Chapter-1

INTRODUCTION

1.1 Introduction

3D Body Measurement System Playing vital role in the current Fashion World. The Body measurement system is developed in an android platform and also on the Web. The Body Measurement system is easy and designed with a user friendly interface.The main purpose of this Body measurement system in android or Web is to help Everyone to take their Body measurements Easily By taking a photo in a standard position without any physical measuring tools. We have use NOMO-3D-400-Scans as a dataset 194 male and 181 female subjects to train this model.

1.2 Statement of the problem

Generally we use to follow traditional method to measure body sizes , but it is very embarrassing and time consuming process . So to overcome these problems we proposed a system which take 3D images as input source and predicts the measurements as output in centimetres by using deep learning with python .

1.3 Objective

The main aim of this project is to obtain the human body measurements without any physical measuring tools. This gives the best Garment Fit for humans. To achieve this we are using Deep learning with python.

1.4 Goals

- The main reason behind the returns in on-line shopping sites are : we are not able to know our body measurements, hence there is a need to upgrade from the existing system to the new system, that’s why we created a body measurement system that gives the accurate measures of the body.
- By this we can easily identify the best garment fit for our body.
- This model reduces the return policy in on-line shopping.
- With this model user can easily identify his/her body measurements.

1.5 Scope

The Scope of this body measurement system includes the following:

- Every E-commerce websites can easily use it.
- Every user can use it to take the body measurements.
- we can use it in any off-line shopping malls

1.6 Applications

- E-comerce applications
- Medical field
- Fitness centers
- Tailoring

1.7 Limitations

- No privacy to user sizes
- Limited applicability
- Awarness of technology is needed
- Internet is included

Chapter - 2

LITERATURE SURVEY

2.1 Collect Information

We have taken the information from sources like IEEE papers and Git hub. We proposed these for ourselves. Deep learning techniques have been applied to this model to develop and predict the body measurements. We have collected the data set from NOMO-3D-400-Scans , 194 male and 181 female subjects. We have observed the conversion of 2D image to 3D image conversion in the process.

Out of many measurement attributes we have chosen only 14 attributes based on the suggestion of chatgpt . Later on we have converted the measurements into CSV files.

2.2 Study

Key features in 3d body measurements :

A literature survey on 3D body measurement project would involve reviewing existing research on the topic, including methods, techniques, and algorithms used in 3D body measurement system. Some important areas to consider in the survey may include:

- Image detection and recognition techniques: This involves studying different techniques used for image processing and recognition of 3D image measurements using deep learning techniques.
- Regression Algorithms : This involves studying different regression algorithms such as CNN and neural network-based approaches, which are commonly used in image processing.
- Datasets: A survey of 3D body measurement system would require a dataset used in 3D body measurement system called NOMO-3D-400-Scans which include 194 male and 181 female subjects.
- Challenges: Understanding the challenges faced in 3D body measurement system such as conversion of 2D image to 3D image and processing of 3D image which consume more RAM storage.
- Recent research: The survey should also include recent research in the field such as the use of deep learning, data augmentation , Anthropometry.

2.3 Benefits

There are several benefits 3D Body Measurement System Project, some of which include:

- Easy way of taking body measurements
- Reduces return policy of E-commerce applications.
- Increased efficiency

2.4 Summary

3D Body Measurement System is a model which measures the sizes of the body based on the 3D object. It converts the 2D object to 3D object then processing the object for further processing.Here , we use Trimesh python library for loading the dataset of 3D image , we store the each object attributes values like height , waist , shoulder etc in array format then we adjust the array shape into one format and for easy calculations we normalize the numerical values of 3D images.We train the model with these numerical values by using Deep Learning Concepts . with the help of test data we got the accurate measurements.

Chapter -3

ANALYSIS

3.1 Existing system

- The existing system takes the measurements using physical tapes. It is very time consuming and involves more physical work. It is very annoying for a woman to take their measurements in this system.
- Another model 3d body measurement system using Open-CV and Tensor Flow which is not giving the accurate result .
- Another model for body size measurement system is 2D image with python which is not giving exact measurements.
- Above models are existed in Git Hub.

3.2 Disadvantages of Existing Methods

- Less accuracy
- More time consuming process
- More man power
- Background Complexity

3.3 Overview of the Proposed Approach

3D Body Measurement System is a model which measures the sizes of the body based on the 3D object. It converts the 2D object to 3D object then processing the object for further processing.Here , we use trimesh python library for loading the dataset of 3D image , we store the each object attributes values like height , waist , shoulder etc in array format then we adjust the array shape into one format and for easy calculations we normalize the numerical values of 3D images.We train the model with these numerical values by using Deep Learning Concepts (Convolution Neural Network-CNN) , Tensor Flow , Keras . with the help of test data we got the accurate measurements.

3.4 Advantages of proposed system

- Improved accuracy
- Fast Response Time
- Reduced costs
- Reduction of complexity
- Accesibility
- Scalability

3.5 System Requirements

Software Requirements:

- Visual Studio Code
- Jupyter note book.
- Go-ogle colab
- Html, CSS, Flask are required.
- Windows 10

Hardware Requirements:

- RAM: 12GB above
- Hard disk: 500 GB above

Chapter- 4

SYSTEM DESIGN

4.1 Design of the system

Unified Modelling Language (UML) was created in 1995 by using merging diagramming conventions used by three application development methodologies: OMT by James Rumbaugh, Objector y by Invar Jacobson and the Brooch procedure by using Grady Brooch. Previous to this time, these three amigos, together with a few dozen other practitioners had promoted competing methodologies for systematic program development, each and every with its possess system of diagramming conventions. The methodologies adopted a sort of cookbook sort of pushing a application task via a succession of life cycle stages, culminating with a delivered and documented software. One purpose of UML was once to slash the proliferation of diagramming techniques by way of standardizing on a original modelling language, as a result facilitating verbal exchange between builders. It performed that goal in 1997 when the (international) Object administration team (OMG) adopted it as a commonplace. Some critics don’t forget that UML is a bloated diagramming language written by means of a committee. That said, I do not forget it to be the nice manner to be had today for documenting object-oriented program progress. It has been and is fitting more and more utilized in industry and academia. Rational Rose is a pc Aided program Engineering (CASE) software developed by way of the Rational organization underneath the course of Brooch, Jacobson and Rumbaugh to support application progress using UML. Rational Rose is always complex due to its mission of wholly supporting UML. Furthermore, Rational Rose has countless language extensions to Ada, C++, VB, Java, J2EE, and many others. Rational Rose supports ahead and reverse engineering to and from these language ages. However, Rational Rose does now not aid some usual design tactics as knowledge drift diagrams and CRC cards, due to the fact that these will not be a part of UML. Considering that Rational Rose has so many capabilities it is a daunting task to master it. Happily, loads can be executed making use of only a small subset of these capabilities. These notes are designed to introduce beginner builders into making productive use of the sort of subset.

