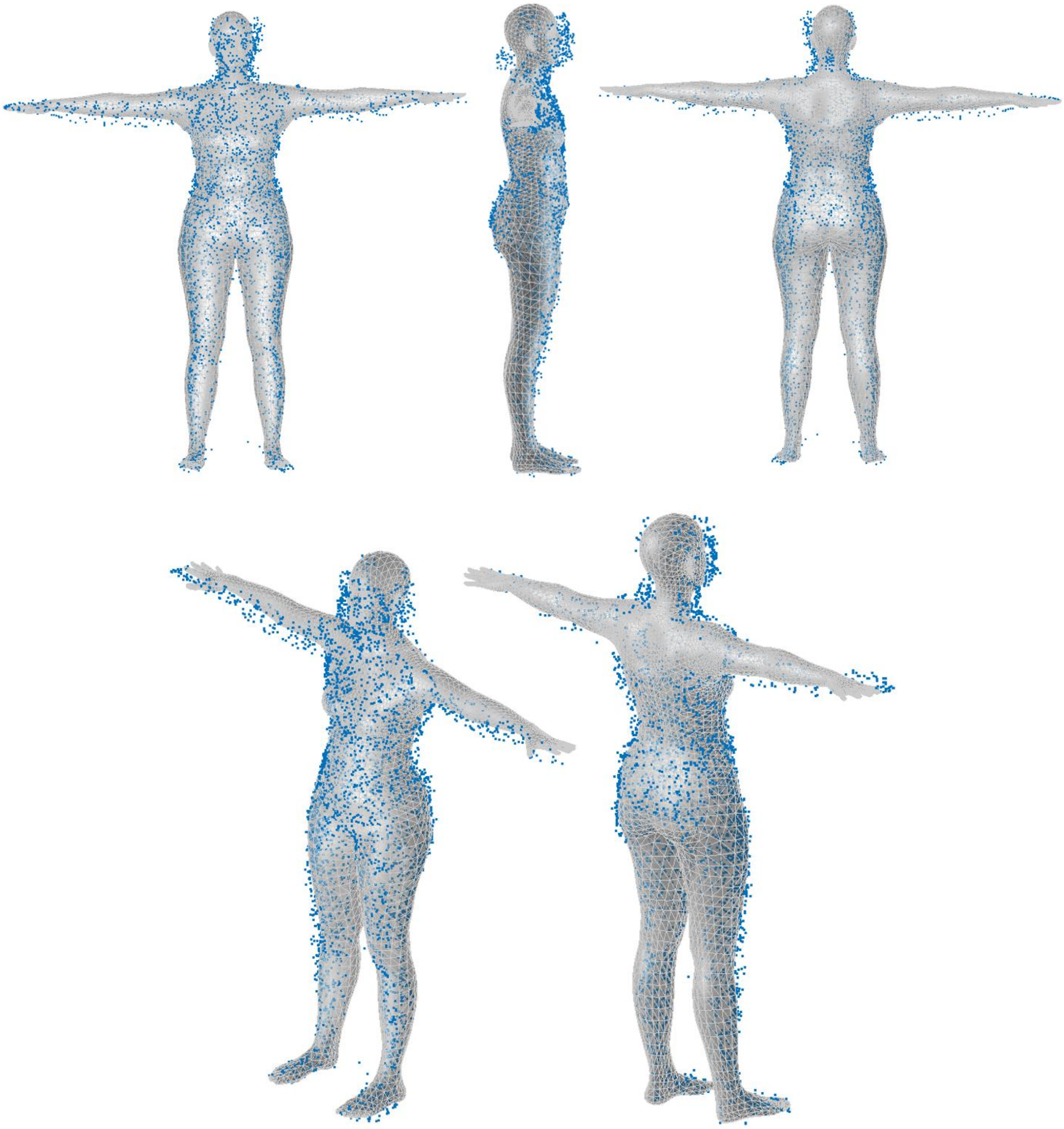


Figure A.6. The SMPL Model generated using partial point cloud data from S07

Table A.6 The shape parameters to generate the S07 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-0.30	0.28	-0.50	0.28	0.16	-2.07	-2.22	2.25	2.07	2.48

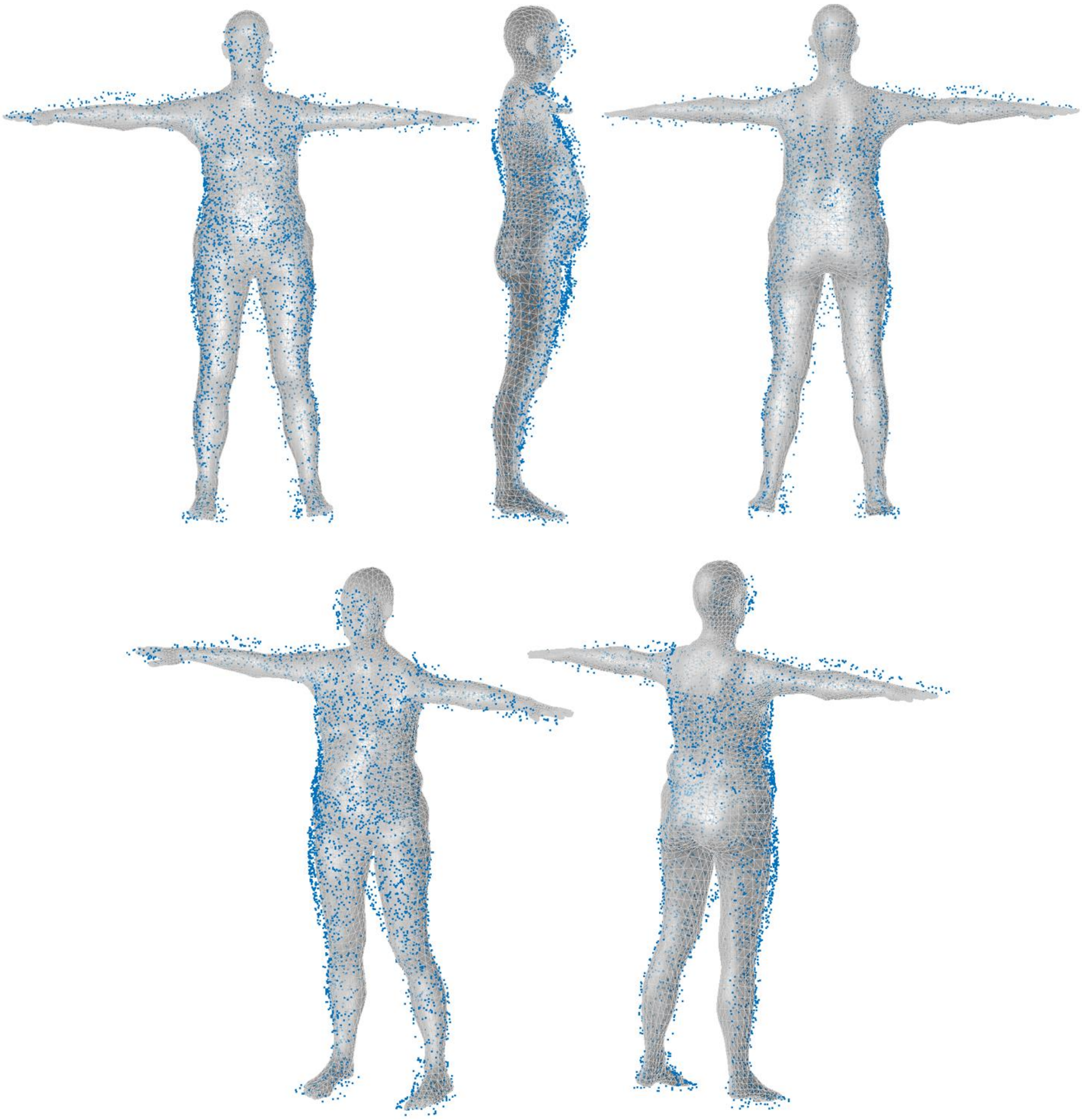


Figure A.7. The SMPL Model generated using partial point cloud data from S09

Table A.7 The shape parameters to generate the S09 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
0.46	1.45	3.40	-1.66	3.05	1.79	0.02	12.95	5.69	2.97

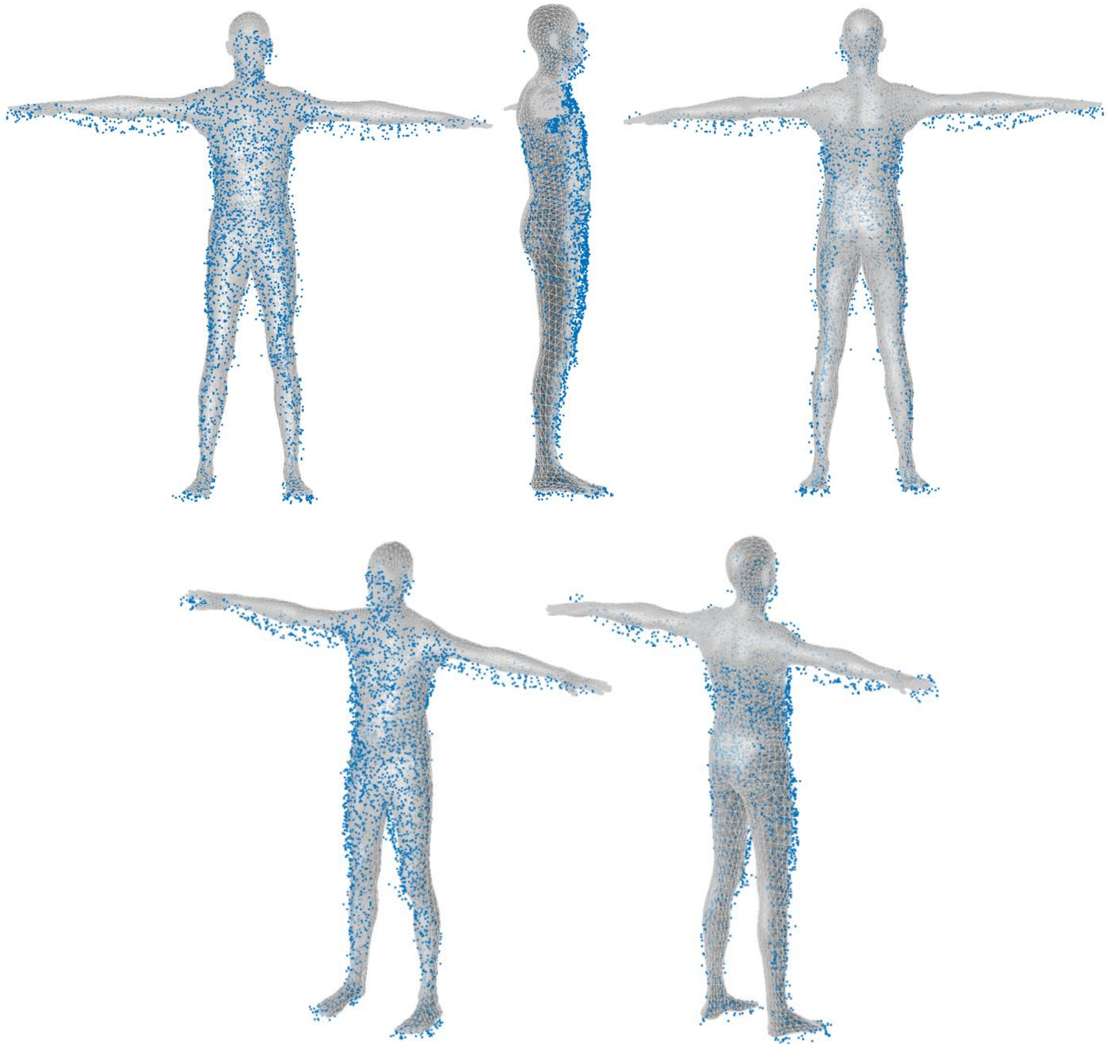


Figure A.8. The SMPL Model generated using partial point cloud data from S10

Table A.8 The shape parameters to generate the S10 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-0.60	1.77	0.66	-1.34	-1.49	-2.65	0.85	0.30	-1.50	0.29

