# BODYWEIGHT ESTIMATION USING 2D BODY IMAGE

### A Project Report

***Submitted by***

**PRASHANTH S [cb.en.u4cse17140] SAI TRINADH [cb.en.u4cse17129]**

**T.S.S BHARADWAJ [cb.en.u4cse17158] DATLA LOKESH KUMAR VARMA [cb.en.u4cse17109]**

*Under the guidance of*

## Ms. Aarthi R.

(Asst. Professor, Department of Computer Science & Engineering)

***in partial fulfillment for the award of the degree of***

### BACHELOR OF TECHNOLOGY

***in***

### COMPUTER SCIENCE & ENGINEERING

**AMRITA VISHWA VIDYAPEETHAM**



Amrita Nagar PO, Coimbatore - 641 112, Tamilnadu

### May 2021

AMRITA VISHWA VIDYAPEETHAM

### AMRITA SCHOOL OF ENGINEERING, COIMBATORE – 641 112



**BONAFIDE CERTIFICATE**

This is to certify that the project report entitled **BODY WEIGHT ES- TIMATION USING 2D BODY IMAGE** is submitted by PRASHANTH S (cb.en.u4cse17140),SAI TRINADH(cb.en.u4cse17129),T.S.S BHRAD- WAJ(cb.en.u4cse17158),DATLA LOKESH KUMAR VARMA

(cb.en.u4cse17109) in partial fulfillment of the requirements for the award of the Degree Bachelor of Technology in Computer Science and Engineering is a bonafide record of the work carried out under our guidance and supervision at the Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore.

#### SIGNATURE

Ms. Aarthi R. PROJECT GUIDE

Asst. Professor

Dept. of Computer Science & Engi-neering

Signature of the Internal Examiner

#### SIGNATURE

Dr. (Col.) P. N. Kumar CHAIRPERSON

Dept. of Computer Science & Engi-neering

Signature of the External Examiner

# DECLARATION

I the undersigned solemnly declare that the project report **BODY WEIGHT ESTIMA-TION USING 2D BODY IMAGE** is based on my own work carried out during the course of our study under the supervision of Ms. Aarthi R., Asst. Professor, Com- puter Science & Engineering, and has not formed the basis for the award of any other degree or diploma, in this or any other Institution or University. In keeping with the ethical practice in reporting scientific information, due acknowledgment has been made wherever the findings of others have been cited.

Prashanth S[cb.en.u4cse17140] SAI TRINADH [cb.en.u4cse17129]

T.S.S Bharadwaj [cb.en.u4cse17158]

Datla Lokesh Kumar Varma[cb.en.u4cse17109]

# ABSTRACT

Images of humans encode several useful soft biometric features such as height, eye- colour, gender, age, bodyweight etc. Among these body weight is a key human feature that is useful in several applications including health apps, quick patient features for remotely practicing doctors, dynamic luggage allowance in airports, and as an added feature for surveillance. With most current work relying on the use of additional features obtained from 3D cameras or kinetic RGB-D sensors ([Velardo and Dugelay, 2010)](#_bookmark110) [(Pfitzner et al., 2017),](#_bookmark104) multiple stereo cameras [(Santner et al., 2009),](#_bookmark105) all of which are not only expensive but require additional apparatus which may not be readily available in all use case scenarios as they aren’t common. Motivated by this need, this work investigates the feasibility of estimating the body weight of a person given only the two-dimensional image of the person using the two most common types of images available which are facial and full-body images.

A framework is developed for this work, which includes, collection of a dataset which

consists of full body images, performing feature extraction using skeleton joint detection, body contour detection, and facial landmarking to obtain key anthropometric features to estimate body and face adiposity (fatness), analyzing the correlation between extracted anthropometric features and ground truth body weight, using deep learning models to predict the body weight given the extracted anthropometric features, using CNN model as well as three transfer learning approaches to estimate weight directly from the image, an extensive causal analysis of the result obtained versus the demographic of the subject images used for model training and testing and a web application which will accept the image of a person and output the estimated weight to highlight the proof of concept.

# ACKNOWLEDGEMENTS

We would like to express my deep gratitude to our beloved Satguru **Sri Mata Amritanandamayi Devi** for providing the bright academic climate at this university, which has made this entire task appreciable. This acknowledgment is intended to be thanks- giving measure to all those people involved directly or indirectly with my project. I would like to thank our Pro Chancellor **Bramachari Abhayamrita Chaitanya**, Vice Chancellor **Dr. Venkat Rangan. P** and **Dr. Sasangan Ramanathan**, Dean Engineering of Amrita Vishwa Vidyapeetham for providing us the necessary infrastructure re-quired for the completion of the project. I express my thanks to **Dr. (Col) P.N. Kumar**, Chairperson of the Department of Computer Science Engineering, **Dr. G. Jeyakumar, and Dr. C. Shunmuga Velayutham,** Vice-Chairpersons of the Department of Computer Science and Engineering for their valuable help and support during our study. I express my gratitude to my guides, **Names of guides, the designation** for their guidance, support, and supervision. I feel extremely grateful to the **Review panel member's names, the designation** for their feedback, and encouragement which helped us to complete the project. I would also like to thank the entire staff of the Department of Computer Science and Engineering. I would like to extend my sincere thanks to my family and friends for helping and motivating me during the course of the project. Finally, I would like to thank all those who have helped, guided, and encouraged me directly or indirectly during the project work. Last but not the least, I thank God for His blessings which made my project a success.

