**sA Project Report**

**On**

**3D Human Body Measurements**

Under the supervision of

Dr. Ankita Jain (ASSISTANT PROFESSOR)

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS OF:

THIRD YEAR PROJECT

**MAHINDRA UNIVERSITY**

**ECOLE CENTRALE SCHOOL OF ENGINEERING**

**HYDERABAD**

**(JUNE 2023)**

**ACKNOWLEDGMENTS**

We would like to share our sincere gratitude to all those who have helped us in the completion of this project. During the course of the project, we faced many challenges, and the following people helped us overcome all the difficulties, to reach the final compilation of our idea into an actual work in progress.

We would like to thank our Project Coordinator and Mentor, Dr. Ankita Jain ma’am, for her continuous help and monitoring during the project work.

We would like to thank the management of Mahindra University - Ecole Centrale School of Engineering for providing us with a chance to practically implement our learnings and exhibit our skills beyond the classroom.

Sincerely,

CHAITANYA SRIKANTH (SE20UARI038)

PRADHAM MUMMALETI (SE20UARI116)

YERRABELLY SUMANTHRA RAO (SE20UECE093)

CONTENTS

Title Page……………………………………………………………………....1

Acknowledgments…………………………………………………………….2

Abstract……………………………………………………………………..….4

Chapter I: Introduction………………………………………………………..5

Chapter II: Problem Definition……………………………………………….6

Chapter III: Background and Related works ………………………………7

Chapter IV: Implementations

4.1 Generating 3D Model………………………………………………...8

4.2 Capturing 2D measurements from image ………………….……...8

4.3 Calculating measurements of the body parts……………………..9

4.4 To estimate the bisection plane of the 3D model…………….....10

ChapterV:Results……………………………………………………………12

ChapterVI:Conclusions……………………………………………………..13

ChapterVII:References……………………………………………………..14

**ABSTRACT**

Accurately calculating 3D body measurements from 3D models is crucial for industries including fashion, fitness, virtual reality, and healthcare. This abstract provides an overview of the process involved in segmenting the body, slicing the model, subdividing the model, estimating edge lengths, and highlights the implementation using Python, as well as their applications.

The process begins with data acquisition, where 3D models are obtained using Kinect sensor and images. In this study, a Python code implementation is utilized for processing the 3D scans. The code extracts the necessary geometric information and converts it into a usable 3D model representation.

Following data acquisition, the body is segmented from the 3D model to isolate relevant regions for accurate measurement extraction. Segmentation of body parts in the 3D model was done through with formulae determined during our work. We segmented the body parts based on the height of the 3D model.

The model may be subdivided, wherein the surface is divided into smaller elements, such as triangles or quadrilaterals. Python-based subdivision algorithms, utilizing packages like Blender or custom implementations, enhance measurement accuracy by capturing finer details and improving the representation of complex body shapes.

Landmarks are then identified on the segmented, sliced, or subdivided model, acting as reference points for measurement extraction. To extract measurements based on the relative positions of these landmarks, incorporating edge lengths obtained from the slicing and subdivision steps.

To ensure measurement accuracy, an assessment is conducted, comparing the calculated 3D body measurements with ground truth data or employing statistical analysis techniques. Python libraries such as pandas or scikit-learn can be utilized for statistical analysis and validation of the obtained measurements.

**INTRODUCTION**

This report details the design and development of a novel approach for generating a 3D model of an exoskeleton suit using 2D images of a human. The primary objective of this project was to remove the background from the images and utilize computer vision techniques to extract features necessary for creating an accurate and detailed 3D model. The implementation was carried out using Python 3 within the Google Colab environment.

To accomplish the task of creating the 3D model from the images, PIFuHD:Multi-level Pixel-Aligned Implicit function for High-Resolution 3D Human Digitization (CVPR 2020) paper was used as reference. The code was taken from their github repository.

To accomplish the task, several libraries and frameworks were employed. The Json library facilitated the handling of configuration and metadata information stored in JSON files.

For image processing operations, the skimage (scikit-image) library was utilized. Skimage offers a collection of algorithms for advanced image processing and analysis.

The project consisted of two tasks: Predict limb lengths and circumferences of various human body parts.

To predict the limb length, “mediapipe” library available on python3 was used to accomplish the task. This library helped us to find the anchor points on the human body from the image captured. All the points acquired were used to predict the pixel distance, later converted to real life metrics using our estimations.