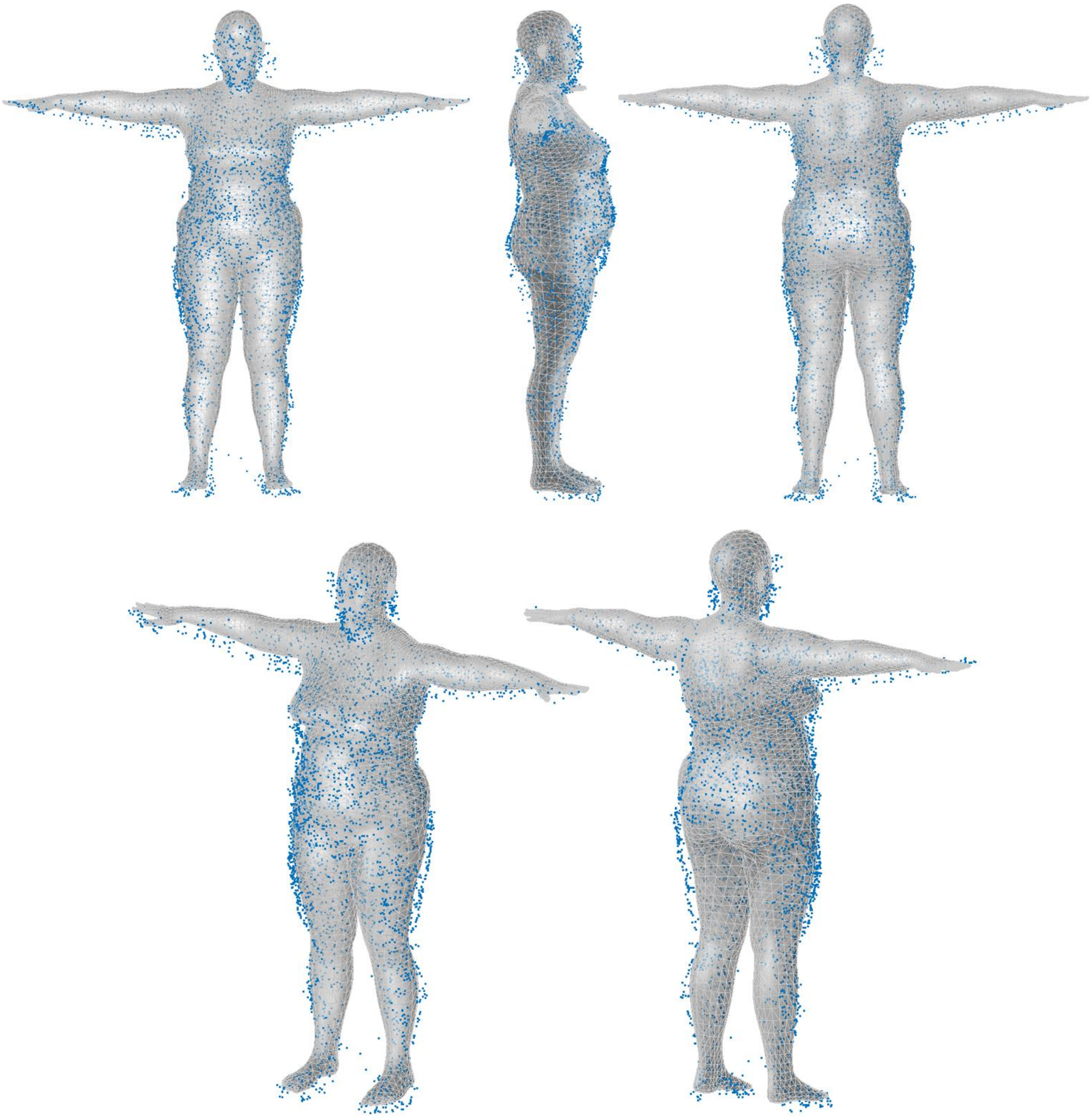


Figure A.9. The SMPL Model generated using partial point cloud data from S11

Table A.9 The shape parameters to generate the S11 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-0.82	-1.56	2.19	1.72	-0.65	-2.46	-0.39	2.62	-1.87	1.06

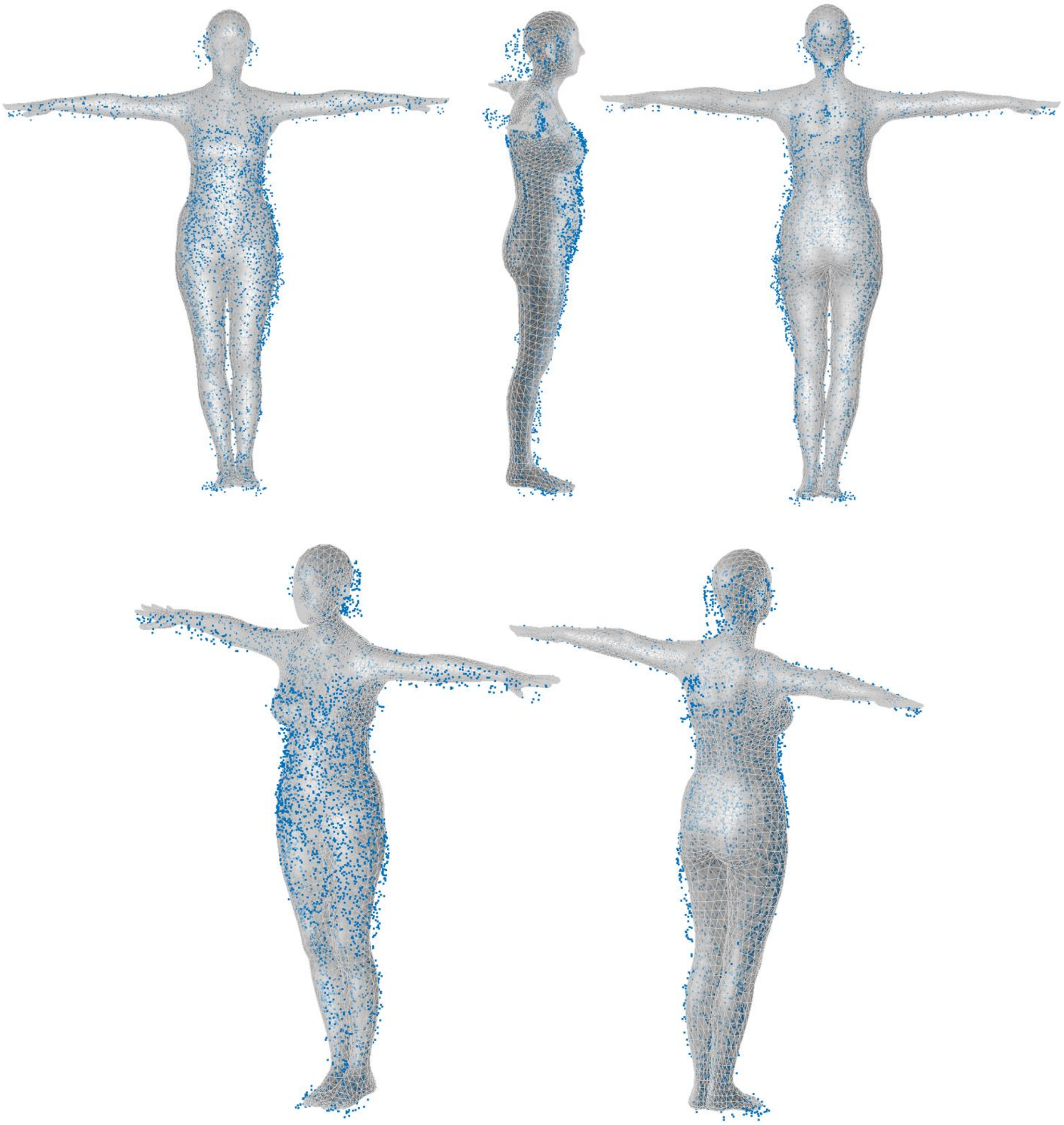


Figure A.10. The SMPL Model generated using partial point cloud data from S15

Table A.10 The shape parameters to generate the S15 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-1.28	0.90	2.75	0.48	0.78	-1.55	0.86	2.55	-0.26	-2.64

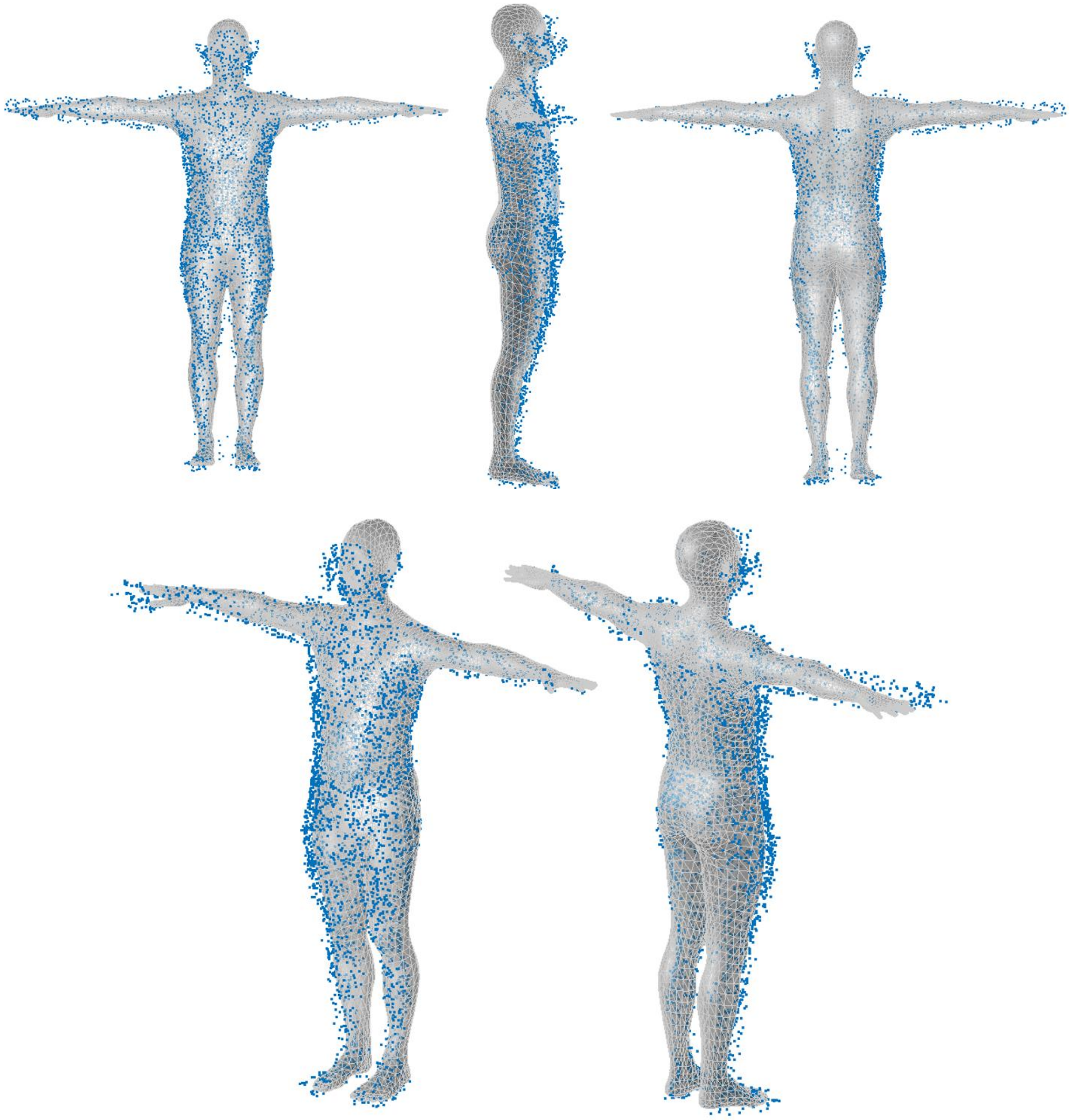


Figure A.11. The SMPL Model generated using partial point cloud data from S16

Table A.11 The shape parameters to generate the S16 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
0.62	0.66	2.00	0.93	0.42	-1.49	-1.24	1.08	-0.50	1.70