4.1.1 Class diagram

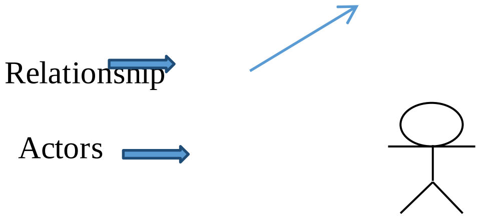
Class diagram in the Unified Modelling Language (UML), is a kind of static structure diagram hat describes the constitution of a process through showing the system's classes, their attributes, and the relationships between the class. The motive of a class diagram is to depict the classes within a model. In an object-oriented software, classes have attributes (member variables), operations (member capabilities) and relation

Fig1 Class Diagram

4.1.2 Use Case Diagram

It is a visually representation what happens when actor interacts with system. A use case diagram captures the functional aspects of a system.

The system is shown as a rectangle with name of the system inside ,the actor are shown as stick figures, the use case are shown as solid bordered ovals labeled with name of the use case and relationships are lines or arrows between actor and use cases. Symbols used in Usecase are as follows-



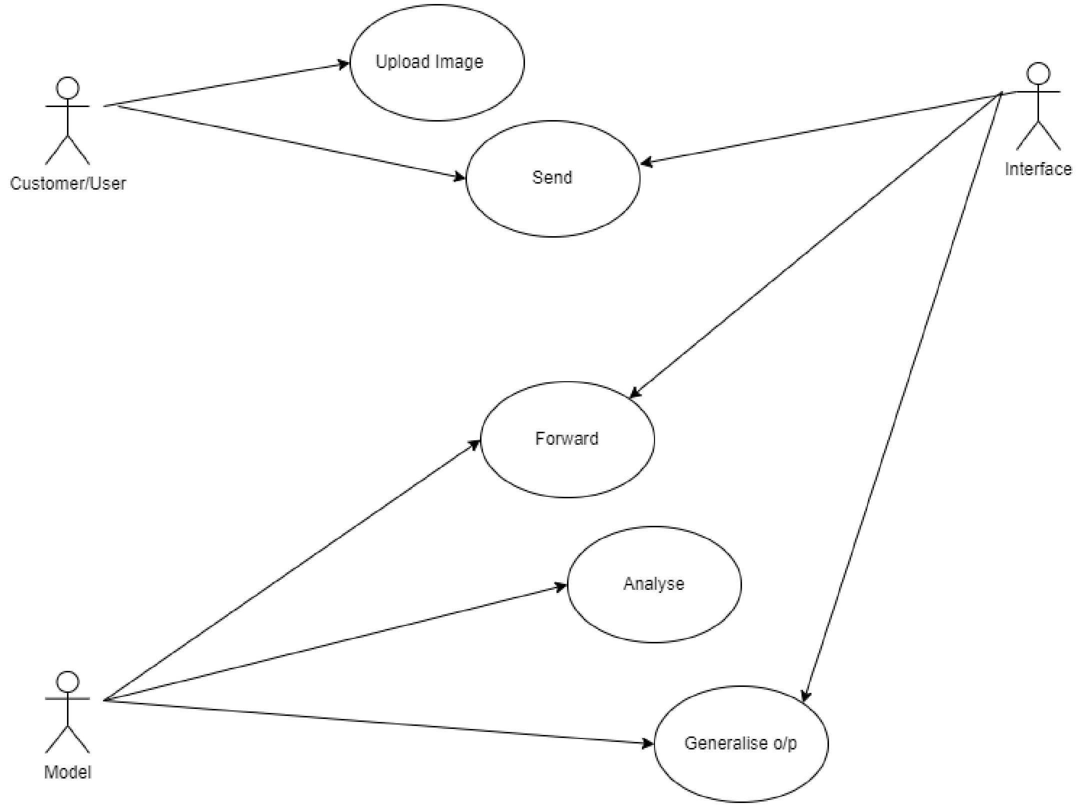
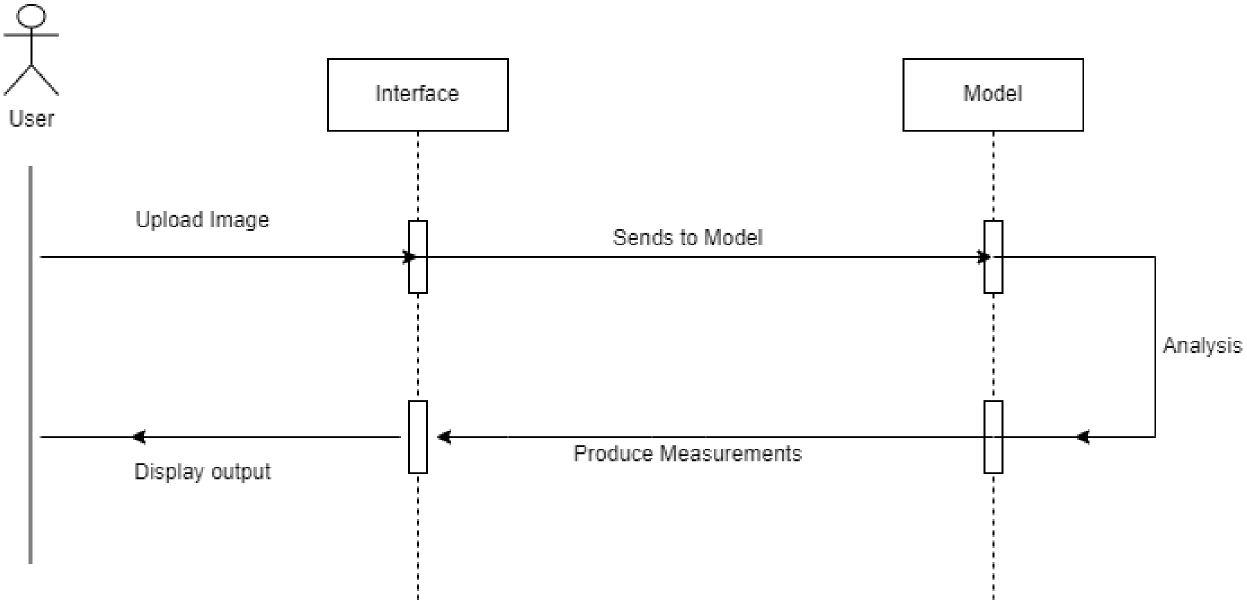


Fig 2 Use Case Diagram

4.1.3 SEQUENCE DIAGRAM

A sequence diagram in Unified Modelling Language (UML) is one variety of interaction diagram that suggests how methods operate with one other and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are quite often referred to as event-hint diagrams, event situations, and timing diagrams. A sequence diagram suggests, as parallel vertical traces (lifelines), special systems or objects that are residing at the same time, and, as horizontal arrows, the messages exchanged between them, within the order the place they occur.



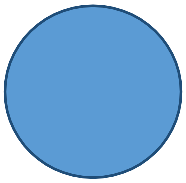
Sequence Diagram

4.1.4 DFD Diagram:

A data flow diagram or bubble chart (DFD) is a graphical representation of the "flow" of data .through an information system, modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kinds of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

The primitive symbols used for constructing DFD's are:Symbols used in DFD



A circle represents a process.