Key numerical computations and data manipulation were performed using the numpy library, which is fundamental for handling arrays and matrices. The Blender Python API’s were used to segment the 3D model. With the help of “bpy” library available in Python3, we could manipulate the model orientations and the model structure for the task.

Our key idea during the project is to find the circumferences and girths of various human body parts. For accomplishing this task, acquiring a 3D model would be crucial to get high accuracy while predicting the measurements. After acquiring the 3D model, many calculations were done to correctly predict the number of edges present in a plane and estimate their edge lengths according to the ground truth and predict the results.

**PROBLEM DEFINITION**

The key idea of the project is to extract human body measurements automatically for the adjustment of the exoskeleton suit.

The two primary problem statements are:

1. Extract limb lengths from an image of a human body as the subject.
2. Extract circumferences of various human body parts such as chest, waist, biceps, wrist, etc.

This valuable information can serve as a critical resource for the Exoskeleton Suit team, aiding them in fine-tuning measurements and optimizing the design of their suit to maximize its benefits across diverse applications. By incorporating these precise body measurements, the team can ensure an optimal fit, enhanced functionality, and improved user experience, ultimately contributing to the widespread adoption and effectiveness of the Exoskeleton Suit technology.

The automatic determination of measurements using computer-based techniques holds immense potential for the textile industry. By leveraging advanced technologies, such as computer vision and machine learning, to extract accurate body measurements, textile companies can revolutionize their design, production, and customization processes.

**RELATED WORK**

Height measurement by smartphone (MDPI, 2022)

This article introduces a new method to measure height using a smartphone. In this method, two silhouette images of the subject, front and side, are used to reconstruct a 3D model of her in the body. The 3D model is then used to calculate the subject's body size. The authors evaluated the accuracy of the method by comparing it with conventional anthropometry. They found their method to be as accurate, more convenient, and less time consuming than traditional methods.

A new smartphone-based height measurement method (ResearchGate, 2021)

In this article, we present a novel height measurement method using smartphones that does not require additional reference to an object of known size. In this method, body proportions and machine learning techniques are used to determine measurement locations, and measurements from two silhouette images are used to perform her 3D reconstruction of the body. The authors used methods to measure height (that is, waist, lower hip, and thigh circumference) for men and women to select well-fitting pants.

NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis

NeRF, or Neural Radiance Field, is a new approach to representing 3D scenes that can be used to synthesize novel views of a scene from a variety of viewpoints. NeRF is a volumetric representation of a scene, meaning that it represents the scene as a continuous function of position and direction. This allows NeRF to represent scenes with complex geometry and appearance, such as those found in natural environments.

**IMPLEMENTATION**

During the ideation phase of the project, we tested out different methodologies to retrieve the 3D model. Firstly, we used a kinect sensor to capture the 3D model of the human body. After sometime, we used the code that we found online to generate the 3D model from an image.

**4.1 Generating the 3D model.**

1. **Retrieving 3D model from the Kinect sensor.**

Two primary software tools were utilized for the process, namely Kinect for Windows Developer Toolkit v1.8.0 and Skanect software. The 3D model was processed using these tools by manually removing background objects. However, several challenges were encountered during the capture of the human body's 3D model. The calibration of the Kinect sensor proved to be particularly problematic, as even minor deviations during the capture process resulted in significant noise within the 3D model.

The Kinect sensor operates by capturing the 3D model as the object of interest rotates in a full 360° motion. Therefore, while capturing the 3D model, it was crucial for the person to rotate at a consistent pace to ensure accurate representation. Any disruptions in the motion would introduce noise into the resulting 3D model.