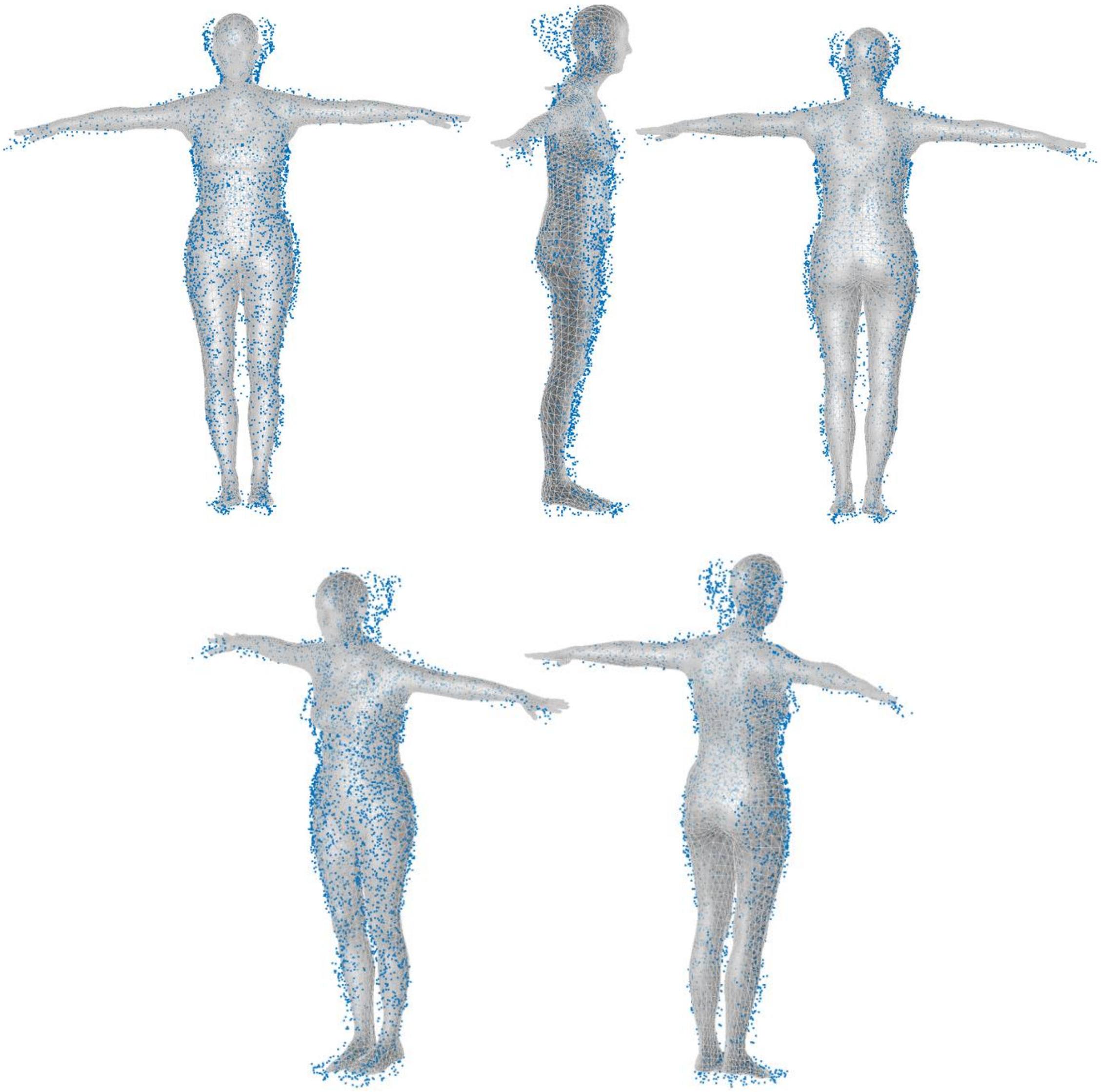


Figure A.12. The SMPL Model generated using partial point cloud data from S19

Table A.12 The shape parameters to generate the S19 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-0.88	0.75	1.99	-0.06	1.42	-2.40	0.29	2.36	1.21	2.91

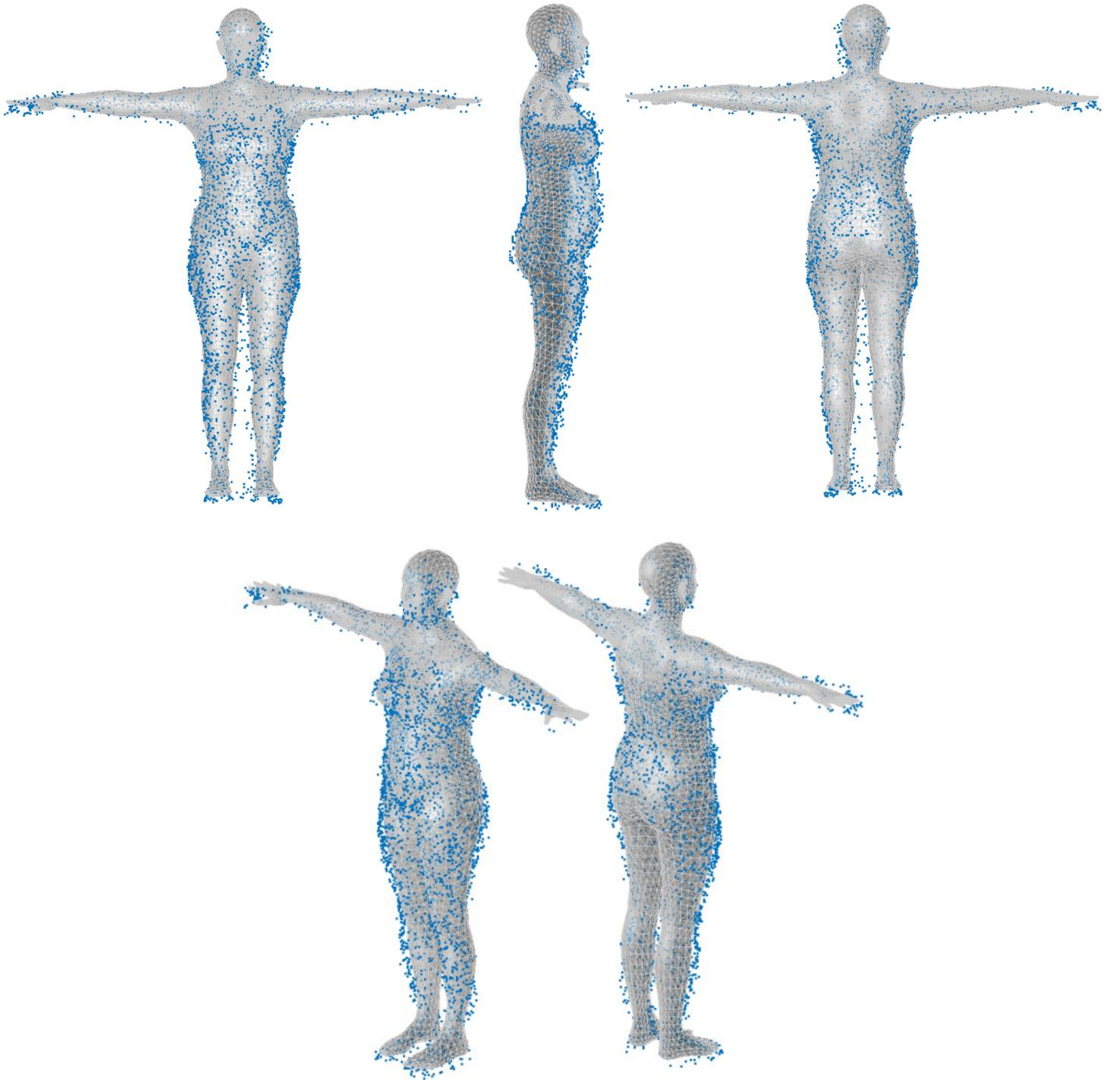


Figure A.13. The SMPL Model generated using partial point cloud data from S20

Table A.13 The shape parameters to generate the S20 SMPL Model- partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-0.23	0.36	1.28	0.25	-1.88	-2.29	1.31	1.41	1.30	0.92

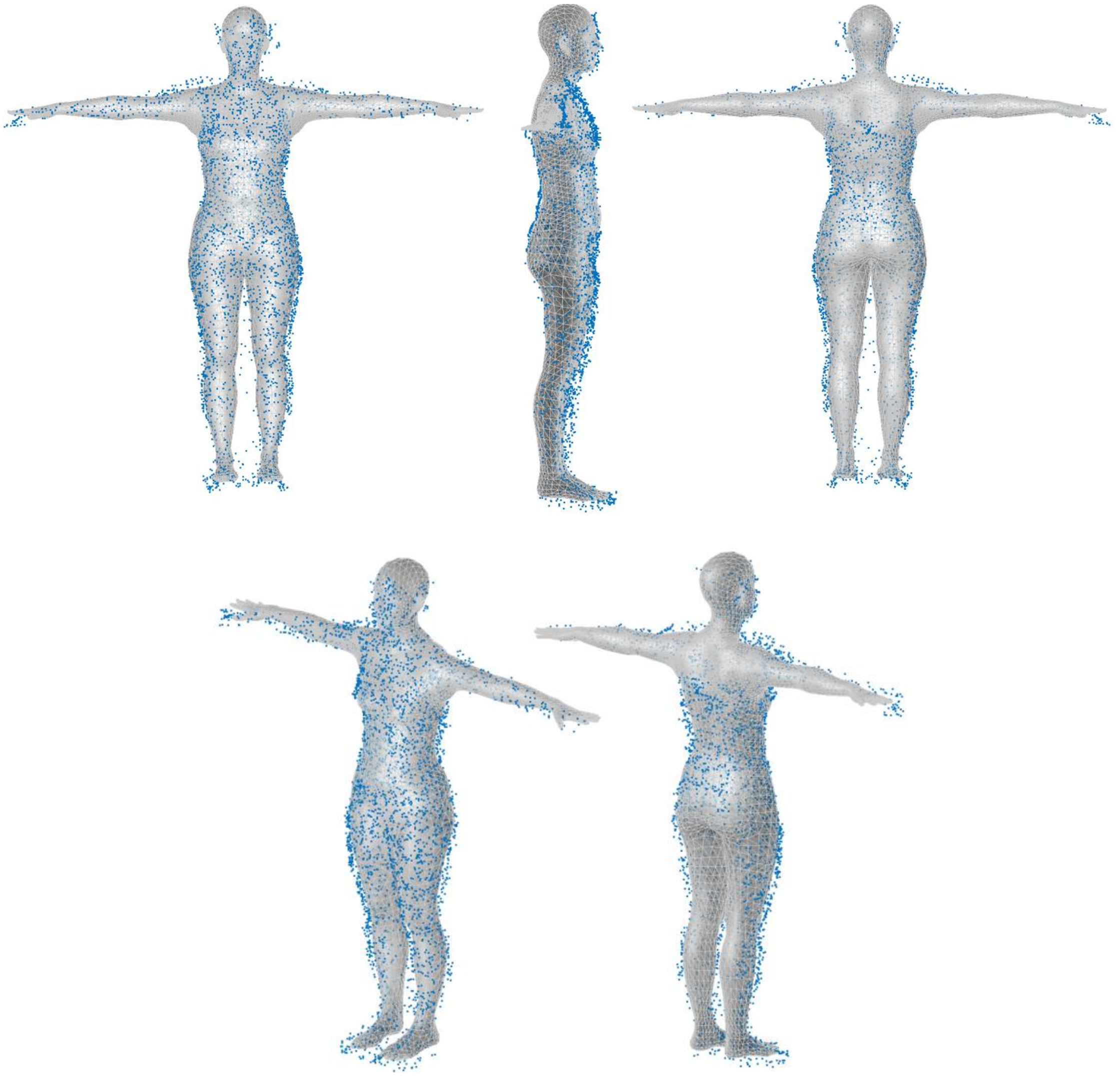


Figure A.14. The SMPL Model generated using partial point cloud data from S21

Table A.14 The shape parameters to generate the S21 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-1.27	0.41	-0.09	-1.89	-0.24	-1.98	-1.47	1.85	2.27	2.29