A rectangle represents external entity

A square defines a source or destination of the system data.

An arrow identifies data flow.

Double line with one end closed indicates data store

Level 0

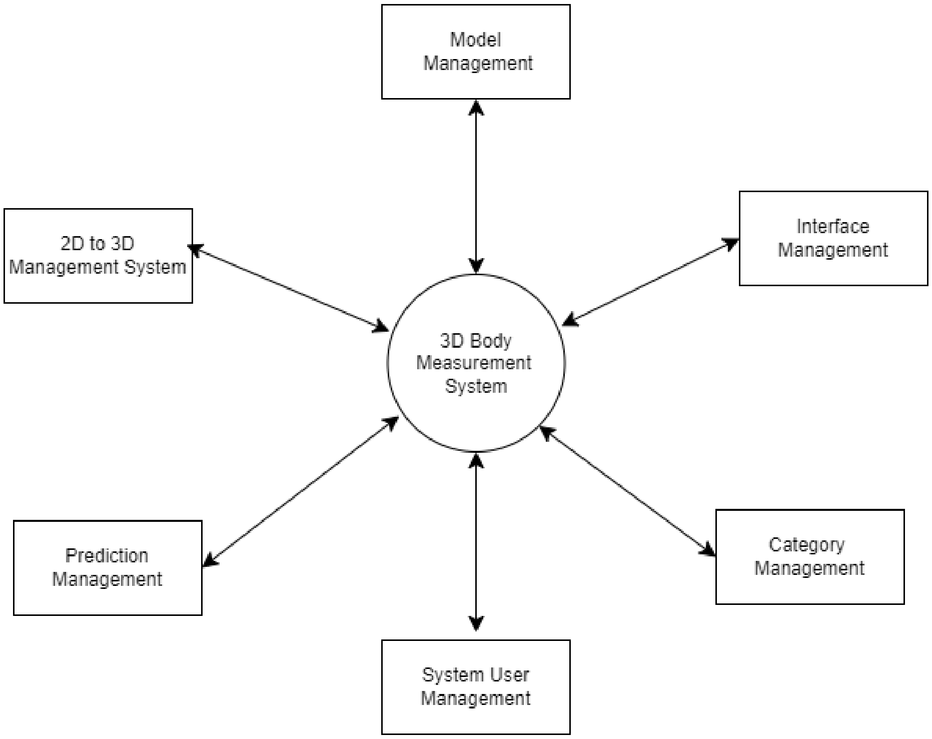


Fig 4 Level

Level 1

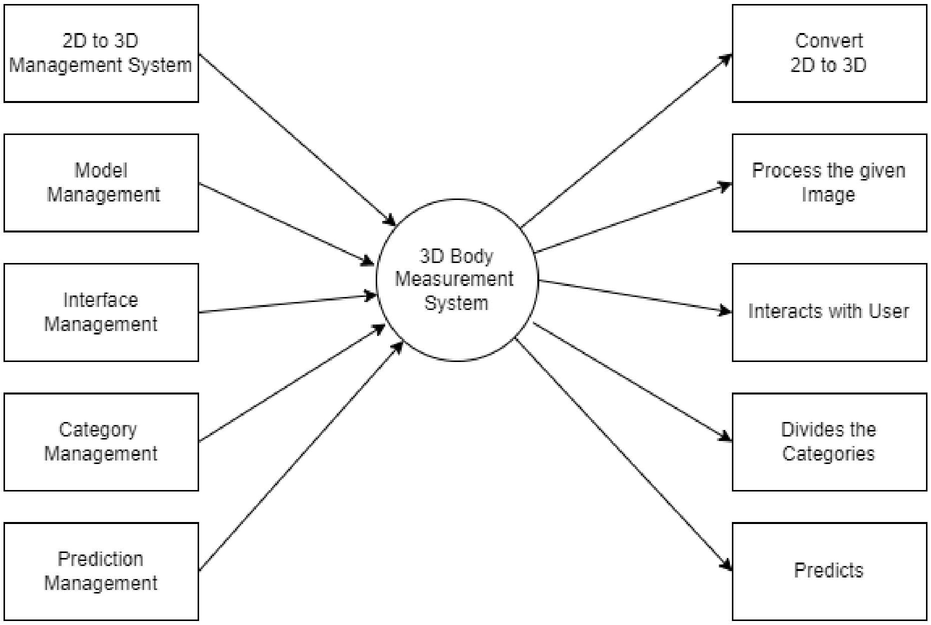


Fig 5 Level 1

Level 2

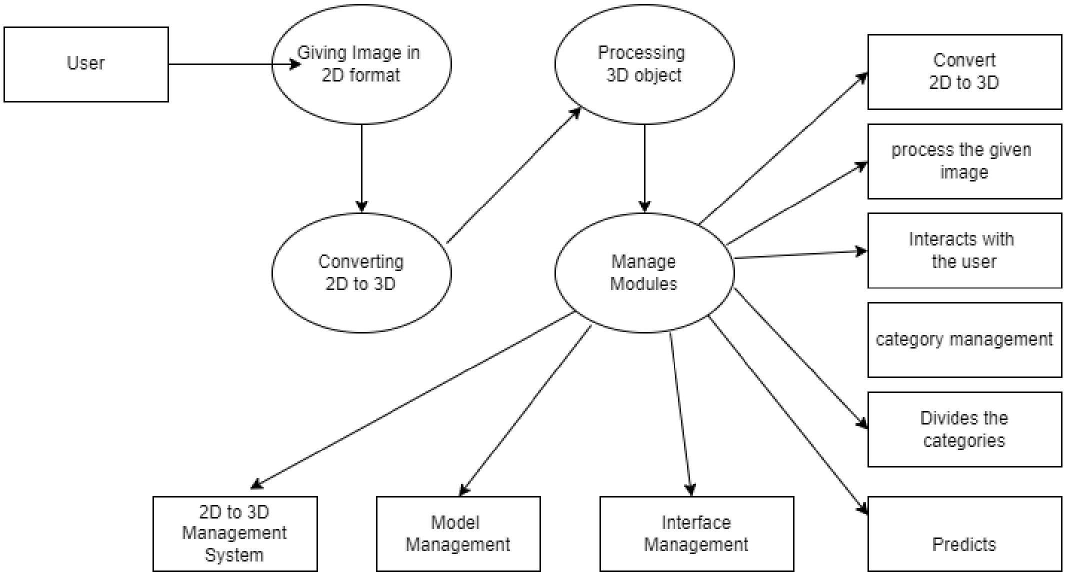


Fig 6 Level

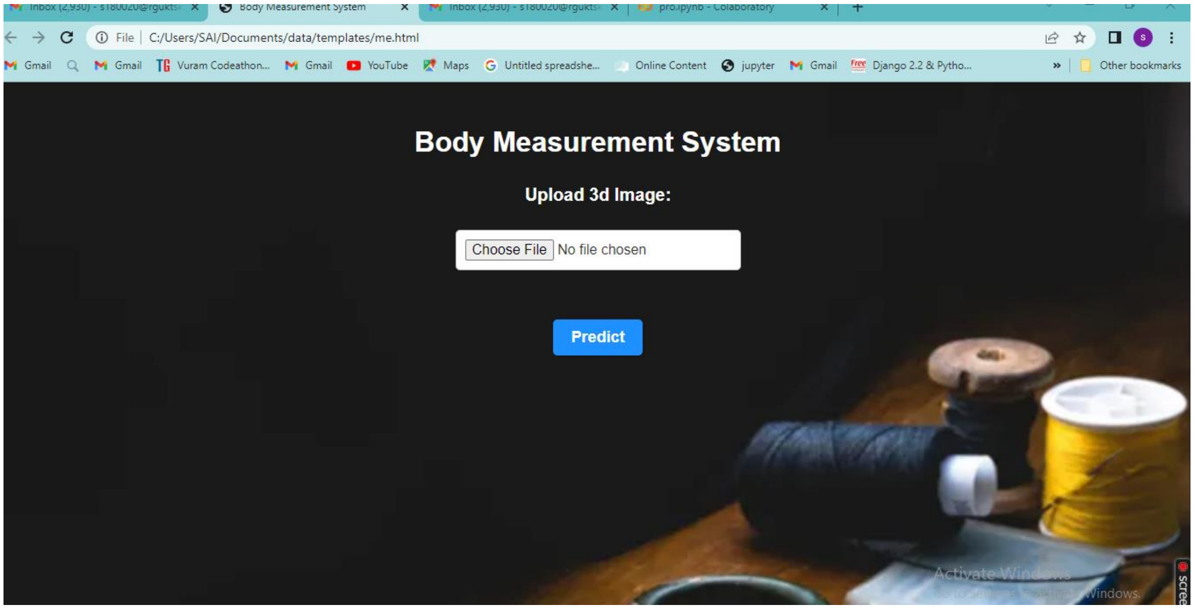
Chapter -5

SYSTEM IMPLEMENTATION

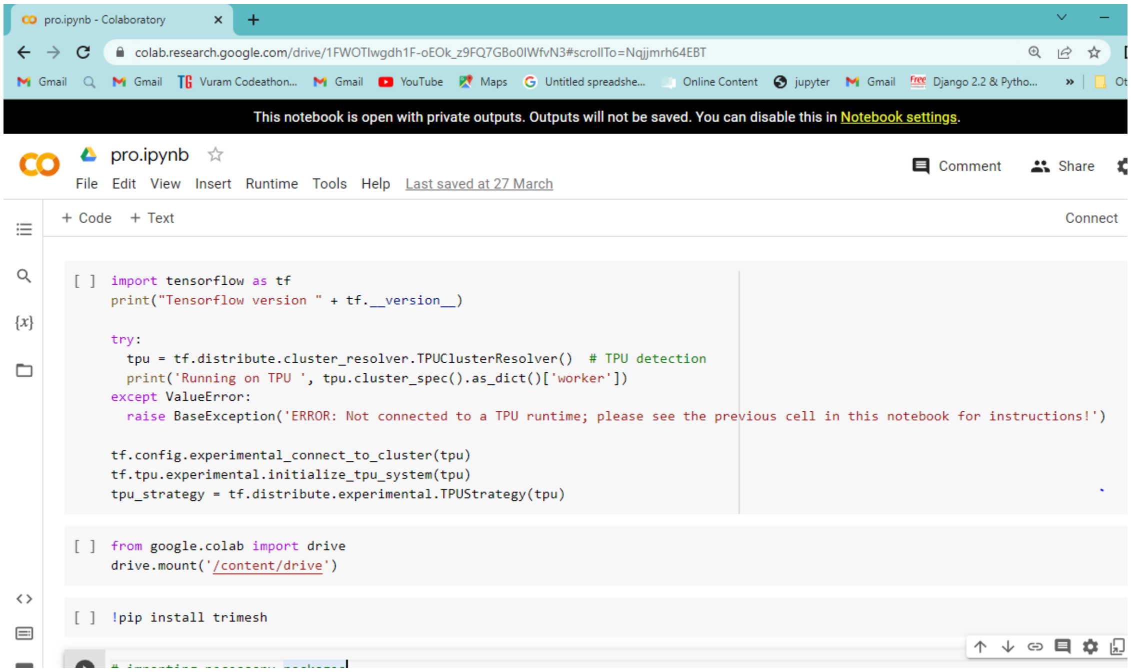
5.1 3D body measurement system

It is done by using HTML, CSS and python with deep learning concepts , we have used the necessary components to implement this. HTML and CSS are used for interface development and python with deep learning concepts for training the model . By using NOMO-3D-400-Scans which include 194 male and 181 female subjects is used to train the model.