Your Name Roll No

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| [**ABSTRACT**](#_bookmark0)  [**ACKNOWLEDGEMENTS**](#_bookmark1) |  | **iv**  **v** |
| [**List of Tables**](#_bookmark3) |  | **viii** |
| [**List of Figures**](#_bookmark4) |  | **ix** |
| [**Abbreviations**](#_bookmark5) |  | **x** |
| [**List of Symbols**](#_bookmark15) |  | **xi** |
| [**1 Introduction**](#_bookmark16)  [1.1 Problem Definition](#_bookmark17) . . . . . . . . . . . . . . . . . . . . . . . . . | .1 | **1** |
| [1.2 Motivation](#_bookmark18) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | .1 |  |
| [1.3 Applications](#_bookmark19) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | .2 |  |
| [1.4 Approaches used for body weight estimation](#_bookmark20) . . . . . . . . . . . . | .2 |  |
| 1. [**Literature Survey**](#_bookmark25)    1. [Weight Estimation using full body data](#_bookmark26) . . . . . . . . . . . . . . . | .4 | **4** |
| [2.2 Weight Estimation Using Facial Image Data](#_bookmark27) . . . . . . . . . . . . | .5 |  |
| [2.3 Approaches for Data Preprocessing and Feature Extraction](#_bookmark28) . . . . | .5 |  |
| [2.4 Summary](#_bookmark29) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | .7 |  |
| [2.5 Datasets](#_bookmark30) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | .8 |  |
| [2.6 Software Requirements](#_bookmark32) . . . . . . . . . . . . . . . . . . . . . . . | .8 |  |
| 1. [**Proposed System**](#_bookmark33)    1. [Flow Diagram of the System](#_bookmark34) . . . . . . . . . . . . . . . . . . . . | .9 | **9** |
| [3.2 Algorithm and Module Description](#_bookmark35) . . . . . . . . . . . . . . . . . | .9 |  |
| [3.2.1 Dataset](#_bookmark37) . . . . . . . . . . . . . . . . . . . . . . . . . . . | .9 |  |
| [3.2.2 Data Pre-processing](#_bookmark38) . . . . . . . . . . . . . . . . . . . . | .10 |  |
| [3.2.3 Feature Selection](#_bookmark40) . . . . . . . . . . . . . . . . . . . . . . | .11 |  |
| [3.2.4 Model Training and Testing](#_bookmark42) . . . . . . . . . . . . . . . . | .12 |  |
| [3.2.5 Evaluation](#_bookmark43) . . . . . . . . . . . . . . . . . . . . . . . . . | .13 |  |
| [3.3 Architecture Diagram](#_bookmark45) . . . . . . . . . . . . . . . . . . . . . . . . | .14 |  |
| [3.3.1 For Full Body Features](#_bookmark46) . . . . . . . . . . . . . . . . . . . | .14 |  |
| [3.3.2 For Localised Face Features](#_bookmark48) . . . . . . . . . . . . . . . . | .14 |  |
| [3.4 Conclusion](#_bookmark50) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | .14 |  |
| [**4 Implementation and Testing**](#_bookmark51)  [4.1 Pre-processing](#_bookmark52) . . . . . . . . . . . . . . . . . . . . . . . . . . . . | .16 | **16** |
| [4.1.1 Detecting Body Contour](#_bookmark53) . . . . . . . . . . . . . . . . . . | .16 |  |
| [4.1.2 Body (Torso part) Land-marking](#_bookmark55) . . . . . . . . . . . . . . | .16 |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | [4.1.3 Segment Isolation and](#_bookmark57)  [Invalid Body Image Elimination](#_bookmark57) . . . . . . . . . . . . . . | .17 | |
| [4.1.4 Gray Scaling and Image Standardisation](#_bookmark58) . . . . . . . . . . | .18 | |
| [4.1.5 Face Localisation and Facial Land-marking](#_bookmark59) . . . . . . . . | .18 | |
| [4.2](#_bookmark61) | [Feature Extraction](#_bookmark61) . . . . . . . . . . . . . . . . . . . . . . . . . . | .19 | |
|  | [4.2.1 Calculating Facial Ratios](#_bookmark62) . . . . . . . . . . . . . . . . . . | .19 | |
|  | [4.2.2 Anthropometric Measurements from Body](#_bookmark63) . . . . . . . . | .19 | |
|  | [4.2.3 Final List of Features](#_bookmark64) . . . . . . . . . . . . . . . . . . . . | .19 | |
| [4.3](#_bookmark65) | [Model Training and Testing](#_bookmark65) . . . . . . . . . . . . . . . . . . . . . | .20 | |
|  | [4.3.1 Deep Learning Model](#_bookmark66) . . . . . . . . . . . . . . . . . . . | .21 | |
|  | [4.3.2 XGBoost Regressor](#_bookmark68) . . . . . . . . . . . . . . . . . . . . | .21 | |
|  | [4.3.3 Convolutional Neural Network](#_bookmark69) [(CNN)](#_bookmark9) [model](#_bookmark69) . . . . . . . | .22 | |
|  | [4.3.4 Transfer Learning](#_bookmark71) . . . . . . . . . . . . . . . . . . . . . . | .22 | |
| [4.4](#_bookmark73) | [Web Application](#_bookmark73) . . . . . . . . . . . . . . . . . . . . . . . . . . | .23 | |
| [**5 Results and Discussion**](#_bookmark75)  [5.1 Measurement Metric](#_bookmark76) . . . . . . . . . . . . . . . . . . . . . . . . | | .25 | **25** |
| [5.2 Model Approach-Wise Analysis](#_bookmark79) . . . . . . . . . . . . . . . . . . | | .25 |  |
| [5.2.1 Deep Learning and XGboost models](#_bookmark80) . . . . . . . . . . . . | | .27 |  |
| [5.2.2 CNN](#_bookmark82) . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | .27 |  |
| [5.2.3 Transfer Learning using Deep](#_bookmark85) [CNNs](#_bookmark9) [Trained on Imagenet](#_bookmark85) | | .28 |  |
| [5.3 Graphical analysis](#_bookmark87) . . . . . . . . . . . . . . . . . . . . . . . . . . | | .29 |  |
| [5.3.1 Ground Truth vs Prediction Scatterplot](#_bookmark89) . . . . . . . . . . | | .29 |  |
| [5.3.2 Error Distribution Histogram](#_bookmark90) . . . . . . . . . . . . . . . . | | .30 |  |
| [5.4 Testing on Novel Self-created dataset for external model validation](#_bookmark92) | | .31 |  |
| [**6 Conclusion**](#_bookmark95) | |  | **33** |
| [**7 Future Enhancement**](#_bookmark96) | |  | **34** |

**LIST OF TABLES**

[2.1 Dataset details](#_bookmark31) 8

* 1. [Model Test Results on Visual-Body-to-BMI dataset](#_bookmark81) 26
  2. [Model Results on Novel Self-Collected Test dataset](#_bookmark93) 32

**LIST OF FIGURES**

[1.1 An overall illustration of the approaches used](#_bookmark21) 2

* 1. [Overall Flow Diagram of the system](#_bookmark36) 9
  2. [Architecture Diagram for Full Body Features](#_bookmark47) 14
  3. [Architecture Diagram for Localised Face Features](#_bookmark49) 14
  4. [Body Contour Detection](#_bookmark54) 16
  5. [Body Land-Marking](#_bookmark56) 17
  6. [Face Land-Marking](#_bookmark60) 18
  7. [Deep Learning Model architecture](#_bookmark67) 21
  8. [CNN model architecture](#_bookmark70) 22
  9. [Transfer Learning model architecture](#_bookmark72) 23
  10. [Web Application Flow Screenshots](#_bookmark74) 24
  11. [CNN model training loss](#_bookmark84) 27
  12. [Transfer Learning training loss curves](#_bookmark86) 28
  13. [Ground Truth vs Prediction Scatterplot](#_bookmark88) 29
  14. [Error Distribution Histogram](#_bookmark91) 30
  15. [Distribution of Absolute error on Testing with Novel dataset](#_bookmark94) 32

# ABBREVIATIONS

[**BCD**](#_bookmark22)[Body Contour Detection](#_bookmark22)

[**SJD**](#_bookmark23)[Skeleton Joint Detection](#_bookmark23)

[**DL**](#_bookmark24)[Deep Learning](#_bookmark24)

[**CNN**](#_bookmark2)[Convolutional Neural Network](#_bookmark2)

[**MAE**](#_bookmark77)[Mean Absolute Error](#_bookmark77)

[**FLD**](#_bookmark39)[Facial Landmark Detection](#_bookmark39)

[**CRF-RNN**](#_bookmark41)[Conditional Random Fields as Recurrent Neural Networks](#_bookmark41)

[**Kg**](#_bookmark83)[Kilo Grams](#_bookmark83)

[**ML**](#_bookmark44)[Machine Learning](#_bookmark44)

# List of Symbols

**Chapter 1**

## INTRODUCTION

* 1. **Problem Definition**

To estimate the body weight of a person given the two-dimensional image of the person.

# Motivation

There are several aspects of motivation for this work. First, we know that human vision can intuitively interpret approximate body fatness/weight by observing 2D images of people. Furthermore, there are several use cases where it is neither desirable nor feasible to measure the bodyweight of people manually as it is both time-consuming and labour intensive and cannot be scaled to meet the demands of handling large groups of people quickly. Studies in health sciences [(Molarius and Seidell, 1998)](#_bookmark102) [(Seidell et al.,](#_bookmark107) [1987)](#_bookmark107) clearly show that obesity and correlated body weight can be estimated by visually perceivable anthropometric features such as hip-shoulder ratio, thigh-waist ratio, waist-height ratio.

Most current works rely on the use of additional features obtained from 3D cameras

or kinetic RGB-D sensors ([Velardo and Dugelay, 2010)](#_bookmark110) [(Pfitzner et al., 2017),](#_bookmark104) the use of multiple stereo cameras [(Santner et al., 2009),](#_bookmark105) all of which are not only more ex- pensive but require additional apparatus which may not be readily available in all use case scenarios. Hence there is a need for a reasonably accurate system for estimation of weight that is both fast and inexpensive. This can be achieved by using existing hardware (cameras), that are already used for surveillance in various public spaces for estimation of weight.