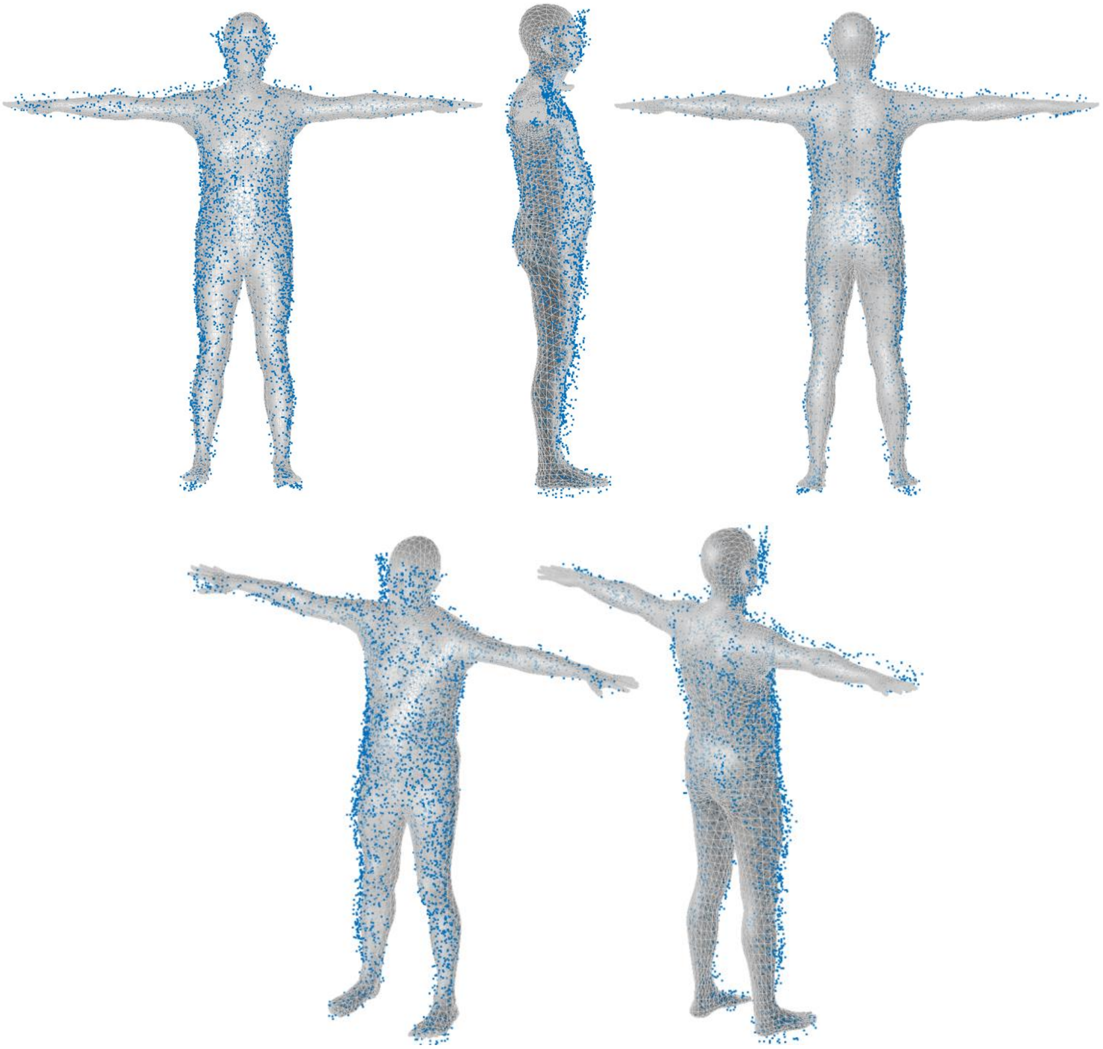


Figure A.15. The SMPL Model generated using partial point cloud data from S22

Table A.15 The shape parameters to generate the S22 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
1.61	-0.30	0.85	-0.80	-1.00	-2.60	-0.21	1.00	2.84	1.80

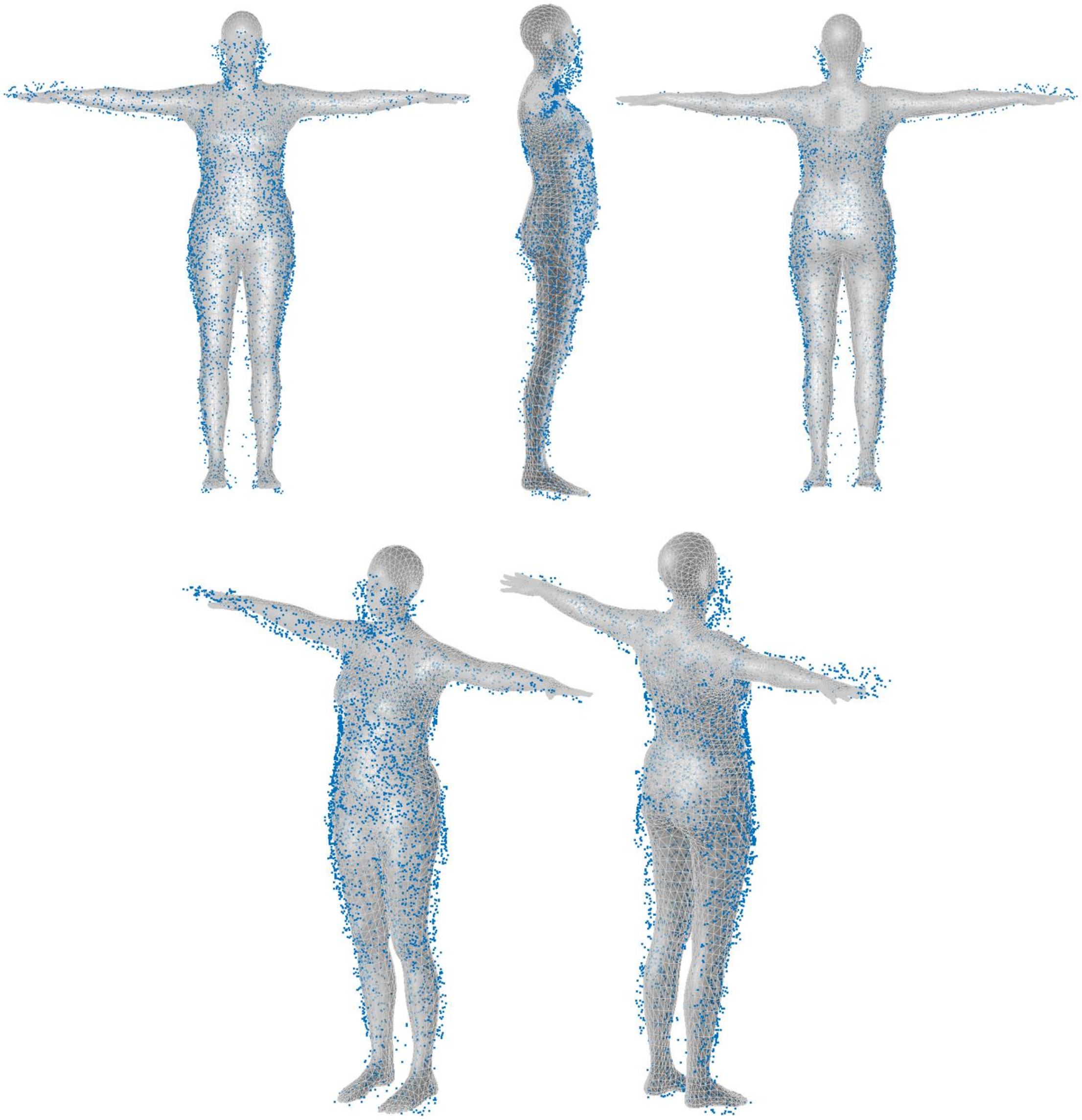


Figure A.16. The SMPL Model generated using partial point cloud data from S23

Table A.16 The shape parameters to generate the S23 SMPL Model - partial point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
2.21	1.10	1.73	1.16	-0.09	-1.90	1.54	1.14	1.36	2.32

APPENDIX B. SUBJECT-SPECIFIC SMPL MODELS GENERATED FROM FULL POINT CLOUD DATA

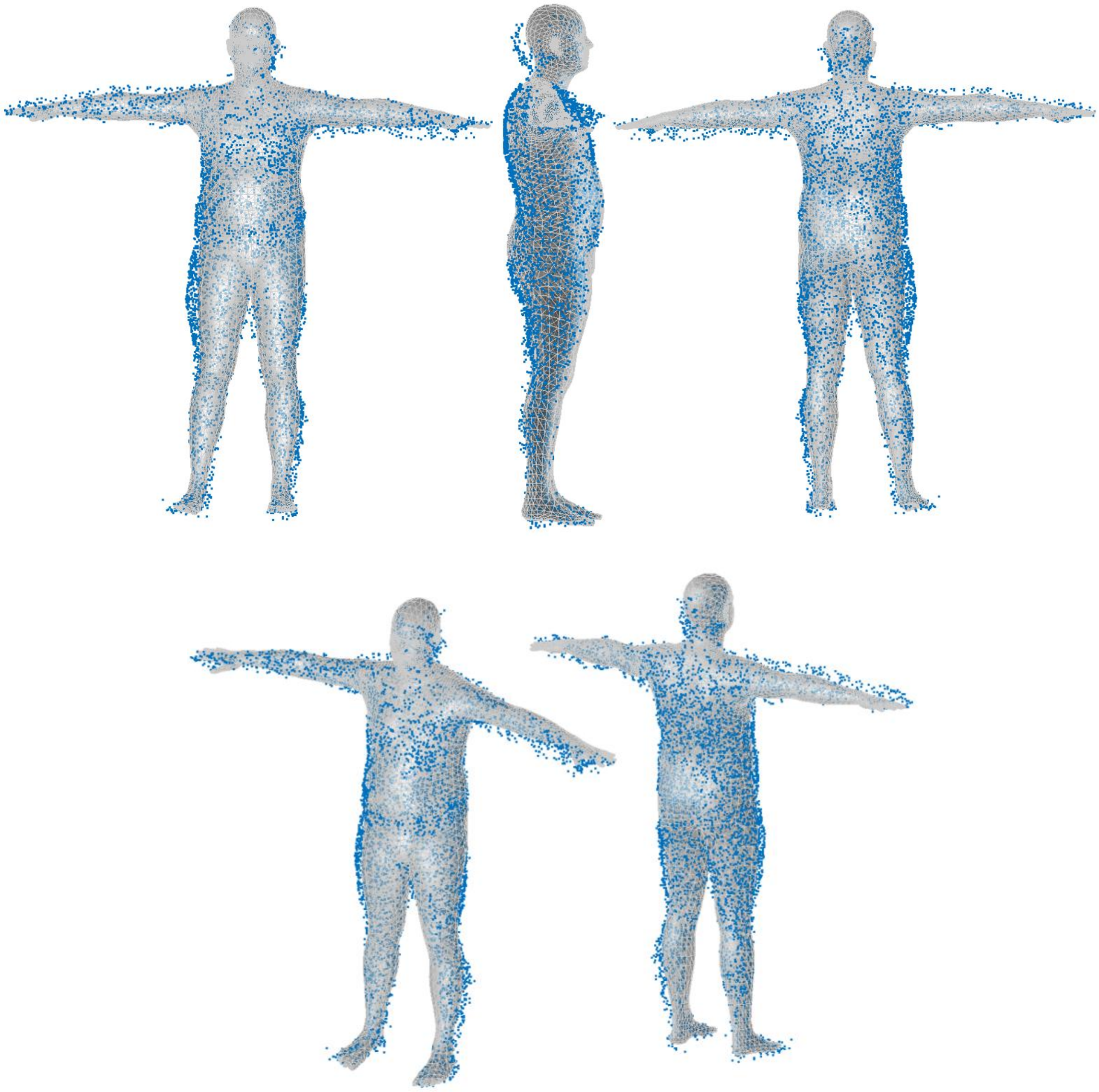


Figure B.1 The SMPL Model generated using full point cloud data from S09

Table B.1 The shape parameters to generate the S09 SMPL Model – full point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-0.14	-0.14	1.30	-2.22	0.23	2.23	0.23	1.33	0.92	2.57