Website over look



Google colab

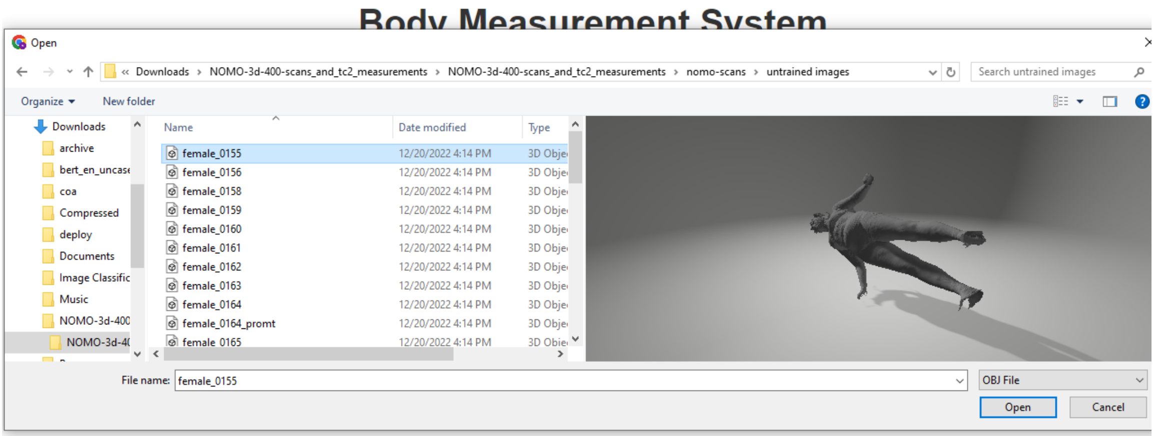


```
import tensorflow as tf
print("Tensorflow version " + tf.__version__)

try:
  tpu = tf.distribute.cluster_resolver.TPUClusterResolver()  # TPU detection
  print('Running on TPU ', tpu.cluster_spec().as_dict()['worker'])
except ValueError:
  raise BaseException('ERROR: Not connected to a TPU runtime; please see the previous cell in this notebook for instructions!')

tf.config.experimental_connect_to_cluster(tpu)
tf.tpu.experimental.initialize_tpu_system(tpu)
tpu_strategy = tf.distribute.experimental.TPUStrategy(tpu)
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
!pip install trimesh
```