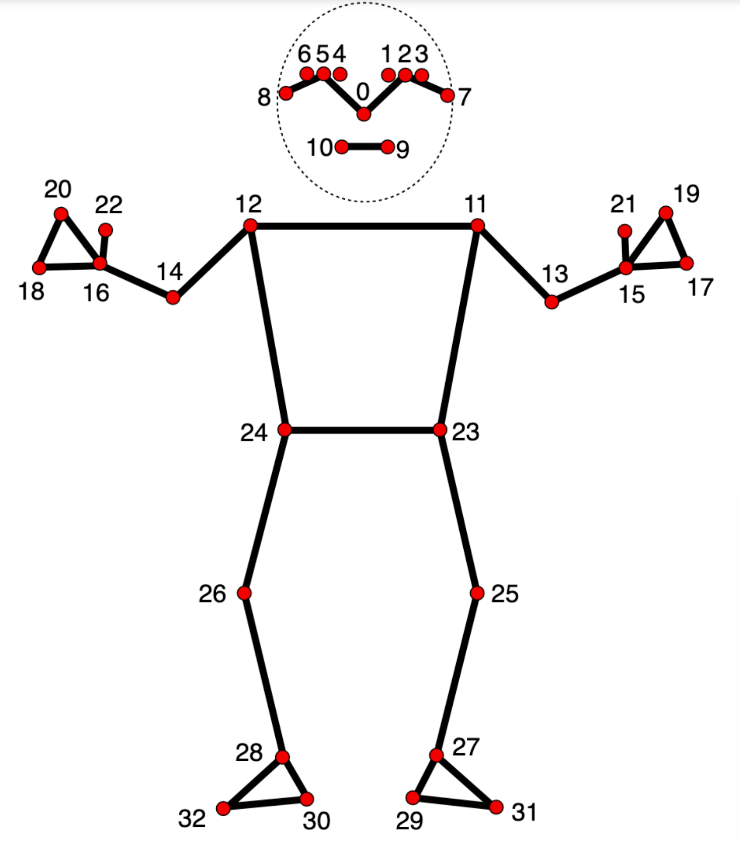
In order to address these issues, an online GitHub repository code was employed to generate the 3D model from an image, effectively resolving the aforementioned challenges.

1. **Generation of 3D model from the github repository, PIFuHD: Multi-Level Pixel-Aligned Implicit Function for High-Resolution 3D Human Digitization.**

By using this code, we generated the 3D models which we used to train in our algorithm.This code generates the 3D model from an image without any background. Online websites remove.bg were used to generate the image without background. This code uses libraries like PyTorch, json, PIL, skimage, cv2 to generate the 3D model.

**4.2 Capturing 2D measurements from image.**

During this phase of the project, we employed the MediaPipe "Pose landmark detection" module to identify the specific points of interest, known as landmarks, on our body. This allowed us to accurately determine the lengths of various limbs. The accompanying image provides a visual representation of these landmarks.



Utilizing the relative coordinates of the anchor points obtained from the mediapipe library, we proceeded to denormalize these points. The denormalization process involved adjusting the coordinates based on the camera resolution of the video recording, specifically (640 x 480 pixels).

The denormalization of the points was performed using the following formula:

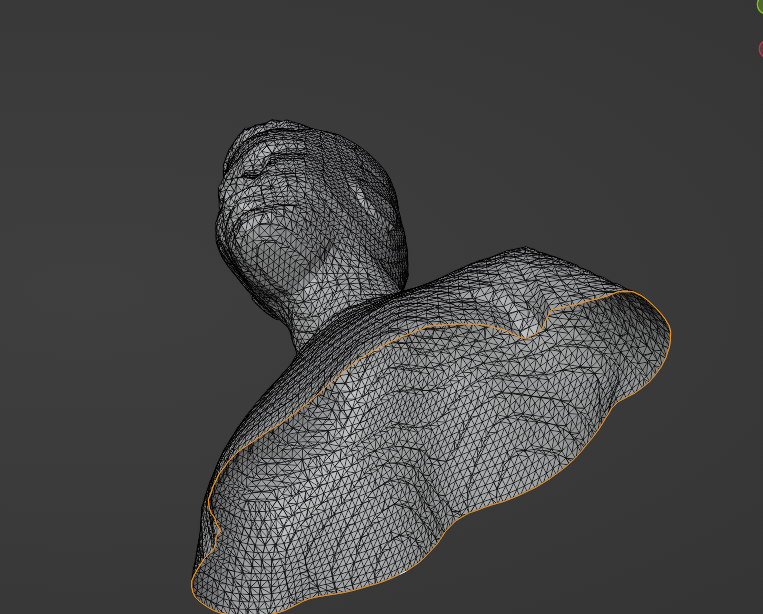
Xoriginal = Xnormalized \* Xcamera

In order to calculate the pixel distance between two coordinates, we made use of the known length of 640 pixels from the background length captured in real life by the image. By establishing this reference length, we were able to determine the equivalent length of a single pixel.

**4.3 Calculating measurements of the body parts from the 3D model**

To accomplish the required calculations, manipulations, and determination of human body part circumferences, we employed Blender software along with the bpy package, which is accessible through Python.