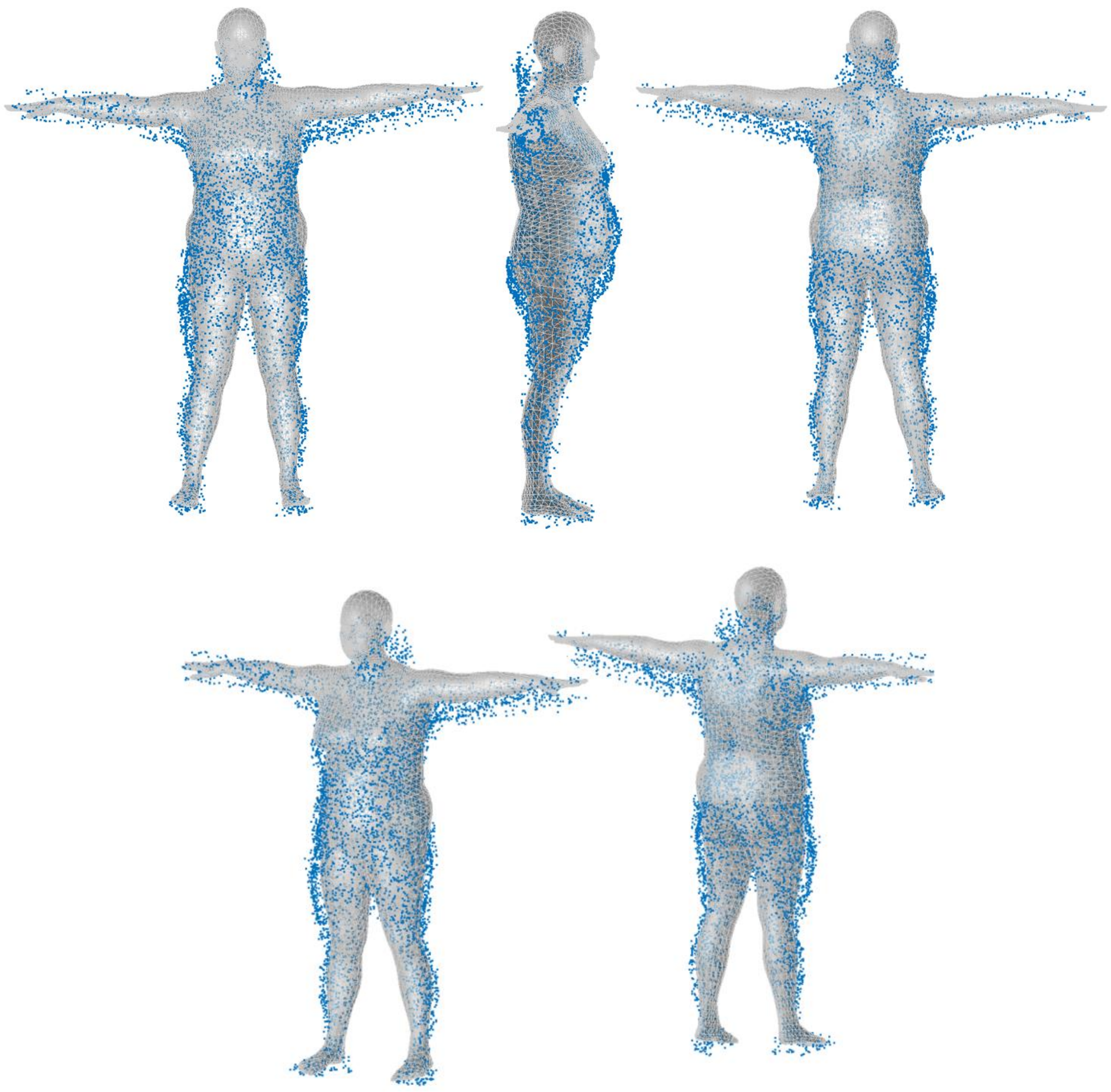


Figure B.2 The SMPL Model generated using full point cloud data from S11

Table B.2 The shape parameters to generate the S11 SMPL Model – full point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
-0.21	-1.60	1.63	0.45	2.02	-2.88	1.70	-2.47	0.57	1.79

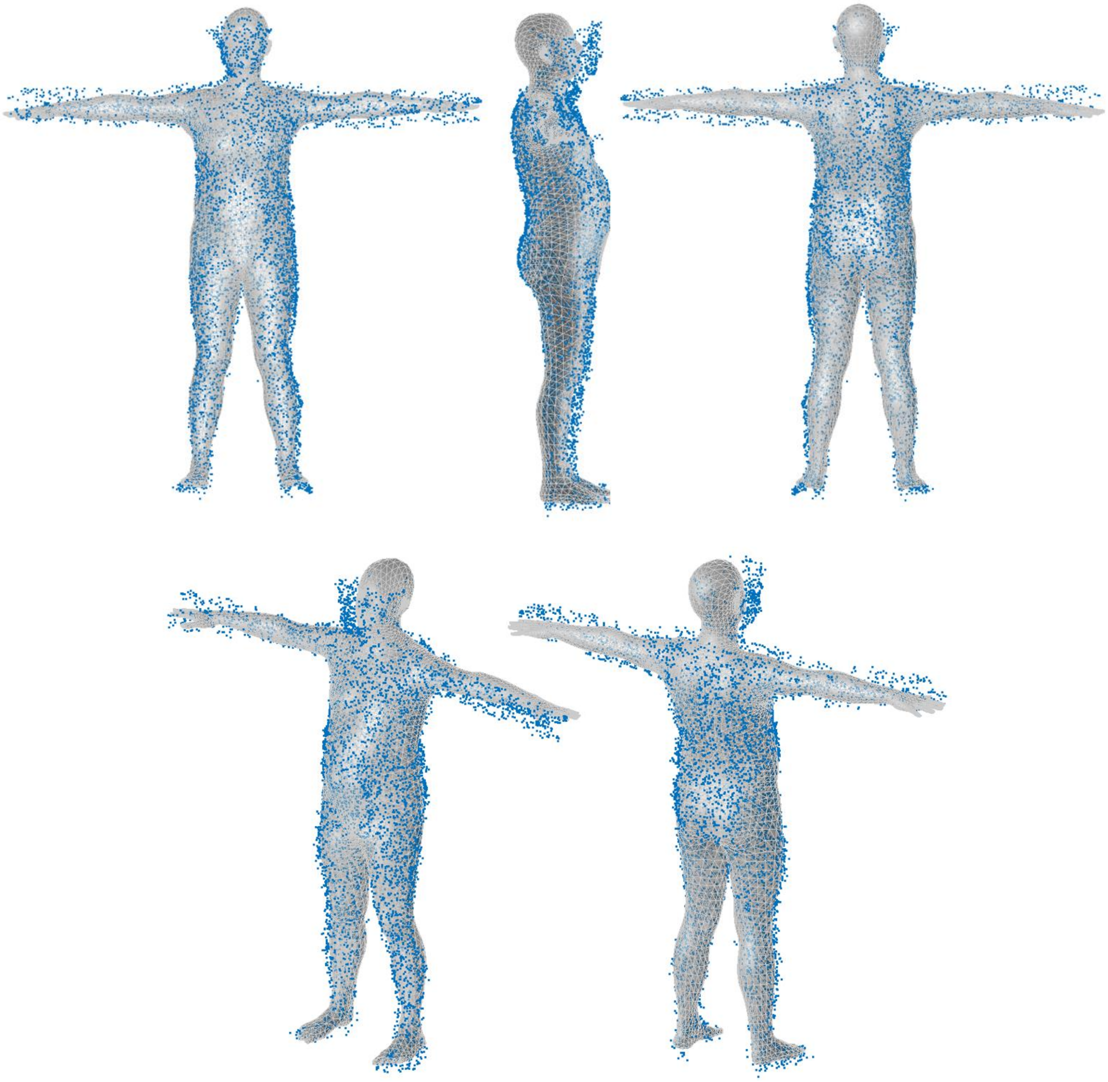


Figure B.3 The SMPL Model generated using full point cloud data from S22

Table B.3 The shape parameters to generate the S22 SMPL Model – full point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
2.01	-0.23	2.14	-1.24	0.90	0.10	1.35	0.56	2.87	-0.16

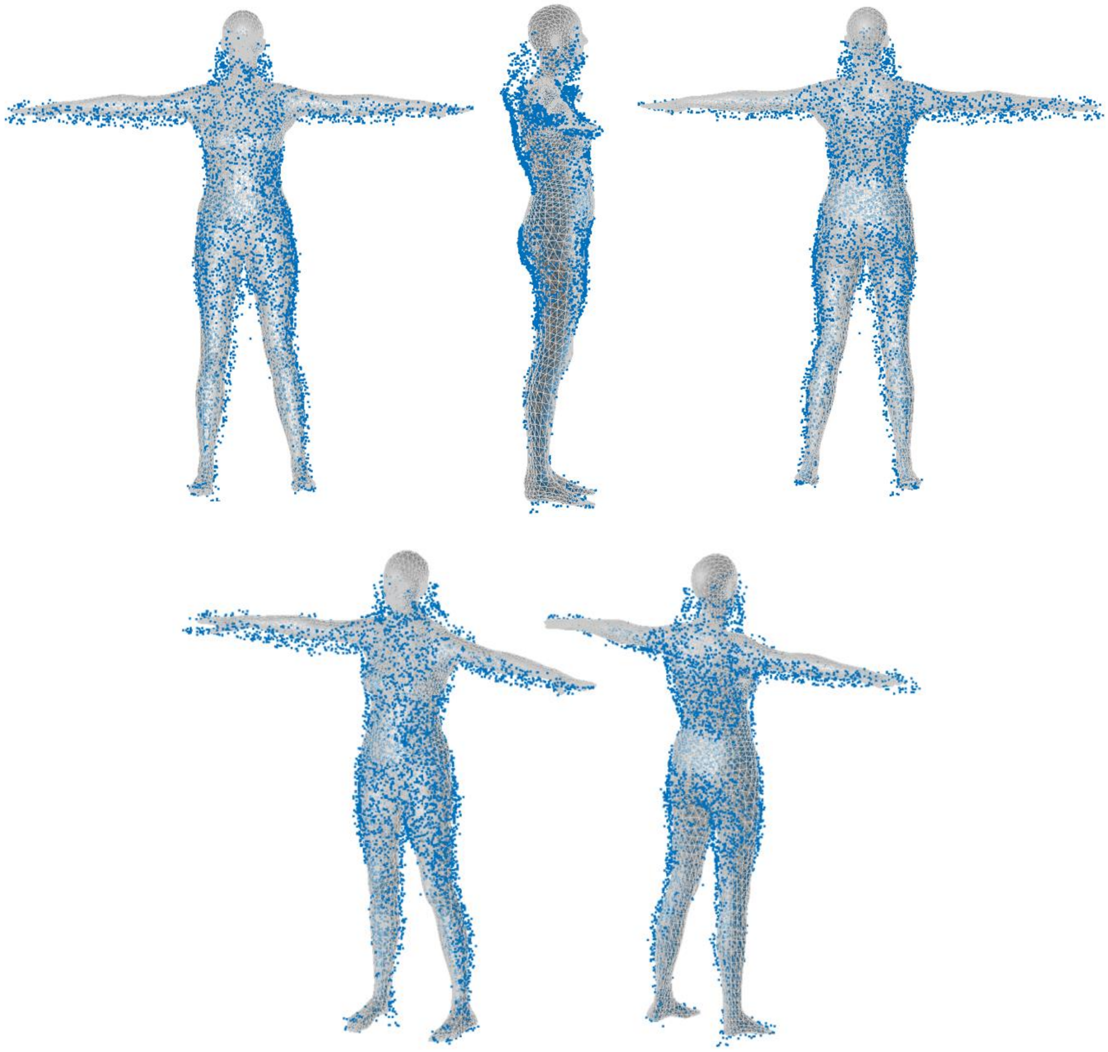


Figure B.4 The SMPL Model generated using full point cloud data from S23

Table B.4 The shape parameters to generate the S23 SMPL Model – full point cloud

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
1.99	1.78	2.61	1.51	1.47	-2.72	1.45	0.10	-2.30	0.03

APPENDIX C. CORRELATIONS BETWEEN BODY MEASUREMENT AND SMPL SHAPE PARAMETER

Table C.1 Correlations between SMPL shape parameters found with the partial point cloud and physical anthropometric measurement (N=16). Strong correlations (>0.6) are indicated in red