Input

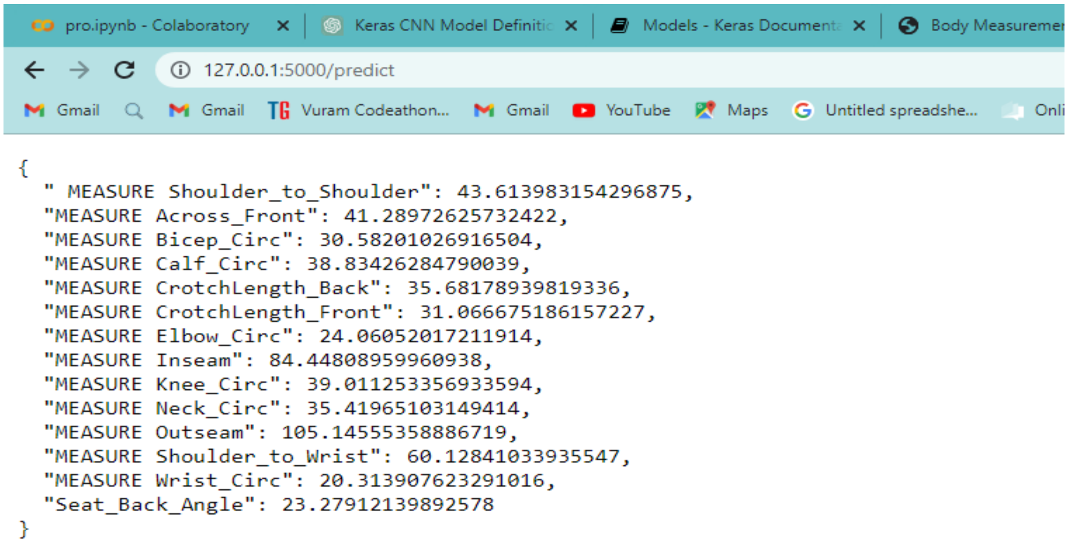


Name	Date modified	Type
female_0155	12/20/2022 4:14 PM	3D Obje
female_0156	12/20/2022 4:14 PM	3D Obje
female_0158	12/20/2022 4:14 PM	3D Obje
female_0159	12/20/2022 4:14 PM	3D Obje
female_0160	12/20/2022 4:14 PM	3D Obje
female_0161	12/20/2022 4:14 PM	3D Obje
female_0162	12/20/2022 4:14 PM	3D Obje
female_0163	12/20/2022 4:14 PM	3D Obje
female_0164	12/20/2022 4:14 PM	3D Obje
female_0164_promt	12/20/2022 4:14 PM	3D Obje
female_0165	12/20/2022 4:14 PM	3D Obje

File name: female_0155

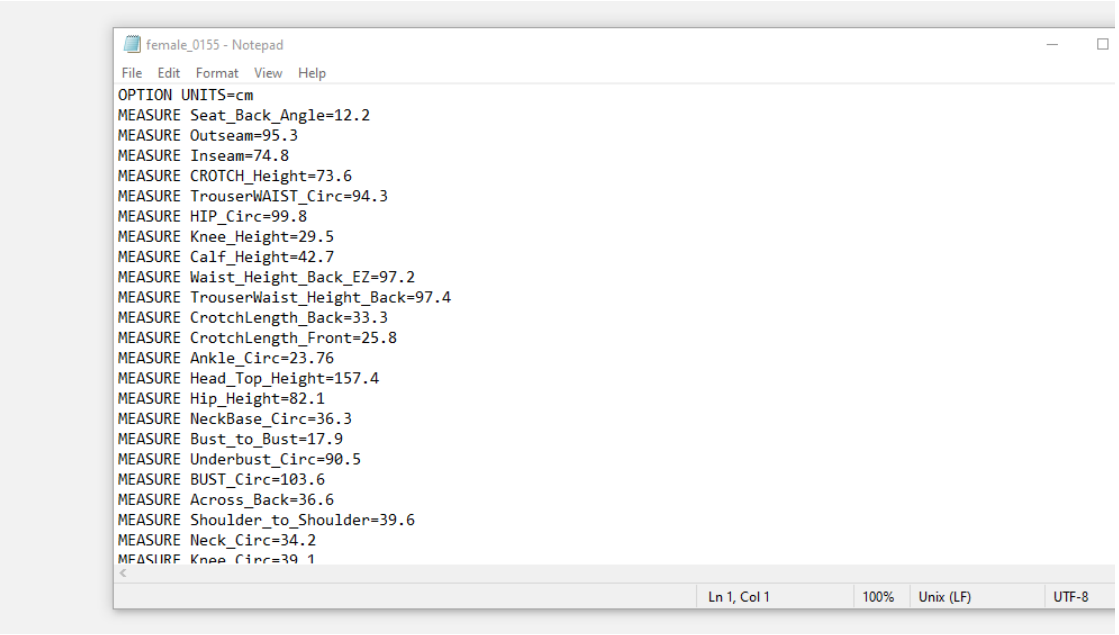
OBJ File

Predicted output:



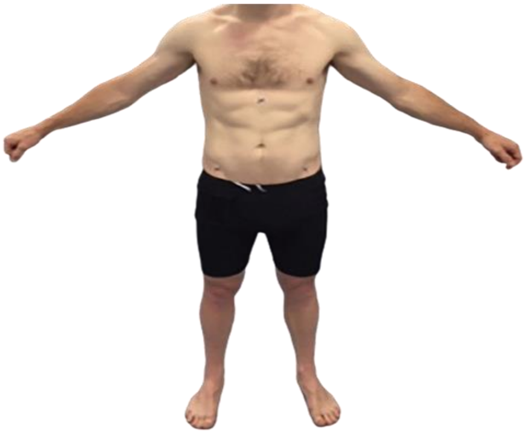
```
{
  " MEASURE Shoulder_to_Shoulder": 43.613983154296875,
  "MEASURE Across_Front": 41.28972625732422,
  "MEASURE Bicep_Circ": 30.58201026916504,
  "MEASURE Calf_Circ": 38.83426284790039,
  "MEASURE CrotchLength_Back": 35.68178939819336,
  "MEASURE CrotchLength_Front": 31.066675186157227,
  "MEASURE Elbow_Circ": 24.06052017211914,
  "MEASURE Inseam": 84.44808959960938,
  "MEASURE Knee_Circ": 39.011253356933594,
  "MEASURE Neck_Circ": 35.41965103149414,
  "MEASURE Outseam": 105.14555358886719,
  "MEASURE Shoulder_to_Wrist": 60.12841033935547,
  "MEASURE Wrist_Circ": 20.313907623291016,
  "Seat_Back_Angle": 23.27912139892578
}
```