To derive the circumferences, we performed a bisection of the model at various levels, including the chest, waist, shoulders, and arms. This allowed us to obtain the circumference by assessing the number of edges present at the same plane and each edge length on blender coordinates. To calculate the lengths, euclidean distance formula was used to calculate each edge distance present on the same z-plane.



The orange highlighted part is the result of bisection of the 3D model. There are ‘n’ edges on the same plane. All the edge lengths have been calculated by the formula

Edge\_length = root[(x1-x2)^2 + (y1-y2)^2 + (z1-z2)^2]

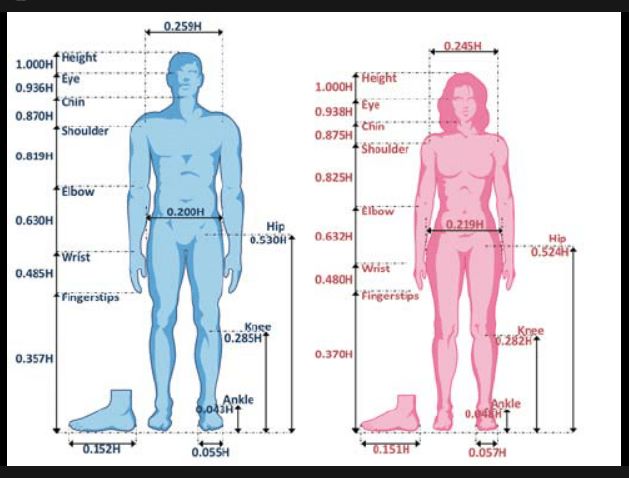
After calculating these edge lengths, we have averaged each blender measurement metric with original distance and came up with an optimal number to measure the distance in real life metric (inches).

Edge\_length(blender metrics)\* 30.384392 = Real life metric circumference (in inches)

With this formula, we calculated one edge length and predicted the measurements for other 3D models.

**4.4 To estimate the bisection plane of the 3D model.**

The image provided below served as our reference point for the calculations. By utilizing the body's height, we determined the appropriate plane to bisect the model at specific points, enabling us to accurately measure the circumferences of different body parts. The entire process was automated using the bpy library in Python, which offered comprehensive functionality support.



**RESULTS**

The outcomes achieved through our project encompass a diverse range of functions specifically designed to determine measurements of various body parts. These meticulously crafted functions form an integral part of our project documentation, offering a comprehensive toolkit for accurately assessing and quantifying dimensions related to the human body. So far, we have obtained the functions for:

- Waist

- Chest

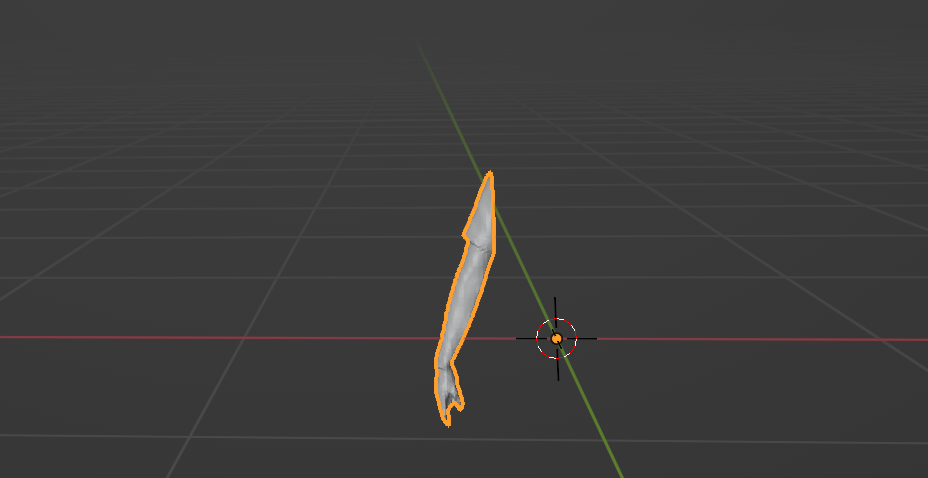
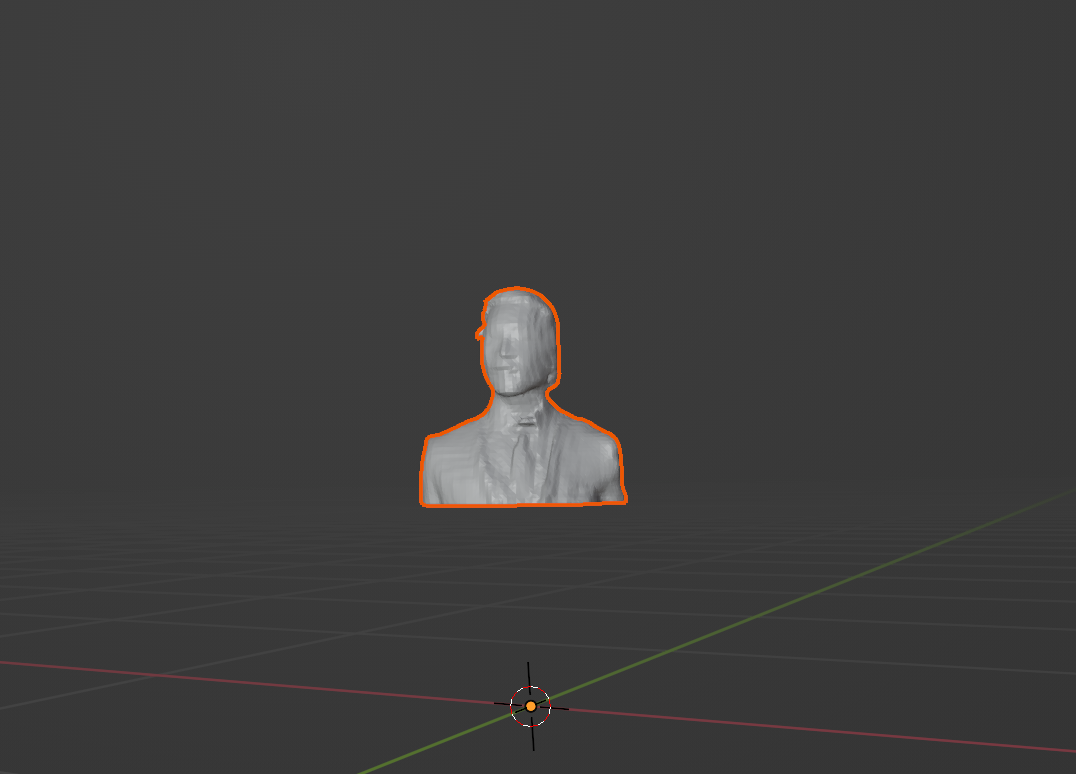
- Left bicep

- Right bicep

- Shoulders

-Wrist

The measurements are in terms of (inches) with a fluctuating error of 5% at most.

**CONCLUSION**

To conclude, extracting 3D measurements from 3D models in an automated manner is a significant need in the industrial field, particularly in fields related to fashion and health care. Hence, we have obtained body measurements of a given 3D model in an automated manner using Blender and its respective Python API. We first started by extracting a 3D model of a body from a picture. Then, after importing the model on Blender we were able to segment the body parts using the preordained ratios the human body is divided into with respect to the height of the person. As the human body is not limited to just 2D measurements, we formulated a function which would calculate the circumference of the waist and chest. In the end, we obtain a marginal error of 5%, which fluctuates with each model and the function used. It can be argued that measurements taken by humans tend to be slightly more accurate in general. However, an automated method of extracting measurements can save time and store the measurements efficiently. In the future, an idea we have thought of to decrease the marginal error is to implement a machine learning model on which we can use error analysis to further minimize the error.

**REFERENCES**

[**https://shunsukesaito.github.io/PIFuHD/**](https://shunsukesaito.github.io/PIFuHD/)

[**https://developers.google.com/mediapipe/solutions/vision/pose\_landmarker**](https://developers.google.com/mediapipe/solutions/vision/pose_landmarker)

**Research Papers:**

[**https://www.researchgate.net/publication/328224713\_Automatic\_Extraction\_of\_3d\_Body\_Measurements\_from\_2d\_Images\_of\_a\_Female\_Form**](https://www.researchgate.net/publication/328224713_Automatic_Extraction_of_3d_Body_Measurements_from_2d_Images_of_a_Female_Form)

[**https://www.researchgate.net/publication/359465937\_Estimate\_human\_body\_measurement\_from\_2D\_image\_using\_computer\_vision**](https://www.researchgate.net/publication/359465937_Estimate_human_body_measurement_from_2D_image_using_computer_vision)