	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
Height	0.47	0.14	0.19	-0.16	0.42	0.56	-0.06	0.09	0.17	0.24
Floor to Shoulder	0.46	0.15	0.13	-0.10	0.43	0.58	-0.03	0.06	0.16	0.22
Bicep Upper	0.23	-0.40	0.14	0.22	0.36	0.69	-0.22	0.16	0.10	0.26
Bicep Lower	0.27	-0.36	0.14	0.14	0.44	0.75	-0.32	0.12	0.15	0.29
Forearm Upper	0.25	-0.31	0.19	0.04	0.51	0.75	-0.34	0.15	0.18	0.22
Wrist	0.26	-0.28	0.09	-0.10	0.40	0.59	-0.32	-0.08	0.09	0.12
Armpit to wrist	0.37	0.37	-0.04	-0.33	0.20	0.28	0.27	-0.03	-0.05	-0.06
Thigh Upper	0.11	-0.43	0.25	0.40	0.42	0.70	-0.28	0.21	0.23	0.41
Thigh Lower	0.18	-0.41	0.32	0.29	0.40	0.71	-0.44	0.19	0.12	0.40
Calf Upper	0.21	-0.33	0.27	0.16	0.55	0.90	-0.35	0.24	0.36	0.41
Calf Largest	0.23	-0.49	0.24	0.17	0.43	0.82	-0.37	0.19	0.27	0.46
Ankle	0.42	-0.28	0.30	-0.09	0.50	0.75	-0.26	0.23	0.35	0.35
groin to floor	0.53	0.28	-0.03	-0.01	0.26	0.29	0.12	-0.17	-0.01	0.06
Hips	0.09	-0.51	0.20	0.38	0.43	0.78	-0.32	0.11	0.20	0.37
Waist	0.23	-0.51	0.31	0.16	0.39	0.69	-0.10	0.28	0.11	0.16

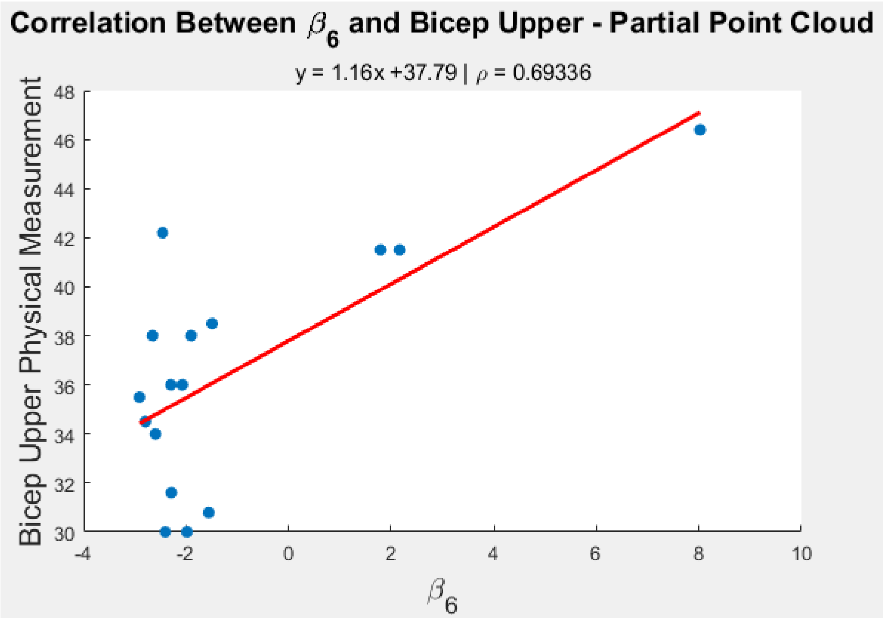


Figure C.1 Correlations visualized between β6 found using the partial point cloud and the bicep upper measurement.

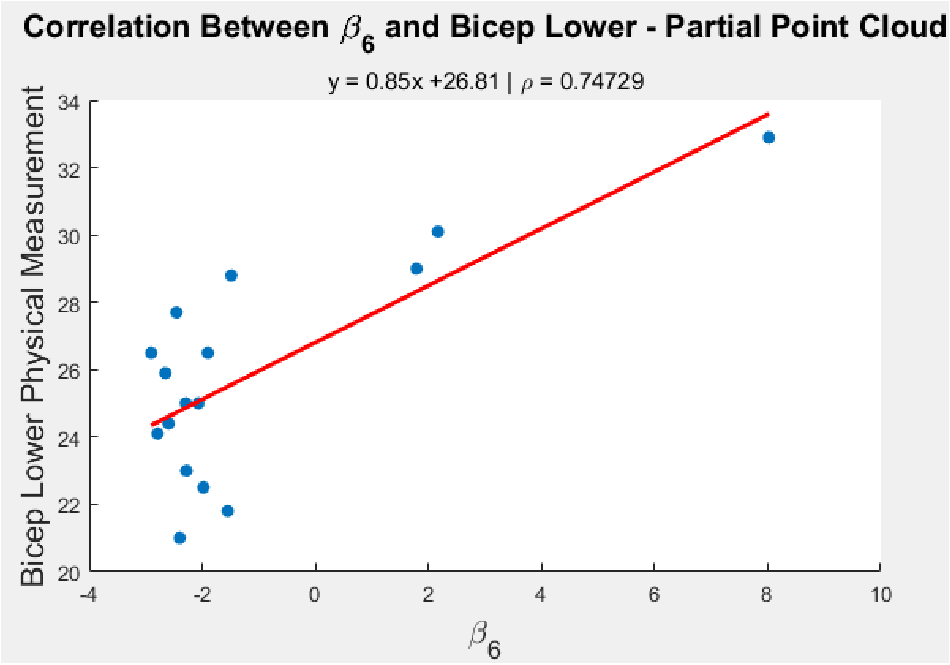


Figure C.2 Correlations visualized between β6 found using the partial point cloud and the bicep lower measurement.

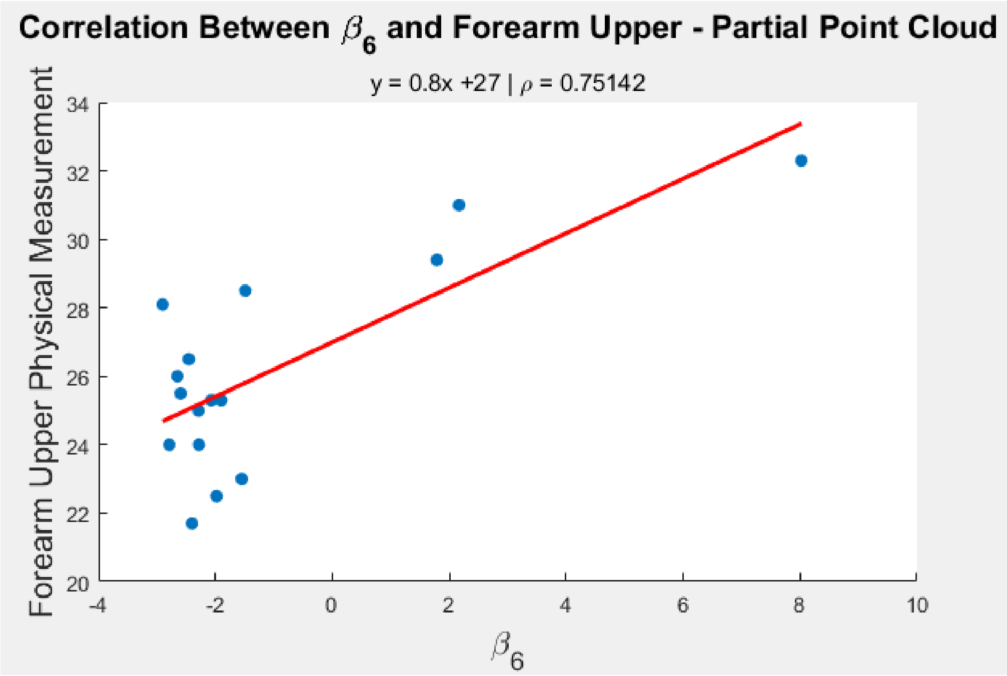


Figure C.3 Correlations visualized between β6 found using the partial point cloud and the forearm upper measurement.

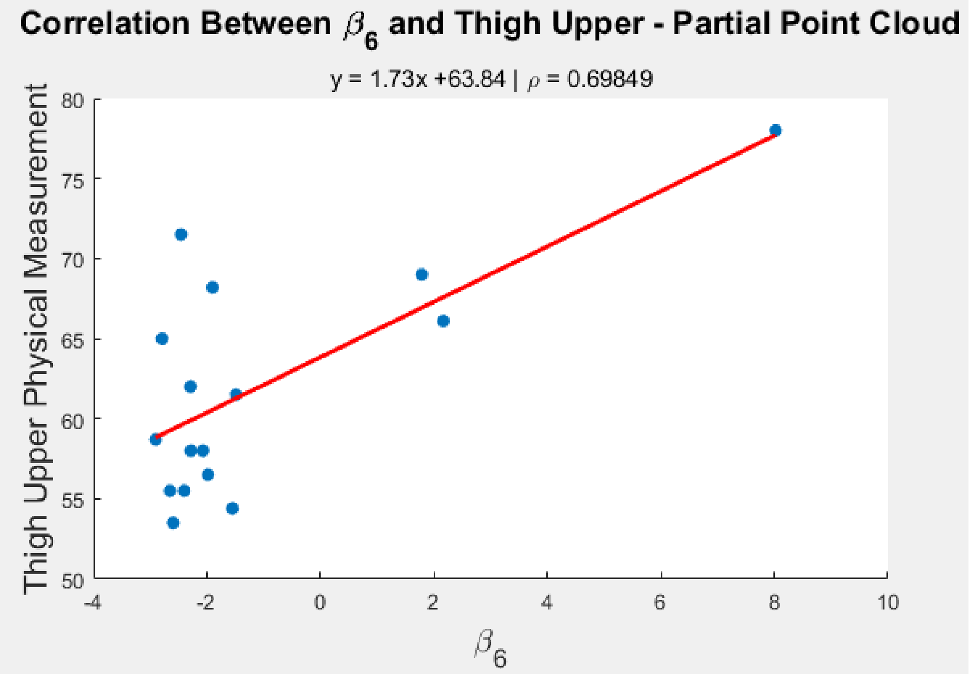


Figure C.4 Correlations visualized between β6 found using the partial point cloud and the thigh upper measurement.

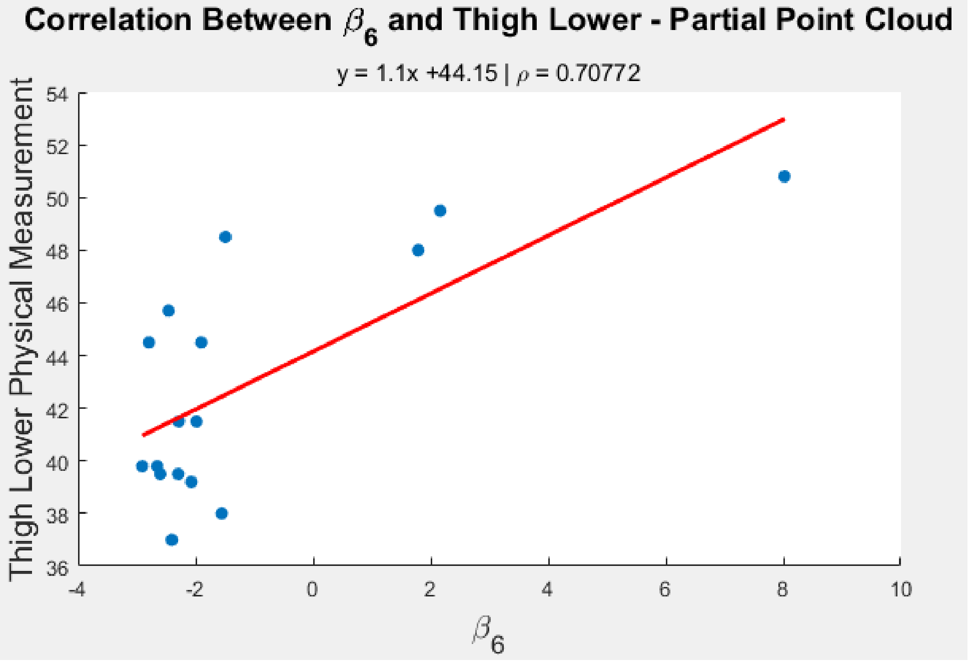


Figure C.5 Correlations visualized between β6 found using the partial point cloud and the thigh lower measurement.