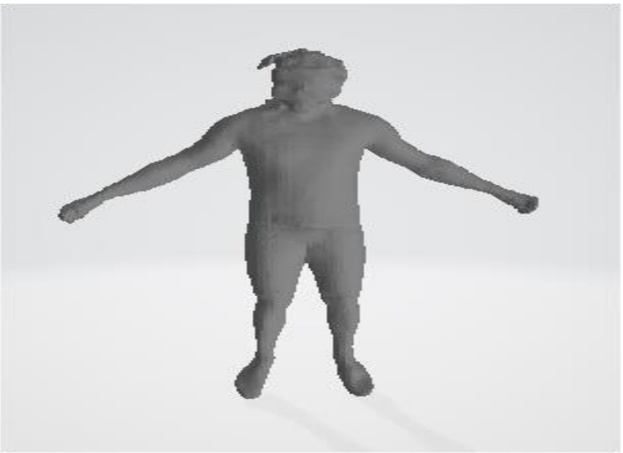
Actual measurements:



```
OPTION UNITS=cm
MEASURE Seat_Back_Angle=12.2
MEASURE Outseam=95.3
MEASURE Inseam=74.8
MEASURE CROTCH_Height=73.6
MEASURE TrouserWAIST_Circ=94.3
MEASURE HIP_Circ=99.8
MEASURE Knee_Height=29.5
MEASURE Calf_Height=42.7
MEASURE Waist_Height_Back_EZ=97.2
MEASURE TrouserWaist_Height_Back=97.4
MEASURE CrotchLength_Back=33.3
MEASURE CrotchLength_Front=25.8
MEASURE Ankle_Circ=23.76
MEASURE Head_Top_Height=157.4
MEASURE Hip_Height=82.1
MEASURE NeckBase_Circ=36.3
MEASURE Bust_to_Bust=17.9
MEASURE Underbust_Circ=90.5
MEASURE BUST_Circ=103.6
MEASURE Across_Back=36.6
MEASURE Shoulder_to_Shoulder=39.6
MEASURE Neck_Circ=34.2
MEASURE Knee_Circ=39.1
```

2D to 3D :





Chapter 6
SOURCE CODE

6.1 Training the model

```
import tensorflow as tf
print("Tensorflow version " + tf.__version__)


try:
  tpu = tf.distribute.cluster_resolver.TPUClusterResolver()  # TPU detection
  print('Running on TPU ', tpu.cluster_spec().as_dict()['worker'])
except ValueError:
  raise BaseException('ERROR: Not connected to a TPU runtime; please see the previous
 cell in this notebook for instructions!')


tf.config.experimental_connect_to_cluster(tpu)
tf.tpu.experimental.initialize_tpu_system(tpu)
tpu_strategy = tf.distribute.experimental.TPUStrategy(tpu)
from google.colab import drive
drive.mount('/content/drive')
!pip install trimesh
# importing necessary packages
import os
import trimesh
import numpy as np
# Female Data
# Reading Female(.obj) data and converting every image into Numerical Data and Storin
g them in list called array
a =[]
for image in os.listdir('/content/drive/MyDrive/Nomo/female_obj'):
   if image.endswith('.obj'):
      image_path = f"/content/drive/MyDrive/Nomo/female_obj/{image}"
      a.append(image_path)
 # .obj
 duplicate = ['/content/drive/MyDrive/Nomo/female_obj/female_0061_promt.obj', '/conte
nt/drive/MyDrive/Nomo/female_obj/female_0089_promt.obj',


 '/content/drive/MyDrive/Nomo/female_obj/female_0092_promt.obj', '/content/drive/My
Drive/Nomo/female_obj/female_0137_promt.obj', '/content/drive/MyDrive/Nomo/female
_obj/female_0164_promt.obj']
```

```
a = [b for b in a if b not in duplicate]
files = []
# iterate through all file
for file in os.listdir('/content/drive/MyDrive/Nomo/female_TC2Meas_txt/female_TC2Me
as_txt'):
  # Check whether file is in text format or not
    file_path = f"C:/Users/SAI/Downloads/NOMO-3d-400-
scans_and_tc2_measurements/NOMO-3d-400-scans_and_tc2_measurements/nomo-
scans/female_TC2Meas_txt/female_TC2Meas_txt/{file}"
    files.append(file_path)
files.sort()
# Extracting Vertices
a
ve=[]
for i in range(154):
  mesh = trimesh.load(a[i])
  vertices = np.array(mesh.vertices)
  rem_vertices = 62244-vertices.shape[0]
  if rem_vertices>0:
    for v in range(rem_vertices):
      vertices = np.append(vertices,np.array([[0,0,0]]),axis=0)
  ve.append(vertices)
import pandas as pd
df = pd.read_csv('/content/drive/MyDrive/Nomo/dataset.csv')
dupl = ['C:/Users/SAI/Downloads/NOMO-3d-400-
scans_and_tc2_measurements/NOMO-3d-400-scans_and_tc2_measurements/nomo-
scans/female_TC2Meas_txt/female_TC2Meas_txt/.ipynb_checkpoints','C:/Users/SAI/Do
wnloads/NOMO-3d-400-scans_and_tc2_measurements/NOMO-3d-400-
scans_and_tc2_measurements/nomo-
scans/female_TC2Meas_txt/female_TC2Meas_txt/female_0062_promt.txt','C:/Users/SAI
/Downloads/NOMO-3d-400-scans_and_tc2_measurements/NOMO-3d-400-
scans_and_tc2_measurements/nomo-
scans/female_TC2Meas_txt/female_TC2Meas_txt/female_0077_promt.txt', 'C:/Users/SA
I/Downloads/NOMO-3d-400-scans_and_tc2_measurements/NOMO-3d-400-
scans_and_tc2_measurements/nomo-
scans/female_TC2Meas_txt/female_TC2Meas_txt/female_0087_copy.txt"C:/Users/SAI/
Downloads/NOMO-3d-400-scans_and_tc2_measurements/NOMO-3d-400-
scans_and_tc2_measurements/nomo-
scans/female_TC2Meas_txt/female_TC2Meas_txt/female_0161_promt.txt','C:/Users/SAI
/Downloads/NOMO-3d-400-scans_and_tc2_measurements/NOMO-3d-400-
scans_and_tc2_measurements/nomo-
scans/female_TC2Meas_txt/female_TC2Meas_txt/female_0174_promt.txt','C:/Users/SAI
/Downloads/NOMO-3d-400-scans_and_tc2_measurements/NOMO-3d-400-
scans_and_tc2_measurements/nomo-
scans/female_TC2Meas_txt/female_TC2Meas_txt/female_0161_promt.txt', 'C:/Users/SA
I/Downloads/NOMO-3d-400-scans_and_tc2_measurements/NOMO-3d-400-
scans_and_tc2_measurements/nomo-
scans/female_TC2Meas_txt/female_TC2Meas_txt/female_0087_copy.txt'
]
# removing dupl rows from the dataframe
df = df[~df['filename'].str.contains('|'.join(dupl))]
female_data = {'MEASURE Seat_Back_Angle':df['MEASURE Seat_Back_Angle'], 'MEASURE CrotchLength_Front':df[ 'MEASURE CrotchLength_Front'], 'MEASURE CrotchLength_Back':df['MEASURE CrotchLength_Back'], 'MEASURE Knee_Circ':df['MEASURE Knee_Circ'], 'MEASURE Calf_Circ':df[ 'MEASURE Calf_Circ'], 'MEASURE Outseam':df['MEASURE Outseam'], 'MEASURE Inseam':df['MEASURE Inseam'], 'MEASURE Shoulder_to_Wrist':df['MEASURE Shoulder_to_Wrist'], 'MEASURE Bicep_Circ':df[ 'MEASURE Bicep_Circ'], 'MEASURE Elbow_Circ':df['MEASURE Elbow_Circ'], 'MEASURE Wrist_Circ':df['MEASURE Wrist_Circ'], 'MEASURE Shoulder_to_Shoulder':df[' MEASURE Shoulder_to_Shoulder'], 'MEASURE Across_Front':df['MEASURE Across_Front'], 'MEASURE Neck_Circ':df['MEASURE Neck_Circ']}


y1 = pd.DataFrame(female_data)
y11 = y1.head(154)
y1.shape
```