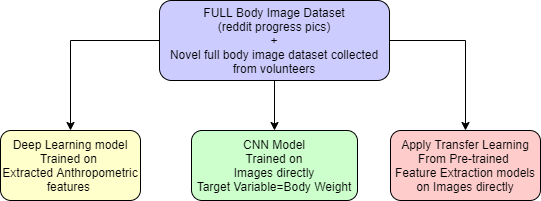
Observers seem to be able to glean body weight information from frontal views of a face [(Schneider et al., 2012)](#_bookmark106) and past research has replicated this using machine learning models. Additionally, it is inevitable that full-body image should provide better insight as anthropometric measurements can be used to further increase accuracy.

# Applications

Following are some of the applications of body weight estimation using 2D full-body images.

* + - Airport dynamic luggage allowance – Airports can use it to dynamically change luggage allowance policy based on how much weight passengers are taking up till a given point. Passengers will usually be hesitant to be weighed during check-in since it is a public space and also, hardware-based weighing takes a few seconds which will cause propagation of delay in check-in queues. Image-based weight determination will help in this case.
    - Fitness recommendation on health apps – Health apps can provide better fitness recommendations based on image data provided by the user instead of having to input weight manually.
    - Surveillance – Extracting body weight is also pertinent in surveillance to add additional features to a photograph which would be useful in short descriptions of people such as "a short stout person", "tall lean person", "average height and fat" that would ease identification of a subject. While some soft biometrics like colour or length of hair can be changed quickly, it is not trivial to change visual body appearance.
    - Remote BMI calculation by doctors – Measure weight in rural settings where hardware is not available such that a doctor practicing remotely can have access to that metric too. Given that an adult human’s height does not change much over time if provided with the height, BMI, which is indicative of a person’s health can be calculated.

# Approaches used for bodyweight estimation



**Figure 1.1:** An overall illustration of the approaches used

Three main approaches are used for the estimation of body weight as shown in Figure [1.1.](#_bookmark21)

#### Deep learning model on Extracted anthropometric features

Given the full body image, using Body Contour Detection [(BCD)](#_bookmark6) a pixel-wise outline of the human subject is isolated which along with Skeleton Joint Detection [(SJD)](#_bookmark7) is used to extract key anthropometric features from the body. Additionally, facial landmarking is used to extract facial features from localized faces from the full-body images, that indicate facial adiposity (fatness). These extracted features are used to train a Deep Learning [(DL)](#_bookmark8) model to estimate the weight of the human. The facial and full-body extracted features are used to train two separate [DL](#_bookmark8) models and another [DL](#_bookmark8) model is trained on the combined features and the results are compared and analyzed.

#### [CNN](#_bookmark9) model on input 2D image

[CNN](#_bookmark9) model is trained on the 2D full-body images with the body weight as the target variable. Also, the [BCD](#_bookmark6) pixel-wise segmented image of the subject is also experimented with to see if elimination of background noise has an effect and the results of both the experimentation are compared and analyzed.

#### Transfer Learning models on input 2D image

Experimentation was done with three transfer learning models namely

* + - ResNet
    - XceptionNet
    - InceptionV3

These three [CNN](#_bookmark9) models are used as feature extractors. All three models have shown exceptional performance in the ImageNet object classification task. We truncate the last softmax layer of the models and add a fully connected layer and a final linear layer since we are treating the weight estimation problem as a regression problem.

In the subsequent chapters, we will deal with how weight can be estimated using Images for both Facial and Full Body image data. In the subsequent chapters, we will go through the datasets collected and the software requirements applicable for the project. Then the architecture diagram with a brief description of all the modules will be de-scribed along with the timeline for completion which will include designation of individual contributions.

# Chapter 2

## LITERATURE SURVEY

* 1. **Weight Estimation using full-body data**

A few studies are working on estimating human body weight or BMI from body-related data, such as body measurements, 3-dimensional (3D) body data, and RGB-D body images. Also, body contour detection based on the characteristics of the surface of the human body using a set of algorithms suggested in [Wang et al. (2017)](#_bookmark111) can be used to the main areas of the cloth, hair, and skin, taking into account the large individual differences in the body color, for a more accurate understanding of Full Body adiposity or fatness than using body joint detection alone. [Jiang and Guo (2019)](#_bookmark101) used images of the entire front body by scraping data from a Reddit page subreddit called preogress pics. To estimate BMI they used body contour and skeleton joints detected by CSJ detector and estimated the weight based on the full-body landmarking of both the skeleton joints and hence the body contours. Several other features such as shoulder width, shoulder to waist ratio were determined to aid in the process. [Velardo and Dugelay (2010)](#_bookmark110) studied the bodyweight directly from anthropometric data (body measurements) collected by National Health and Nutrition Examination Survey, CDC. A polynomial regression model was employed to analyze the anthropometric data. [Velardo and Dugelay](#_bookmark110) [(2010)](#_bookmark110) estimated the weight of a person within 4% error using 2D and 3D data extracted from a low-cost Kinect RGB-D camera output. [Pfitzner et al. (2017)](#_bookmark104) also used RGB-D camera data, this demonstrated a bodyweight estimation by volume extraction from RGB-D data. The presented algorithm provided an accuracy of 79% for a cumulative error of ±10%. The approach was tested with 110 patients from the trauma room, focusing on bodyweight estimation for stroke patients. Due to uncertainties in volume estimation, this approach had outliers up to 32%. Compared to a physician’s estimation this approach is already more suitable for drug dosing. Also, anthropometric features from depth data can be found in related work: [Santner et al. (2009)](#_bookmark105) estimated human body volume in a clinical environment by eight stereo cameras around a stretcher and bioelec

trical impedance analysis. With a 3D reconstruction, the volume of a frontal surface can be calculated towards the medical stretcher which has been modeled as a plane. [Nguyen](#_bookmark103) [et al. (2014)](#_bookmark103) developed a method to predict body weight by a side view feature and a support vector regression model. Separating datasets by gender their approach reached an average error of 4.62 kg for females and 5.59 kg for males. Finally, they compared the bodyweight estimation by the algorithm in contrast to visual estimation.

# Weight Estimation Using Facial Image Data

Instead of estimating body weight or BMI from body images, some work analyzed body weight or BMI from face images. [Wen and Guo (2013)](#_bookmark112) first proposed a computational method for BMI prediction from face images based on the MORPH-II dataset, which obtained mean absolute errors (MAEs) for BMI in the range from 2.65-4.29 for different ethnic categories. They also analyzed the correlations between facial features and BMI values. An Active Shape Model is used to extract facial features which are used to predict BMI using various regression techniques. In [Barr et al. (2018)](#_bookmark97) Facial landmarking was used to figure out adiposity (facial fatness) which positively correlates to the weight of the person. This method is less accurate in extreme underweight and obese cases though. An SVR regression model was used. [Windhager et al. (2017)](#_bookmark113) showed that the Shape of the face has a direct correlation with several body characteristics such as height and weight and can be determined by facial landmarking and spatial scaling, A total of 71 landmarks, and semi landmarks were digitized to capture facial shape. Regression and Geometric morphometric toolkit tool used to estimate facial fatness. Additionally, as a feature set height measurement using anthropometer and saliva sample testing done apart from the facial front photograph. [Tai and Lin (2015)](#_bookmark109) used where estimation of BMI is done using facial data on a regression model.