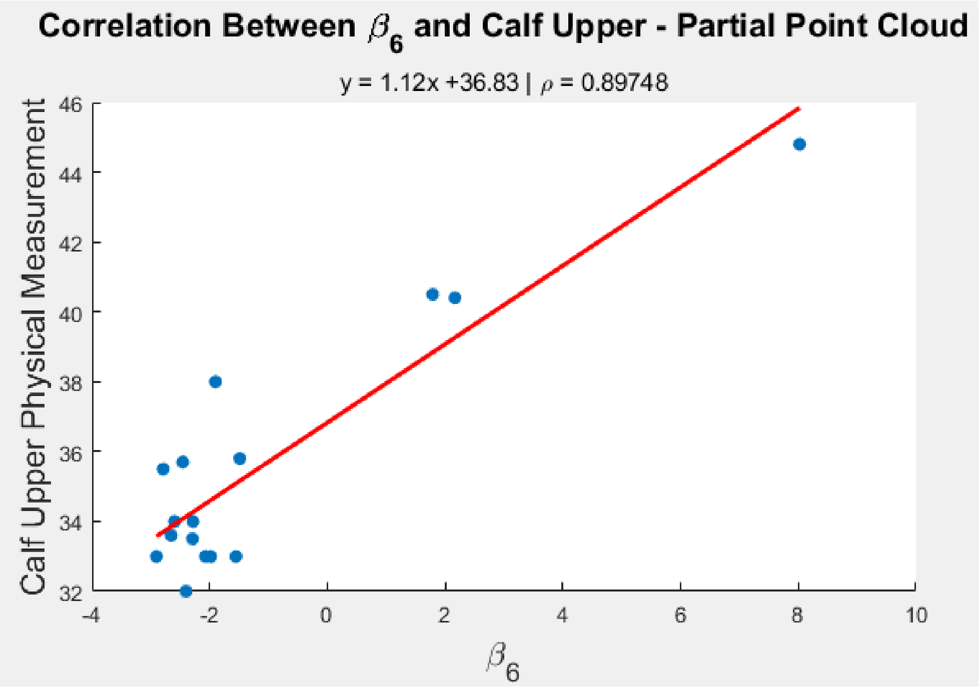


Figure C.6 Correlations visualized between β6 found using the partial point cloud and the calf upper measurement.

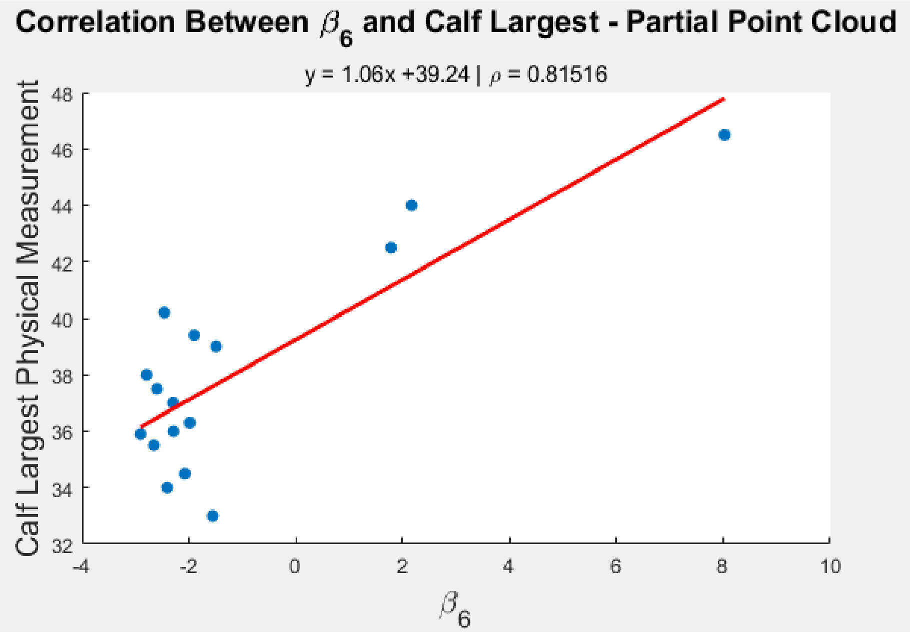


Figure C.7 Correlations visualized between β6 found using the partial point cloud and the calf largest measurement.

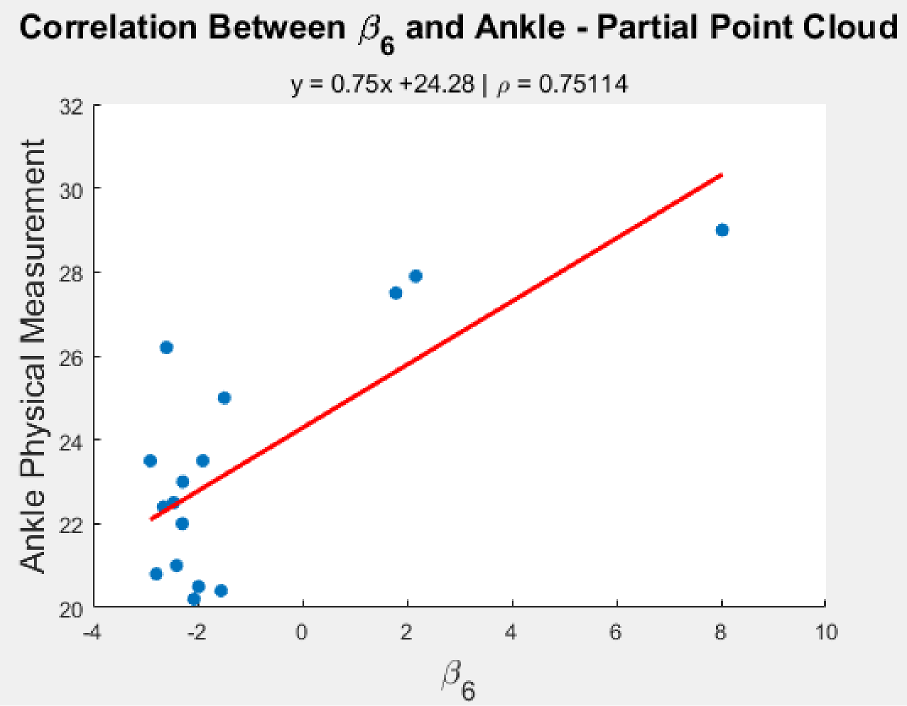


Figure C.8 Correlations visualized between β6 found using the partial point cloud and the ankle measurement.

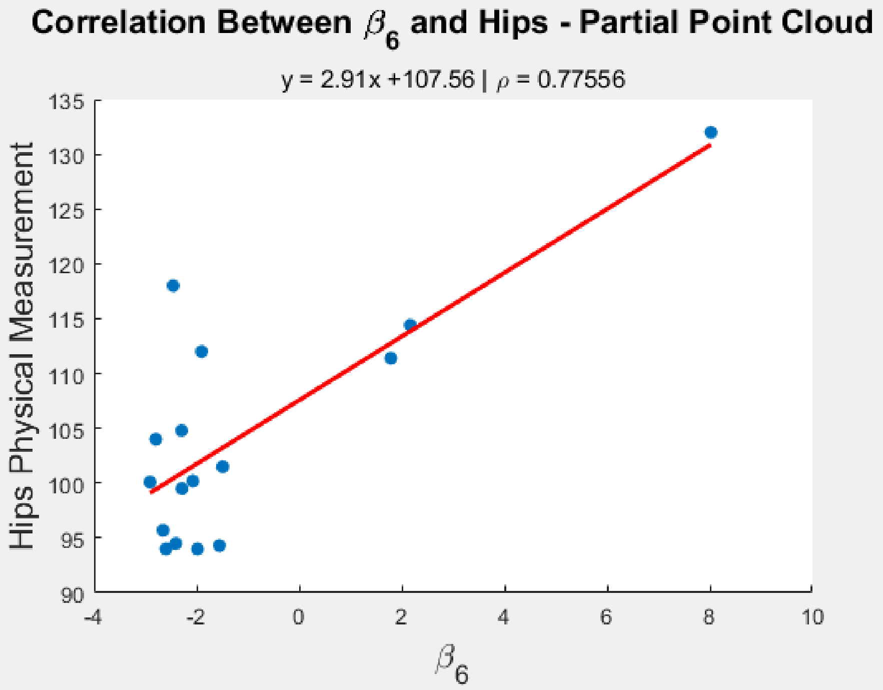


Figure C.9 Correlations visualized between β6 found using the partial point cloud and the hips measurement.

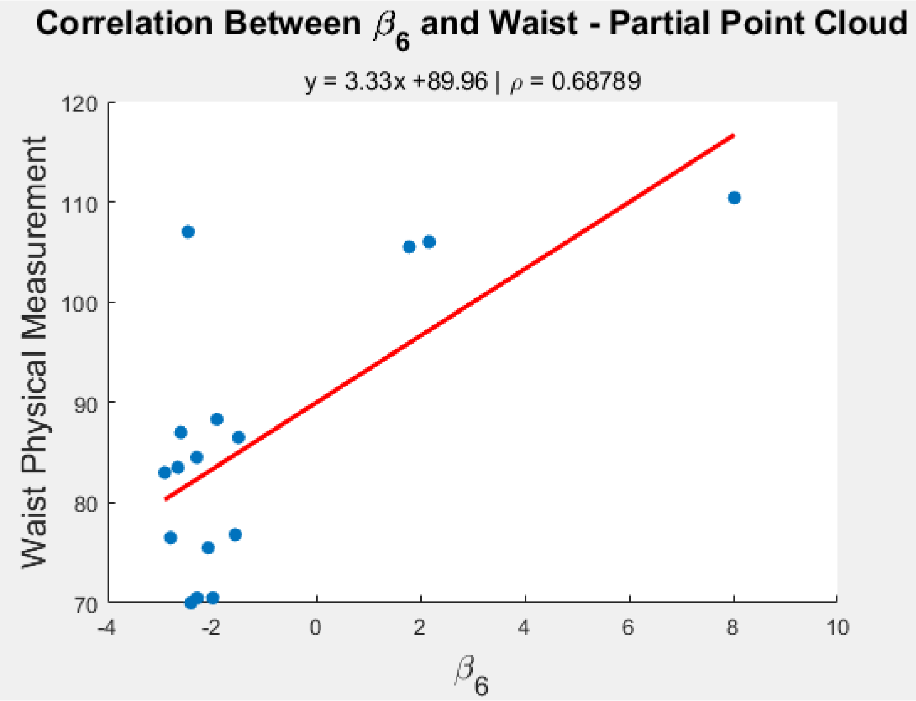


Figure C.9 Correlations visualized between β6 found using the partial point cloud and the waist measurement.

Table C.2 Correlations between SMPL shape parameters found with the partial point cloud and physical anthropometric measurement – outlier removed (N=15). Strong correlations (>0.6) are indicated in red

	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
Height	0.56	0.27	0.28	-0.18	0.28	0.50	0.16	0.15	0.02	0.13
Floor to Shoulder	0.57	0.30	0.23	-0.12	0.27	0.47	0.23	0.12	-0.03	0.10
Bicep Upper	0.36	-0.31	0.29	0.25	0.09	0.51	0.08	0.29	-0.19	0.10
Bicep Lower	0.43	-0.26	0.30	0.16	0.20	0.59	-0.05	0.25	-0.14	0.13
Forearm Upper	0.39	-0.20	0.36	0.04	0.31	0.65	-0.10	0.27	-0.08	0.05
Wrist	0.36	-0.19	0.19	-0.13	0.21	0.43	-0.12	-0.02	-0.14	-0.04
Armpit to wrist	0.39	0.43	-0.01	-0.33	0.15	0.29	0.40	-0.01	-0.13	-0.12
Thigh Upper	0.24	-0.35	0.47	0.48	0.14	0.44	0.03	0.37	-0.04	0.28
Thigh Lower	0.27	-0.33	0.47	0.32	0.20	0.66	-0.27	0.29	-0.10	0.29
Calf Upper	0.42	-0.21	0.55	0.19	0.31	0.82	-0.02	0.47	0.10	0.27
Calf Largest	0.39	-0.43	0.45	0.20	0.16	0.72	-0.10	0.34	0.02	0.34
Ankle	0.58	-0.17	0.47	-0.12	0.31	0.69	0.00	0.35	0.16	0.22
groin to floor	0.57	0.35	0.00	-0.02	0.19	0.26	0.25	-0.15	-0.10	0.00
Hips	0.25	-0.46	0.46	0.50	0.10	0.50	0.03	0.28	-0.15	0.20
Waist	0.34	-0.45	0.45	0.17	0.19	0.63	0.17	0.39	-0.11	0.00

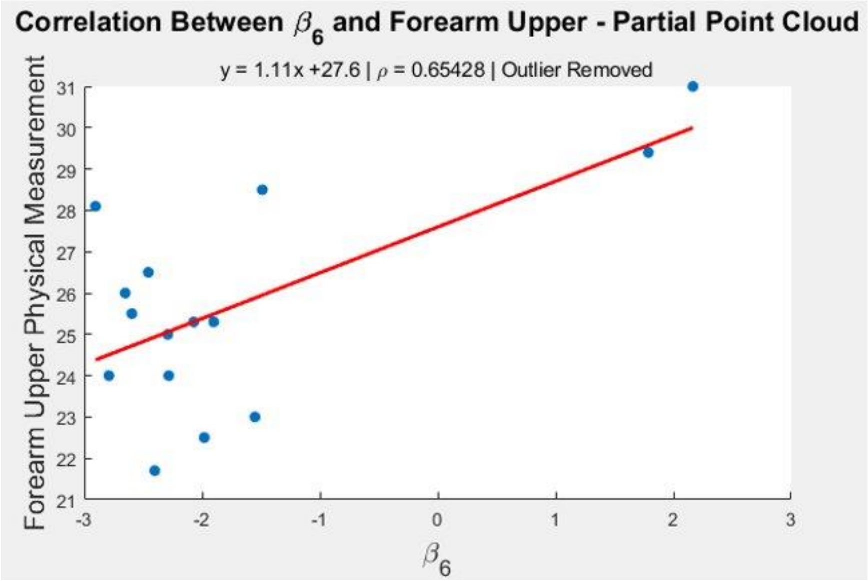


Figure C.10 Correlations visualized between β₆ found using the partial point cloud and the forearm upper measurement – outlier removed.