Male Data

```
# Reading Female(.obj) data and converting every image into Numerical Data and Storing them in list called array
b =[]
for image in os.listdir('/content/drive/MyDrive/Nomo/male_obj/male_obj'):
  if image.endswith('.obj'):
    image_path = f"/content/drive/MyDrive/Nomo/male_obj/male_obj/{image}"
    b.append(image_path)

# Extracting vertices from male data
b
```

```
# sorting the images
b.sort()
#duplicated images
n = [ '/content/drive/MyDrive/Nomo/male_obj/male_obj/male_0147_promt.obj', '/content
/drive/MyDrive/Nomo/male_obj/male_obj/male_0102_promt.obj', '/content/drive/MyDri
ve/Nomo/male_obj/male_obj/male_0100_promt.obj',
 '/content/drive/MyDrive/Nomo/male_obj/male_obj/male_0104_promt.obj',


 '/content/drive/MyDrive/Nomo/male_obj/male_obj/male_0111_promt.obj', '/content/driv
e/MyDrive/Nomo/male_obj/male_obj/male_0169_promt.obj', '/content/drive/MyDrive/N
omo/male_obj/male_obj/male_0122_promt.obj',


 '/content/drive/MyDrive/Nomo/male_obj/male_obj/male_0123_promt.obj',


]
imgs = [img for img in b if img not in n]
imgs.sort()
imgs[159]
arre = []
for i in range(159):
  mesh = trimesh.load(imgs[i])
  vertices = np.array(mesh.vertices)
  remaining_rows=62244-vertices.shape[0]
  for j in range(remaining_rows):
    vertices = np.append(vertices,np.array([[0,0,0]]),axis=0)
  arre.append(vertices)
arre = np.array(arre)
ve = np.array(ve)
array = np.concatenate((arre,ve),axis=0)
arre.shape
y1.shape
len(ve)
ve.shape
x = array
male = pd.read_csv('/content/drive/MyDrive/Nomo/file.csv')
male_data = {'MEASURE Seat_Back_Angle':male['MEASURE Seat_Back_Angle'], 'ME
ASURE CrotchLength_Front':male[ 'MEASURE CrotchLength_Front'], 'MEASURE Cro
tchLength_Back':male['MEASURE CrotchLength_Back'], 'MEASURE Knee_Circ':male[
'MEASURE Knee_Circ'], 'MEASURE Calf_Circ':male[ 'MEASURE Calf_Circ'], 'MEAS
URE Outseam':male['MEASURE Outseam'], 'MEASURE Inseam':male['MEASURE Ins
eam'], 'MEASURE Shoulder_to_Wrist':male['MEASURE Shoulder_to_Wrist'], 'MEASURE Bicep_Circ':male['MEASURE Bicep_Circ'], 'MEASURE Elbow_Circ':male['MEASURE Elbow_Circ'], 'MEASURE Wrist_Circ':male['MEASURE Wrist_Circ'], 'MEASURE Shoulder_to_Shoulder':male['MEASURE Shoulder_to_Shoulder'], 'MEASURE Across_Front':male['MEASURE Across_Front'], 'MEASURE Neck_Circ':male['MEASURE Neck_Circ']}
y2 = pd.DataFrame(male_data)
y22 = y2.head(159)
y = pd.concat([y11,y22],axis=0)
```

Normalization

```
# Normalizing the array values
x = x/x.max()
X
```

Creating a Convolutional Architecture

```
from keras.models import Sequential
from keras.layers import Conv2D, MaxPool2D, Flatten, Dense,BatchNormalization
model = Sequential()
model.add(Conv2D(32, (3, 3),padding='same',activation='relu', input_shape=(62244,3,1)))
model.add(MaxPool2D(pool_size=(2, 2),padding='same'))
model.add(Conv2D(64, (3, 3),padding='same',activation='relu'))
model.add(Conv2D(128, (3, 3),padding='same',activation='relu'))
model.add(MaxPool2D(pool_size=(2, 2),padding='same'))
model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dense(14, activation='linear'))
# Compile the model
model.compile(loss='mean_squared_error', optimizer='adam',metrics=['mean_absolute_error'])


model.fit(x,y,epochs=15,batch_size=120)
```

Creating Validation data to validate model

```
a.sort()
val_ver=[]
for i in range(154,176):
  mesh = trimesh.load(a[i])
  val_vertices = np.array(mesh.vertices)33
  remaining_rows=62244-val_vertices.shape[0]
  for j in range(remaining_rows):
    val_vertices = np.append(val_vertices,np.array([[0,0,0]]),axis=0)
  val_ver.append(val_vertices)
val_vert=[]
```

```
for i in range(160,179):
  mesh = trimesh.load(imgs[i])
  Val_vertices = np.array(mesh.vertices)
  remaining_rows=62244-Val_vertices.shape[0]
  for j in range(remaining_rows):
    Val_vertices = np.append(Val_vertices,np.array([[0,0,0]]),axis=0)
  val_vert.append(Val_vertices)
y111 = y1.tail(22)
y222 = y2.tail(19)
val_x = np.concatenate((val_ver,val_vert),axis=0)
val_x = val_x/val_x.max()
val_y = pd.concat((y111,y222),axis=0)
pred_y = model.predict(val_x)
val_x.shape
val_y.shape
```

Evaluating the model performance

```
from sklearn.metrics import mean_absolute_error
mean_absolute_error(pred_y,val_y)
# now the mean_absolute_error of the model is 4.026460503036552 which is a god sign t
hat model is performing very well
```