# Approaches for Data Preprocessing and Feature Extraction

BlazePose [(Bazarevsky et al., 2020a)](#_bookmark98) is a lightweight convolutional neural network architecture for estimating human pose on mobile devices that are optimized for real-time

inference. Producing heatmaps for each joint, as well as refining offsets for each coordinate, is a typical method. By doing this the heatmaps allow the model to scale to several users with a minimal overhead which makes the model for a single person much larger than is appropriate for real-time inference on mobile phones. A lightweight body pose detector is accompanied by a pose tracker network in the pipeline. The pose estimation tracker uses the person alignment proposal provided by the first stage of the pipeline to predict the position of all 33 person key points along with the confidence value. The Body contour detection algorithm can use these identified body joints as a map to extract main anthropometric features that indicate body adiposity. The detected joints’ confidence level can be used to filter out photos of too many occlusions from the training dataset. They used the Percent of Correct Points with a 20 percent tolerance (PCK@0.2) as an estimation metric (where we presume the point is correctly identified if the 2D Euclidean error is less than 20% of the corresponding person’s torso size).

Image comprehension relies heavily on pixel-level labeling tasks such as semantic seg-

mentation. The latest methods have attempted to use deep learning techniques for image recognition to solve pixel-level labeling problems. The inability of deep learning techniques to distinguish visual objects is a major flaw in this approach. So [(Zheng et al.,](#_bookmark115) [2015a)](#_bookmark115) have introduced a new form of convolutional neural network that combines the strengths of Convolutional Neural Networks (CNNs) and Probabilistic Graphical Modeling based on Conditional Random Fields (CRFs). This CRF-RNN network is then plugged into a CNN to create a deep network that combines the best features of both CNNs and CRFs. The Body contour detection can be done using CRF as an RNN, and any imperfections can be smoothed out using morphological image processing techniques.They illustrated this strategy by integrating the CRF-RNN with a completely convolutional neural network. On the famous Pascal VOC segmentation benchmark, they set a new record. The ability to combine the strengths of CNNs and CRFs in a single deep network explains this change.

Bottom-up DCNN-based methods are insufficient for obtaining semantic segmentation

results, especially for fitting object boundaries. They proposed creating a probabilistic superpixel-based dense CRF model (PSP-CRF) as a post-processing approach to refine label assignments using superpixels that contain local detailed information in a top-down manner. It's an entropy-based strategy for converting pixel-level features (colour,

position) and probabilistic values at each pixel into normalized superpixels that match the CRF well. The methods [(Zhang et al., 2018)](#_bookmark114) increase segmentation efficiency while reducing processing time as compared to other post-processing methods, according to the results of their experiments. In comparison to other approaches, CRF would be the preferred model for semantic segmentation.

This study [(Sánchez Hernández et al., 2020)](#_bookmark108) included a systematic analysis of more than 50 scientific articles, showing that deep learning networks have made considerable progress in object detection, and that deep models have greatly improved efficiency, but that there are still many problems and challenges. Vgg is faster than other models at image localization. It almost detects objects in real-time for a given image. According to the metrics, the MAP value (73.2) is equivalent to the other versions; it can detect an object in a given image more easily, accurately, and quickly which made vgg an ideal model.

Body surface area (BSA) regression is used to measure the weight of the observed sub-

jects. To extract practical features and predict BMI scores with 95% precision, the proposed system [(Zhang et al., 2018)](#_bookmark114) uses a state-of-the-art deep residual network. Weights are calculated based on Body Surface Area. The models’ ability to generalize from learn- ing on synthetic images to real images was demonstrated by the results. This enables the solution to be packaged into a low-cost embedded RGB-D unit, such as Kinect, with a frame rate of up to 30 frames per second.

# Summary

Most of the research in the past years has been to estimate BMI using facial images and in some cases body weight. While that is shown to have a high correlation to BMI and body weight, it is prone to high error. Since the estimation of height is difficult without the perception of depth in 2D images and height for adults remains unchanged over time unlike weight, we propose to estimate body weight alone through measurement of body adiposity (fatness). To reduce the error in using only face images, body image has been incorporated in the newer research although in some cases, at the cost of expensive hardware (thermal camera/depth sensor). [Jiang and Guo (2019)](#_bookmark101) showed that using a 2D body image dataset is viable to estimate BMI with reasonable accuracy.

With our project, we intend to combine both – facial and full-body image data and develop a model to estimate the bodyweight with improved accuracy. And use contour detection in addition to a CNN model for improved accuracy. This will be extended to logging the weight of people from a video stream with a time stamp.

# Datasets

Table 2.1: Dataset details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S. N.** | **Name** | **Source** | **Size** | **Features** | **Remarks** |
| 1. | Visual-Body  -to-BMI  Dataset | Received on request, Min Jiang | 5900  Images | body image pair (before and after weight change), weight, height, sex | full body image pairs scraped from ProgressPics subreddit |
| 2. | Novel  Self-collected Full Body ImageDataset | Collected  from Volunteers | 232  Images | Full body image, weight, age | Manually collected Images of known volunteers |