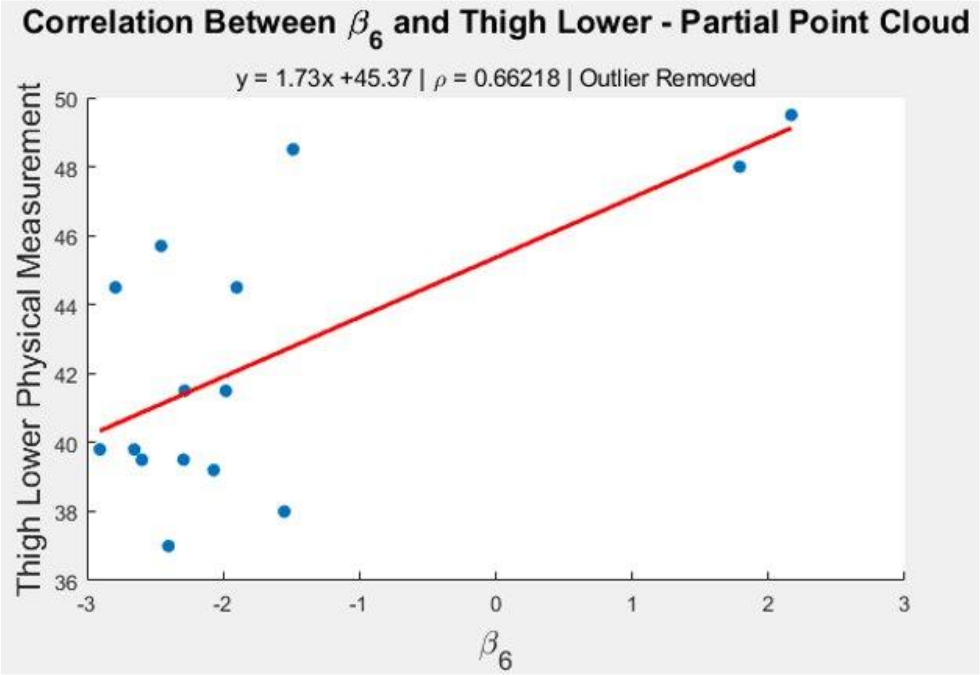


Figure C.11 Correlations visualized between β₆ found using the partial point cloud and the thigh lower measurement – outlier removed.

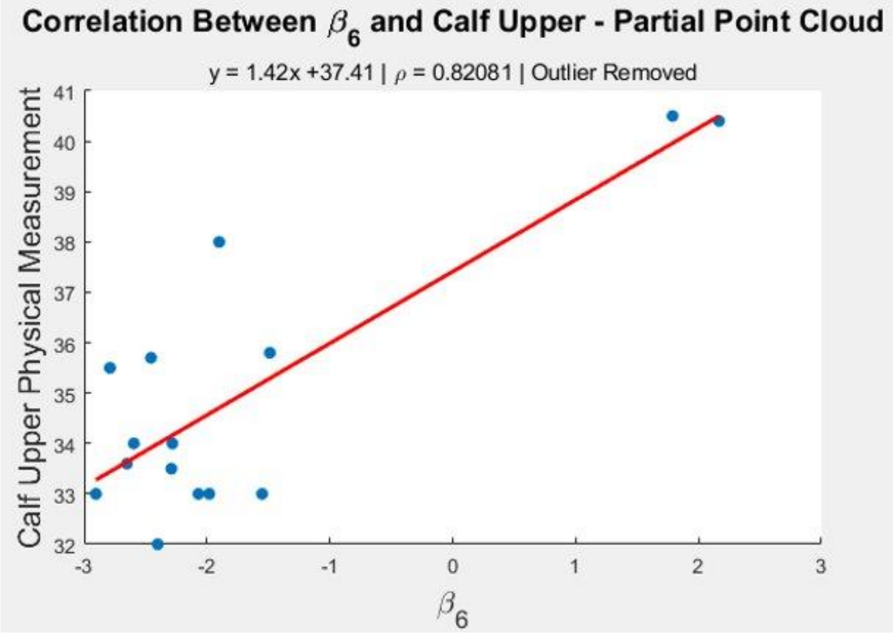


Figure C.12 Correlations visualized between β₆ found using the partial point cloud and the calf upper measurement – outlier removed.

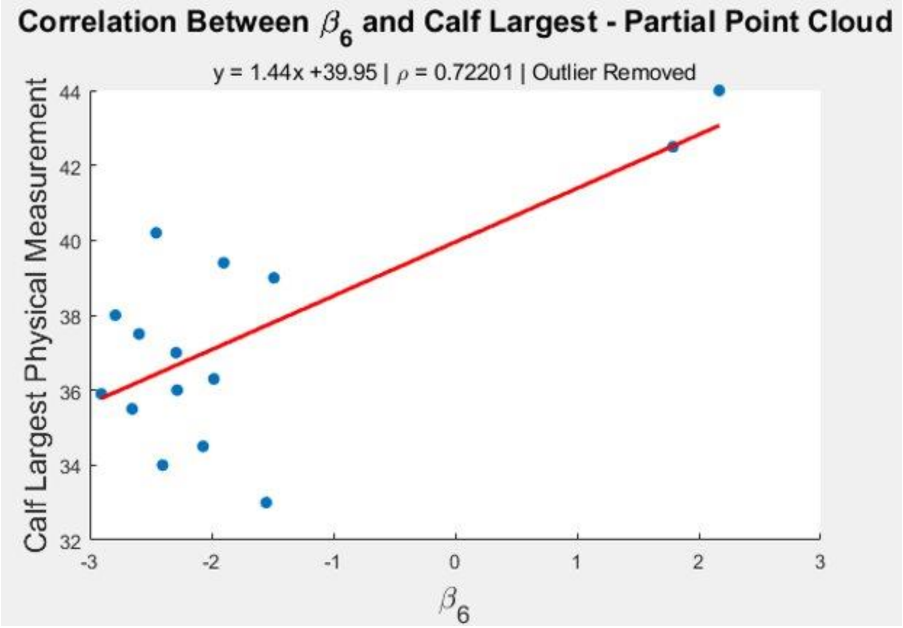


Figure C.13 Correlations visualized between β₆ found using the partial point cloud and the calf largest measurement – outlier removed.

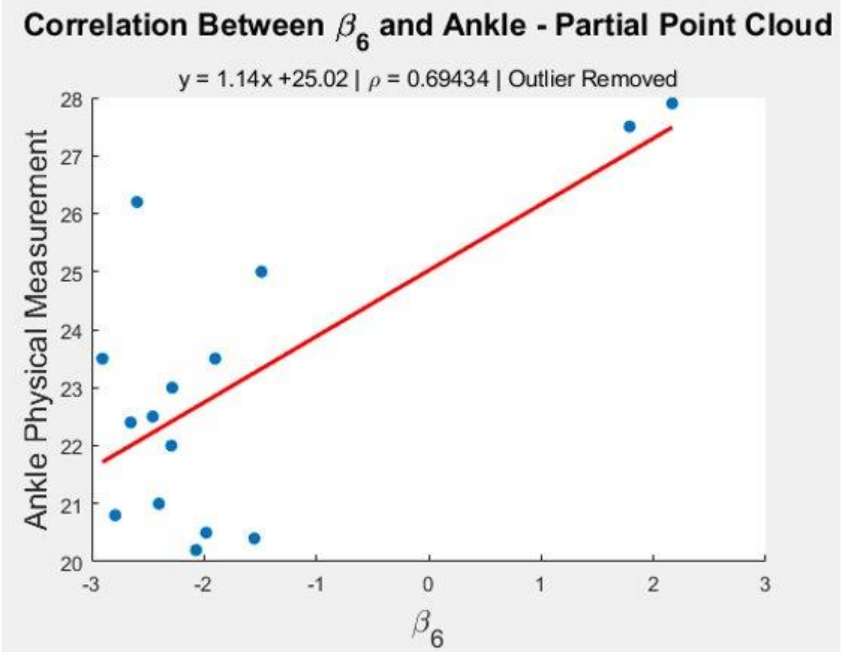


Figure C.14 Correlations visualized between β6 found using the partial point cloud and the ankle measurement – outlier removed.

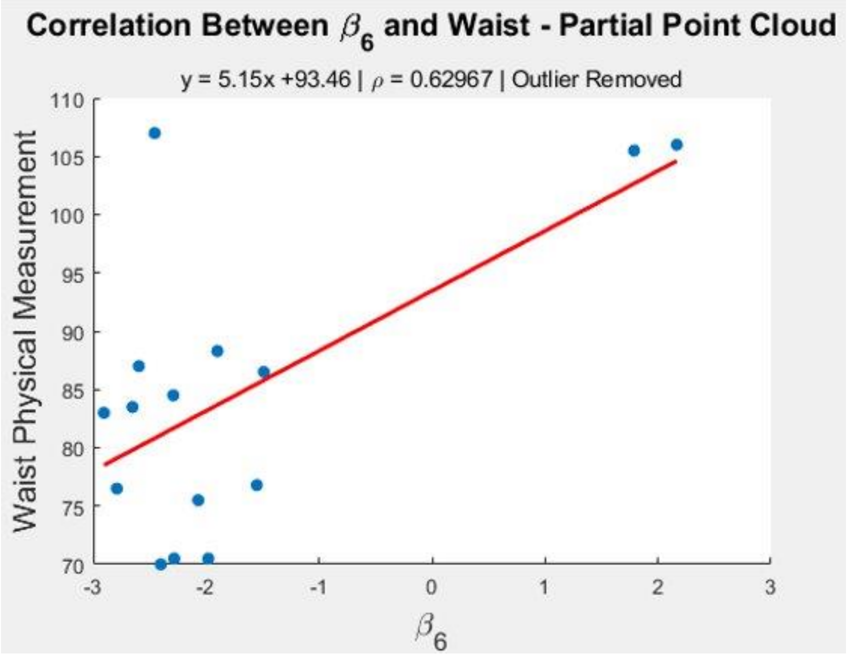


Figure C.15 Correlations visualized between β6 found using the partial point cloud and the waist measurement – outlier removed.

APPENDIX D. RELATED PUBLICATIONS

E. Young, A. Eustace, C. Myers, and P. Laz, “Computational modeling of human shape and pose,” presented at the Rocky Mountain American Society of Biomechanics Regional Meeting, Estes Park, CO, United States, April 8-9, 2022.

E. Young, C. Myers, A. Eustace, P. Rullkoetter, P. Laz, “Novel approach for non-invasive prediction of body shape and habitus,” presented at the Orthopaedic Research Society 2023 Annual Meeting, Dallas, TX, United States, February 10-14, 2023.