FLASK creation

```
from flask import Flask, request, render_template
import trimesh
import numpy as np
import pickle
from tensorflow.keras.models import load_model
import os
from io import BytesIO
model = load_model('nemodel.h5')
app = Flask(__name__)

@app.route('/')
def ma():
    return render_template("me.html")

@app.route('/predict', methods=['POST'])
def index():
    if request.method == 'POST':
        file = request.files['file']
        data = file.read()
        mesh = trimesh.load(BytesIO(data), file_type='obj')
        vertices = np.array(mesh.vertices)
        remaining_rows=62244-vertices.shape[0]
        if remaining_rows>0:
            for i in range(remaining_rows):
                vertices = np.append(vertices,np.array([[0,0,0]]),axis=0)
        vertices = vertices/vertices.max()
        p = np.reshape(vertices, [1,62244,3,1])
        predictions = model.predict(p).tolist()
        results = {'Seat_Back_Angle':predictions[0],'MEASURE
CrotchLength_Front':predictions[1],'MEASURE
CrotchLength_Back':predictions[2],'MEASURE Knee_Circ':predictions[3],'MEASURE
Calf_Circ':predictions[4],'MEASURE Outseam':predictions[5],'MEASURE
Inseam':predictions[6],'MEASURE Shoulder_to_Wrist':predictions[7],'MEASURE
Bicep_Circ':predictions[8],'MEASURE Elbow_Circ':predictions[9],'MEASURE
Wrist_Circ':predictions[10],' MEASURE
Shoulder_to_Shoulder':predictions[11],'MEASURE
Across_Front':predictions[12],'MEASURE Neck_Circ':predictions[13]}
    return results
if __name__ == '__main__':
    app.run(debug=True)
                document.getElementById(tabname).classList.add("active-tab");
```

6.2 HTML CODE

```
HTML
<!DOCTYPE html>
<html>
<head>
    <title>Body Measurement System</title>
    <style>
        body {
            font-family: Arial, sans-serif;
        }



        h1 {
            text-align: center;
            color: #333;
            margin-top: 50px;
        }
        form {
            display: flex;
            flex-direction: column;
            align-items: center;
            margin-top: 30px;
        }
        label {
            font-size: 20px;
            font-weight: bold;
            color: #333;
            margin-bottom: 10px;
        }
        input[type="file"] {
            font-size: 16px;
            padding: 10px;
            border-radius: 5px;
            border: 1px solid #ccc;
            margin-bottom: 20px;
            background-color: #fff;
            color: #333;
            box-shadow: 1px 1px 1px rgba(0,0,0,0.1);
        }
        button[type="submit"] {
            font-size: 18px;
            font-weight: bold;
            padding: 10px 20px;
            border-radius: 5px;
            border: none;
            background-color: #1E90FF;
            color: #fff;
            box-shadow: 2px 2px 2px rgba(0,0,0,0.2);
            cursor: pointer;
            transition: background-color 0.3s ease;
        }
        button[type="submit"]:hover {
            background-color: #008B8B;
        }
    </style>
</head>
<body>
    <p style="background-image:url('bg1.png');">
    <h1>Body Measurement System</h1>
    <form action="/predict" method="post" enctype="multipart/form-data">
        <label for="file">Upload 3d Image:</label><br>
        <input type="file" id="file" name="file" accept=".obj" required><br><br>
        <button type="submit" name="submit">Predict</button>
    </form>
```

</body></html>

Chapter 7
SYSTEM TESTING

7.1 Introduction

The cause of testing is to detect mistakes. Making an attempt out is the technique of looking for to realize each viable fault or weakness in a piece product. It presents a method to determine the performance of add-ons, sub-assemblies, assemblies and/or a completed product. It is the method of ex excising g program with the intent of constructing certain that the application procedure meets its necessities and client expectations and does no longer fail in an unacceptable process. There are rather plenty of forms of scan. Each experiment sort addresses a special trying out requirement.

7.2 Types of tests

Unit testing

Unit checking out involves the design of scan circumstances that validate that the Internal application good judgment is functioning safely, and that program inputs produce legitimate outputs. All decision branches and interior code float must be validated. It's the checking out of character application items of the application. It is achieved after the completion of an person unit earlier than integration. It is a structural checking out, that relies on competencies of its construction and is invasive. Unit exams participate in common exams at component level and scan a distinct business approach, utility, and/or process configuration.

Unit assessments be certain that every specified course of a industry method performs appropriately to the documented requisites and involves clearly outlined inputs and anticipated results.

Integration testing

Integration Testing are designed to scan built-in program accessories to determine within the occasion that they evidently run as one software. Trying out is occasion driven and is more concerned with the fundamental final result of screens or fields. Integration assessments reveal that despite the fact that the accessories had been for my part pleasure, as proven through effectively unit checking out, the combo of accessories is correct and regular. Integration checking out is chiefly aimed at exposing the issues that come up from the performance of different components.

Functional testing

Functional Testing checks provide systematic demonstrations that capabilities established are to be had as particular by means of the business and technical specifications, method documentation, and consumer manuals. Functional testing is working on below mentioned data:

Legitimate input : identified lessons of legitimate input ought to be accredited.

Invalid enter : recognized lessons of unacceptable effort must be rejected.

Capabilities : recognized features ought to be exercised.

Output : recognized courses of software outputs have got to be

exercised.Systems/Procedures : performance of the system here was invoked Individual and team work of useful checks is fascinated by specifications, key capabilities, or special scan instances. Moreover, systematic insurance plan concerning establish business method flows; data fields, predefined processes, andsuccessive strategies have to be regarded for trying out. Before useful trying out iswhole, extra checks are recognized and the strong price of present checks be strong-minded.

System testing

scheme difficult ensure so as to the whole included agenda process meets principles. It exams a pattern to make sure identified and predictable outcome. An illustration of procedure testing is the configuration oriented approach integration scan. System testing is based on approach descriptions and flows, emphasizing pre-driven systemlinks and integration aspects.

White Box Testing

This testing is a trying out wherein where the application tester has competencies of the interior workings, constitution and software language, or at least its cause. It's rationale. It's used to test areas that can't be reached from a black box stage.

Black Box Testing

This is testing the software with none advantage of the inside workings, establishment or words of the unit life form veteran.

7.3 Levels of testing

Unit testing strategy

Unit checking out is most commonly performed as a part of a mixed code and unit experiment part of the software lifecycle, though it be not exceptional for coding and unit checking out to be performed as two targeted phases.

Test strategy and approach:

Field testing out can be carried out manually and sensible assessments shall be writtenin element.

Test objectives

Each field must be work correctly.

Each page must be activated through the specified link.

Features to be tested Verify that the entries are of the correct format No duplicateentries should be allowed

Integration testing strategy

Software integration testing is the incremental integration checking out of two otherwise further included software gears on top of a solo stage to fabricate failure induced with the aid of interface defects. The project of the mixing scan is to checkthat components or program applications,

e.g. Components in a program approach or œ one step up œ software purposes at the company degree œ interact without error.

Test Results:

All of the scan circumstances recounted above passed efficiently. No defects encountered.

Acceptance Testing

User Acceptance testing trying out is a crucial section of any mission and requires enormous participation by the tip user. It additionally ensures that the procedure meets the functional specifications.

Test Results:

The entire test cases recounted above passed effectually. No defects Encountered

CONCLUSION:

Our project aims to bring out an optimize results of 3D measurements. The project concerns about the suggestions of body measurement which is applicable for e-commerce applications . data set is collected from NOMO-3D-400-Scans which contains 194 male and 181 female subjects to train the model. Full code implementation is produced.

FUTURE ENHANCEMENT

Further we can include brand size recommendation system which reduces the return policy in e-commerce applications.

REFERENCE

NOMO-3D-400-SACNS for dataset collection
2D to 3D conversion code from face book research