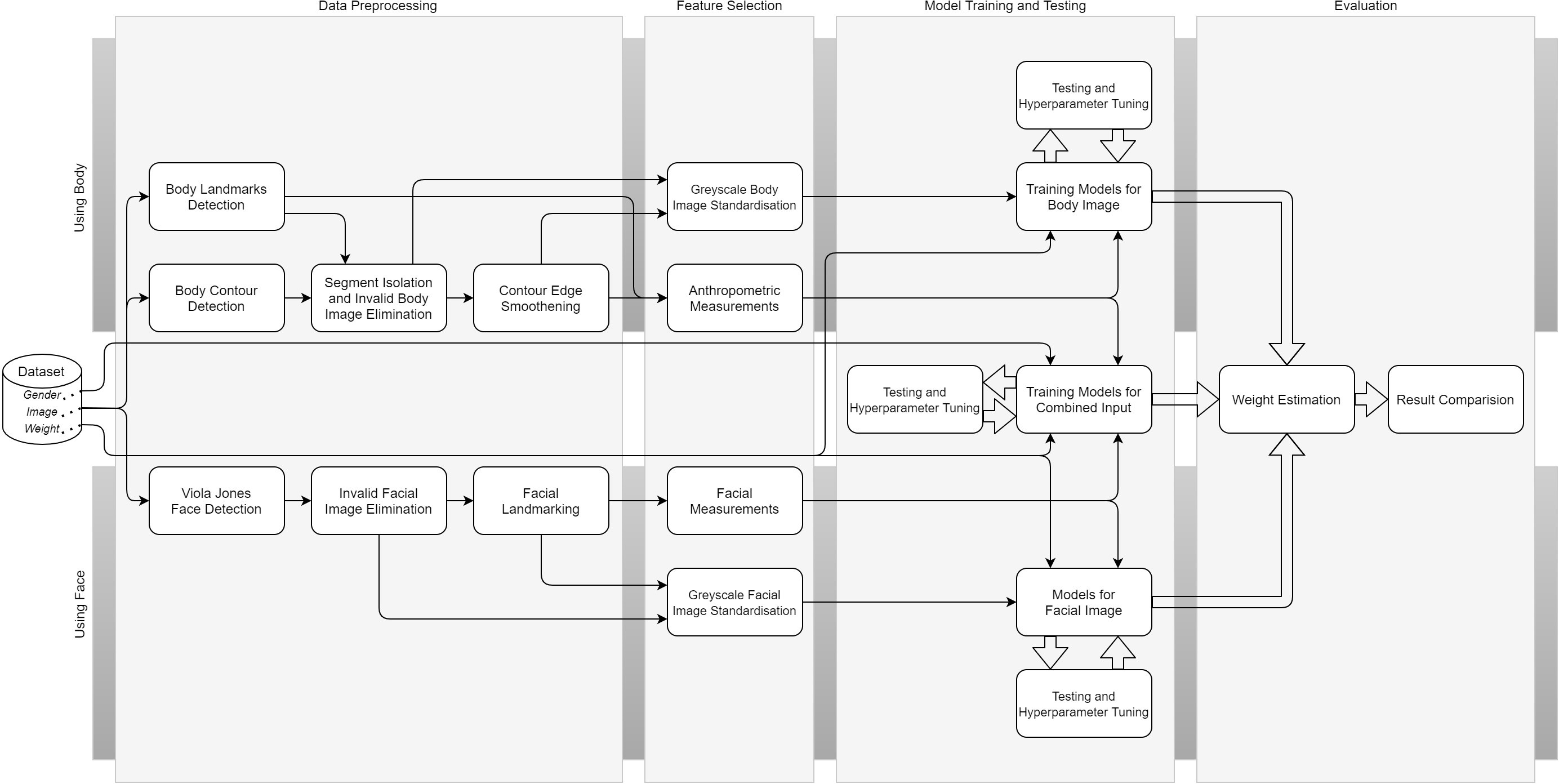
# Software Requirements

* + - Python – a language of choice for implementing core programming
    - OpenCV – image pre-processing and standardization
    - VGG – face localization and feature extraction
    - MediaPipe Blazepose – body landmarks feature extraction
    - TensorFlow – implementing ML model
    - CRFRNN – body contour detection
    - Flask – Web application

# Chapter 3

## PROPOSED SYSTEM

* 1. **Flow Diagram of the System**



**Figure 3.1:** Overall Flow Diagram of the system (zoom in to view diagram)

# Algorithm and Module Description

### Dataset

We have used the Visual-Body-to-BMI dataset that consists of 5900 images of humans which contain their entire body. This dataset was received on request from Author Min Jiang [(Jiang and Guo, 2019).](#_bookmark101) It contains 2950 pairs of images scraped from the subreddit r/ProgressPics. Each pair of images is the image of a human subject before and after weight loss which they have voluntarily uploaded in Reddit which is publicly available. Though the dataset consists of additional features such as age and gender, these features aren’t used since the use cases we have mentioned do not allow for any

additional information on the human subject apart from their 2D photograph.

To test the robustness of the trained model. An additional set of 232 full-body images is consolidated by us. These have been collected from volunteers and will help in determining if the proposed model is capable of estimating the weight of real-world human subjects.

Table [2.1](#_bookmark31) illustrates the details of the dataset in a concise format.

### Data Pre-processing

This module consists of all the stages that deal with the cleaning and preprocessing of the dataset that will aid in the stage of feature extraction and selection.

#### For Full Body Image used in DL and XGBoost regressor

This module consists of three submodules namely Body Contour detection, Invalid Body Image Elimination, and Contour Edge Smoothening. This is done to first detect the contour of the body using an existing model that extracts the exact shape of the body within the image and hence isolates it from the environment thus aiding in measurements done on the body. Then certain images that haven’t been properly diagnosed and have a fair bit of noise are eliminated. Then to make the body contour even sharper it is smoothened.

* + - * Body Contour Detection

The extraction of human contours is an essential part of the development of vision-based non-contact human body measurements and modeling systems. Here we colour code the human body shape alone from the image thus enabling ease in segregation from the background environment to aid measurement of various features that will serve as input features to the [DL](#_bookmark8) model.

* + - * Contour Edge Smoothening

Due to several factors such as noise in the image, posture, background discrepancy and inaccuracy of the contour detection algorithm itself, this becomes a crucial step in ensuring the measurements derived from the image in the feature selection stage are as accurate as possible. This will in turn positively affect the accuracy of the overall model.

#### For Full Body Image used in CNN model and 3 Transfer Learning Approaches

Due to the raw nature of the Images scraped from Reddit, the Images ( as in the real world ) do not conform to any standards apart from being portrait-oriented images. This

causes an issue as each image will have a varying amount of background noise that can cause the accuracy of the prediction to be dependant on the amount of background in an image. Also, all images need to be of a particular dimension to be processed by these models.

To solve this human body localization is done and the bounding box is expanded to be of the size 59 x 106 which was the average size that all images would conform to.

#### For Facial Landmark Detection [(FLD)](#_bookmark11) from full-body image used in DL and XG- Boost regressor

Since the dataset used consists of full-body images, it is imperative that any form of feature extraction from the face would benefit from the localization of the face. Hence, the [FLD](#_bookmark11) algorithm is fed with the localized face as the input instead of the entire full-body image to improve the results from the [FLD.](#_bookmark11)

### Feature Selection

This is a vital stage between the raw data preprocessing and the model training stage that involves using the cleaned and labeled image from the data preprocessing stage to extract features to be used in model training which are unique for the three models being built.

* + - * Model using extracted anthropometric features from the full-body using Blazepose [SJD](#_bookmark7) and [BCD](#_bookmark6)
      * Model using extracted facial features from localized face image using [FLD](#_bookmark11)
      * The combined model that uses both the features mentioned above

For the purely facial image model, facial measurements are extracted from the [FLD](#_bookmark11) performed in the preprocessing stage and for the full-body image-based model anthropometric measurements are derived from the labeled contour detected on the body image. This stage consists of another sub-module involving the Grayscale Image standardization to be used in all three of the models.

#### Body Features (Anthropometric Measurements) using [SJD](#_bookmark7) and [BCD](#_bookmark6)

From the labeled image with body contouring done using Conditional Random Fields as Recurrent Neural Networks [(CRF-RNN)](#_bookmark12) approach. Key body anthropometric fea

tures such as ratios of shoulder-length, height to waist length ratio, etc. are extracted using the Blazepose [SJD.](#_bookmark7) These become features that will be used for training the [DL](#_bookmark8) model and multiple regression models such as XGBoost, Adaboost, etc. In the results section, the best performing regression model is illustrated for the sake of brevity.

#### Using Facial Image (Facial Measurements)

From the localized face from the full-body image with facial landmarking done several vital features such as cheekbone to jaw width, width to upper facial height ratio, perimeter to area ratio, eye size, lower face to face height ratio, face width to lower face height ratio and mean of eyebrow height, etc. These become features that will be used for training the [DL](#_bookmark8) model and multiple regression models such as XGBoost, Adaboost, etc. In the results section, the best-performing regression model is illustrated for the sake of brevity.

### Model Training and Testing

This module consists of all the stages involved in building the [DL,](#_bookmark8) regression models, [CNN,](#_bookmark9) and Transfer Learning models using the extracted features from the feature selection module for the [DL](#_bookmark8) and regression models and the cropped full-body image for the [CNN](#_bookmark9) and Transfer Learning models. This module involves the building of 3 [DL](#_bookmark8) and XGboost regressor models using

* + - * extracted anthropometric features from full-body using Blazepose [SJD](#_bookmark7) and [BCD](#_bookmark6)
      * extracted facial features from localized face image using [FLD](#_bookmark11)
      * The combined model that uses both the features mentioned above

Additionally, [CNN](#_bookmark9) and

3 Transfer Learning models namely

* ResNet
* XceptionNet
* InceptionV3

which are trained on the image directly are experimented with.

The models will be subjected to continuous testing and tuning of various hyperparameters to have the best fit on the dataset.

### Evaluation

#### Weight Estimation

Using the prediction output from the models, the weight of the person is estimated from the image given as input.

#### Result Comparision

Here, we compare the results obtained from all the models built:

* + - * DL model for localized facial features
      * DL model for body anthropometric features
      * DL model for combined localized facial features and body anthropometric features
      * XGBoost model for localized facial features
      * XGBoost model for body anthropometric features
      * XGBoost model for combined localized facial features and body anthropometric features
      * [CNN](#_bookmark9) model on pre-processed full-body image
      * InceptionV3 model on pre-processed full-body image
      * ResNet model on pre-processed full-body image
      * XceptionNet model on pre-processed full-body image

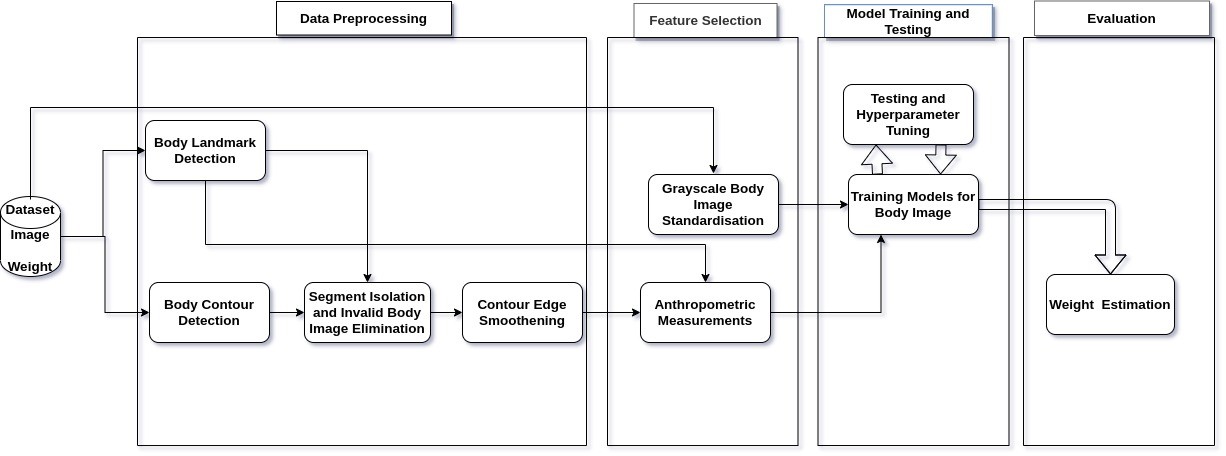
*the results of only the best performing Regression* *Machine Learning* [*(ML)*](#_bookmark14) *model which is the XGBoost regressor is illustrated for the sake of brevity.*

and compare them using various metrics of supervised models and come to a con

clusion on which model best suits the use case of Estimating the weight of a person using only the image of the person which has proven to be a useful feature in several fields of study.

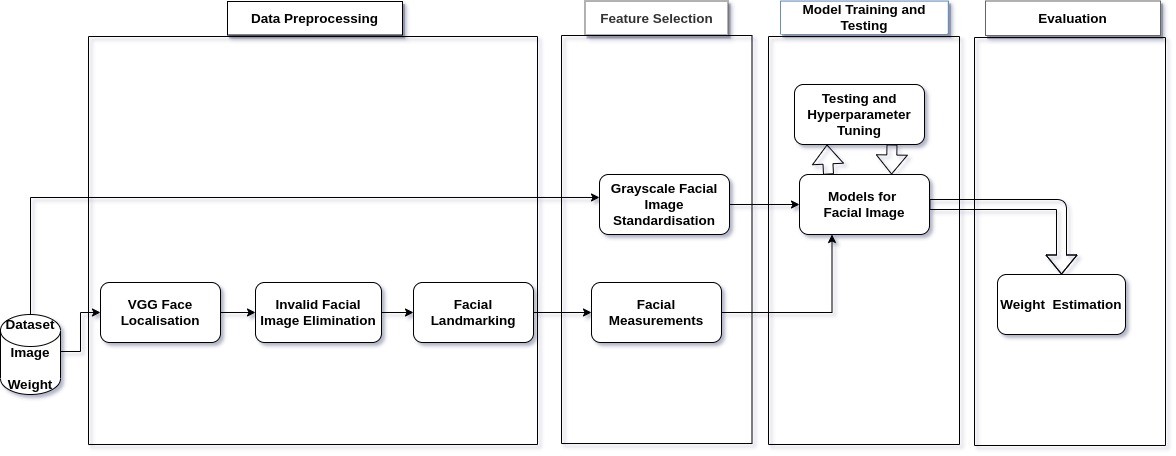
# Architecture Diagram

### For Full Body Features



**Figure 3.2:** Architecture Diagram for Full Body Features

### For Localised Face Features



**Figure 3.3:** Architecture Diagram for Localised Face Features

# Conclusion

* Through the research, we intend to use full-body images to predict the weight of the person.
* We achieve this using various novel stages of preprocessing and feature selection that optimize the dataset thus enabling extraction of latent features of pertinence to help improve the decision making of the chosen model built.
* We then intend to build and test multiple models and approaches for making this prediction.
* Finally, we will compare the results of each of the models and look for ways to improve overall accuracy.

# Chapter 4

## IMPLEMENTATION AND TESTING

* 1. **Pre-processing**

*Since weight is to be estimated only from the images, all features except the weight* *are dropped from the feature set.*

### Detecting Body Contour

As implemented by [Zheng et al. (2015b),](#_bookmark116) we use a custom Python library based on Conditional Random Fields as Recurrent Neural Networks [(CRF-RNN)](#_bookmark12) to isolate the body of the person from the image. The output of the preprocessed image consists of a labeled image with each labeled item indicating an object.

Figure [4.1](#_bookmark54) shows how the segmented output looks like for a sample image.

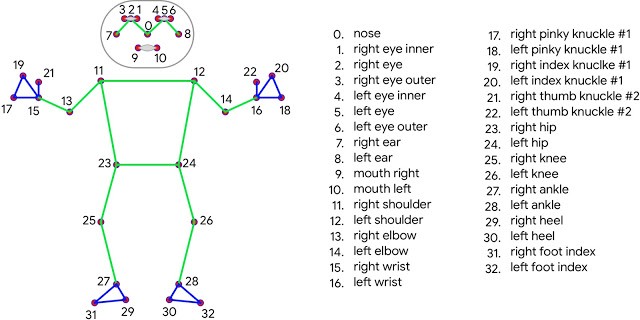


**Figure 4.1:** Body Contour Detection

### Body (Torso part) Land-marking

As implemented by [Bazarevsky et al. (2020b),](#_bookmark99) we use the pose detection framework available through Google’s MediaPipe library to return coordinates as well as their respective confidence values for a given body image. It is a skeleton joint detector. Key coordinates include shoulder joints, hip joints, elbow joints, etc.

Figure [4.2](#_bookmark56) shows the list of landmarks available in the full-fledged model.



**Figure 4.2:** Body Land-Marking

The API in its current iteration does not provide any landmarks below the hip joints. Crucially, the BlazePose model is optimized for on-board mobile computation. Therefore, the features calculated from it are extremely fast to obtain and they can be used to make a rough estimation of weight if allowed by the use case. Furthermore, the landmarks are used to assist the anthropometric measurement algorithm make required measurements on the body segment.

### Segment Isolation and

**Invalid Body Image Elimination**

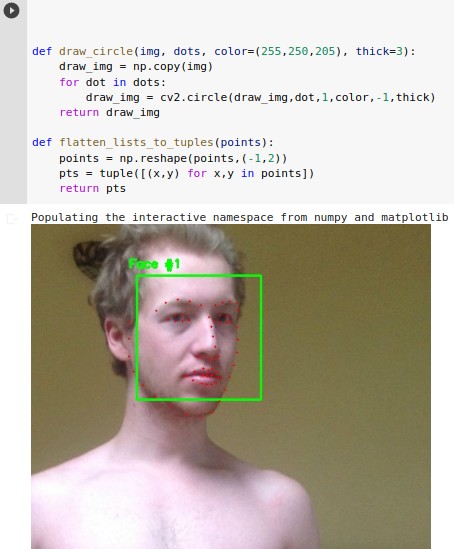
Since the [BCD](#_bookmark6) model using [CRF-RNN](#_bookmark12) is pre-trained to isolate several kinds of objects, the output may guess other objects in the background. To solve this the nose coordinates of the [SJD](#_bookmark7) algorithm are used to isolate the connected component that is identified as a human, these human pixels are labeled with ’0’, and all the remaining background pixels are labeled as ’1’. Since the used dataset represents the real world where images scraped from the website may be unsuitable for experimentation mainly due to occlusion that creates that absence of certain key body land-mark, we eliminate such images from the training set. Using the confidence scores that range from 0 - 1 provided by the [SJD](#_bookmark7) [4.2,](#_bookmark56) which tell how well the [SJD](#_bookmark7) algorithm is capable of detecting the co-ordinates mentioned in Figure [4.2,](#_bookmark56) an image that has a confidence value below 0.5 for any of the co-ordinates is considered to be suffering from occlusions. Such images are isolated and removed from the training dataset.

### Gray Scaling and Image Standardisation

The input to the [CNN](#_bookmark9) and the Transfer Learning models is the full-body image itself. Since the models are used to extract adiposity (fatness) features and colour of the image will not be necessary for the prediction of body weight, all the images are grayscaled. Furthermore, the input to these models needs to be of uniform dimension. Since the images are scraped from a website this could not be ensured during dataset collection. Hence the human Body is localized using the YOLO object detection and the bounding box is extended to the dimension of 59 x 106, as this was found to be the ideal dimension after multiple trial and error.

### Face Localisation and Facial Land-marking

Since the dataset used contains images of the entire human body, any [FLD](#_bookmark11) algorithm would benefit from being provided only the face as input to the landmark detector. Hence face detection is performed using a pre-trained VGG-16 model, this localized face is fed into the [FLD](#_bookmark11) model. The output of the landmarking is a total of 68 co-



**Figure 4.3:** Face Land-Marking

ordinates Figure [4.3.](#_bookmark60) These are further used to extract facial features that determine face adiposity which will be vital for the estimation of weight.

# Feature Extraction

### Calculating Facial Ratios

From the landmarks on the face, we can calculate the length of various features in a face such as a nose width, nose length, eye width, jaw width, face height, outer and inner lip widths, etc.

Since the length may not be the same depending on the distance from the subject ratio of the various extracted features to the jaw width was taken to remove the bias induced by the depth of the subject from the camera in the calculation of the length of the facial features.

### Anthropometric Measurements from Body

We can use the body segment returned by the [BCD](#_bookmark6) in conjunction with body landmarks to make measurements related to body dimensions by using pixel counting operations. Following are some examples.

* + - * Arm Width

The mid-point of the shoulder joint and elbow joint can be taken and a number of pixels on either side can be calculated. Since the orientation of the arm may differ for different images, the count is multiplied by the sine of the angle made by the connecting line.

* + - * Neck Width

Mid-point of mid-shoulders-point and nose can be taken and a number of pixels on either side can be calculated.

* + - * Hip Width

Mid-point of hip points can be taken and a number of pixels on either side can be calculated.

* + - * Waist Width

The mid-point of hip points and shoulder points can be taken and a number of pixels on either side can be calculated.

Due to the unreliability of the segmentation algorithm and the huge variance in the poses of bodies, at this stage, these features are not included for training the models.

### Final List of Features

Following facial distances (scaled down by jaw width) are considered.

* + - * Left-eye width
      * Right-eye width
      * Face height
      * Outer-lip width
      * Inner-lip width
      * Nose width
      * Nose length
      * Left-eyebrow width
      * Right-eyebrow width

Following body distances (scaled down by inter-shoulder-joint distance) are considered.

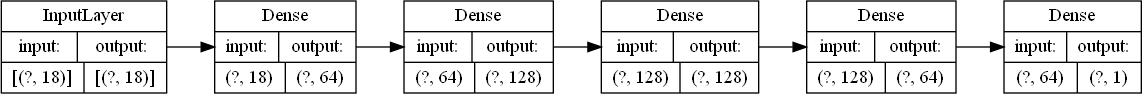
* + - * Left-shoulder to left-hip
      * Right-shoulder to right-hip
      * Left-hip to right-hip
      * Left-shoulder to right-hip
      * Right-shoulder to left-hip
      * Left-shoulder to left-elbow
      * Left-elbow to left-wrist
      * Right-shoulder to right-elbow
      * Right-elbow to right-wrist

# Model Training and Testing

All the models that are used for experimentation use the Visual-Body-to-BMI dataset with *a train-test split* of *70-30*. Additionally, a novel dataset collected by us for the experimentation containing 232 images of volunteers with the labeled body weight is also used for testing. These datasets are illustrated in Table [2.1.](#_bookmark31)

### Deep Learning Model

In prediction problems involving unstructured data (eg. images, text, etc.) sequential neural networks tend to outperform all other algorithms or frameworks. Hence, we have used this [DL](#_bookmark8) model Figure [4.4.](#_bookmark67) The library of choice is Keras which is built upon TensorFlow. Three approaches have been experimented with for training and testing



**Figure 4.4:** Deep Learning Model architecture

using both the Visual-Body-to-BMI dataset and the Novel Self-Created dataset. These three approaches are

* + - * extracted anthropometric features from full-body using Blazepose [SJD](#_bookmark7) and [BCD](#_bookmark6)
      * extracted facial features from localized face image using [FLD](#_bookmark11)
      * The combined model that uses both the features mentioned above

### XGBoost Regressor

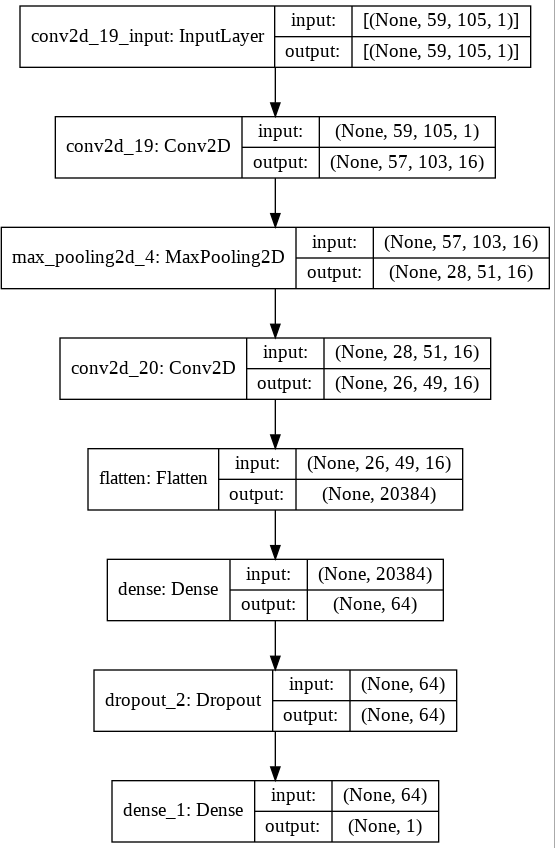
Since the problem of finding the weight, a continuous variable, requires regression also, XGBoost is a decision-tree-based ensemble machine learning algorithm that uses a gradient boosting framework. When it comes to small-to-medium structured/tabular data, decision tree-based algorithms perform at a much better level. It also has a perfect combination of software and hardware optimization techniques to yield superior results using fewer computing resources in the shortest amount of time. The XGBoost regressor also uses the three approaches that have been experimented with for training and testing using both the Visual-Body-to-BMI dataset and the Novel Self-Created dataset. These three approaches are

* + - * extracted anthropometric features from full-body using Blazepose [SJD](#_bookmark7) and [BCD](#_bookmark6)
      * extracted facial features from localized face image using [FLD](#_bookmark11)
      * The combined model that uses both the features mentioned above

Multiple regression models such as XGBoost, Adaboost, etc experimented with, out of which XGBoost performed the best. In the results section the best performing regression model ie. XGBoost is illustrated for the sake of brevity.

### CNN model

The pre-processed full-body images, on which gray scaling and image standardization is performed as described in the pre-processing module, are directly used as input to the [CNN](#_bookmark9) model. Like the Transfer, Learning approaches the [CNN](#_bookmark9) model is trained to identify features from the images that would aid in the estimation of Body Weight. Figure [4.5](#_bookmark70) illustrates the used [CNN](#_bookmark9) architecture



**Figure 4.5:** CNN model architecture

### Transfer Learning

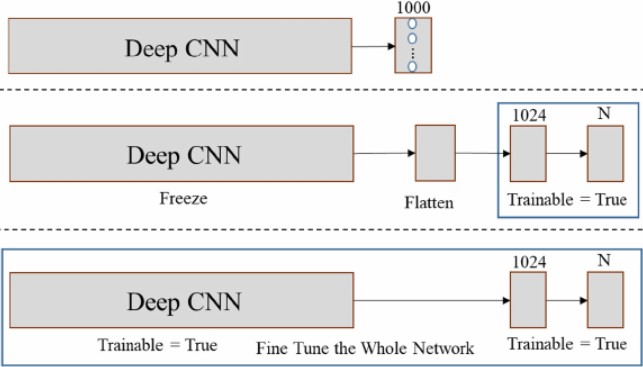
Three transfer learning approaches were experimented with

* + - * ResNet
      * XceptionNet
      * InceptionV3

These pre-trained models are trained on the ImageNet dataset and were among the best performing models. Previous approaches [(Haritosh et al., 2019)](#_bookmark100) for weight estimation on face images alone have shown good results using 2D face images in comparison

to other approaches that relied on additional features such as depth (from rgb-d sensors)(Barr [et al., 2018).](#_bookmark97)

In each of the approaches, the Deep [CNN’](#_bookmark9)s final layer has 1000 units since the

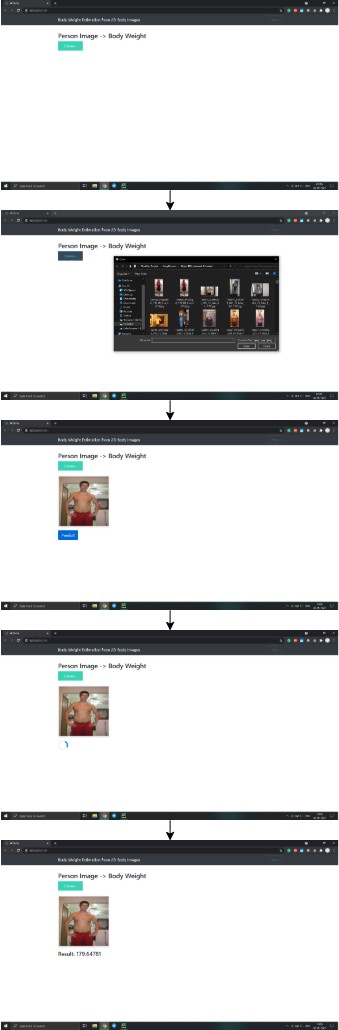


**Figure 4.6:** Transfer Learning model architecture

architectures were established on the ImageNet object classification task. So, the last layer (i.e., the last softmax layer) is truncated; thus, the last pooling layer becomes outputs of the network. These outputs are flattened into a vector, and two fully connected layers are added. The first fully connected layer has 1024 units and the last layer is linear as we are treating weight estimation as a Regression problem. This is illustrated in Figure [4.6.](#_bookmark72) These models are trained directly on the preprocessed images.

# Web Application

As a proof of concept, a web application is built using the *Flask* micro web framework. The Web App takes a person’s image as the input and estimates their body weight which is then displayed. We experimented with the model that performed the best among all *refer* Table [5.1,](#_bookmark81) which was the XGBoost regression model using the combined face and full-body anthropometric features. However, due to the computational complexity of the [BCD](#_bookmark6) approach which scans and labels the image pixel-by-pixel, the page is often rendered unresponsive. This necessitated a model that works directly on the image. The best model among these from Table [5.1](#_bookmark81) was the [CNN](#_bookmark9) model. Hence, it was the chosen model for Body Weight estimation. Thus, when a user clicks the Predict button on the web page the loaded [CNN](#_bookmark9) model is used to predict the bodyweight of the person in the image. The process flow is illustrated in the following figure Figure [5.1.](#_bookmark84)



**Figure 4.7:** Web Application Flow Screenshots

# Chapter 5

## RESULTS AND DISCUSSION

* 1. **Measurement Metric**

To measure the performance of the models, a common metric of Mean Absolute Error [(MAE)](#_bookmark10) is used, taken from the sklearn library. It is the most common metric preferred by previous experiments that have worked on estimation of weight that is cited below. Hence, using [MAE](#_bookmark10) would allow easy comparison of our results with previous approaches.

1

*MAE* =

*N*

*N*

*|Yk*

Σ ^

*k*=1

* *Yk*

*|* (5.1)

The calculation on [MAE](#_bookmark10) is done by taking the average of absolute errors between the Predicted Body Weight and the Ground Truth Body Weight. In Equation [(5.1),](#_bookmark78) N is the count of the total number of test images, *Yk* represents the Predicted Body weight and *Yk* represents the Ground Truth Body Weight.

^

# Model Approach-Wise Analysis

From Table [5.1](#_bookmark81) we can see that there are a total of 4 approaches that have been experimented with for the estimation of body weight from 2D full-body Images. These 4 approaches are

* + Deep Learning
  + Regression ML model
  + [CNN](#_bookmark9)
  + Transfer Learning using Deep [CNNs](#_bookmark9) Trained on Imagenet

In the subsequent sub-sections, the results for each of these approaches are discussed in detail.

Table 5.1: Model Test Results on Visual-Body-to-BMI dataset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S. N. | Approach | Model Input Feature | Model | MAE (kg) |
| 1. | Deep Learning | Anthropometric Features from Full Body Image | Deep Learning (Sequential)  [64*,* 128*,* 128*,* 64] \* relu \* linear | 18.37 |
| 2. | Extracted Facial Feature Measurements from  Localized Face in Full Body Image | Deep Learning (Sequential)  [8*,* 16*,* 8] \* relu + [1] \* linear | 22.67 |
| 3. | Combined Features from Full Body and Face | Deep Learning (Sequential)  [64*,* 128*,* 128*,* 64] \* relu + [1] \* linear | 17.02 |
| 4. | Regression ML model XGBoost | Anthropometric Features from Full Body Image | XGBoost Regressor n\_estimators=6 | 18.46 |
| 5. | Extracted Facial Feature Measurements from Localised Face in Full Body Image | XGBoost Regressor n\_estimators=100 | 18.58 |
| 6. | Combined Features from Full Body and Face | XGBoost Regressor n\_estimators=6 | 16.65 |
| 7. | CNN | Full Body Image | CNN  [16*,* (3*,* 3)]\*Conv2D+[2*,* 2]  \* MaxPooling+[16*,* (3*,* 3)]\*Conv2D  +Flatten+[64]\*Dense+[0*.*1]\*Dropout  +[64] *∗ Dense* | 18 |
| 8. | Transfer Learning using Deep CNNs  Trained on Imagenet | Full Body Image | Resnet archiecture[Freeze]  +Flatten+[1024]\*Dense  +[1]\*Dense | 18.7 |
| 9. | Full Body Image | XceptionNet architecture  [*Freeze*]+Flatten  +[1024]\*Dense+[1]\*Dense | 19.7 |
| 10. | Full Body Image | InceptionV3 architecture[*F reeze*]  +Flatten+[1024]\*Dense+[1]\*Dense | 18.94 |

### Deep Learning and XGboost models

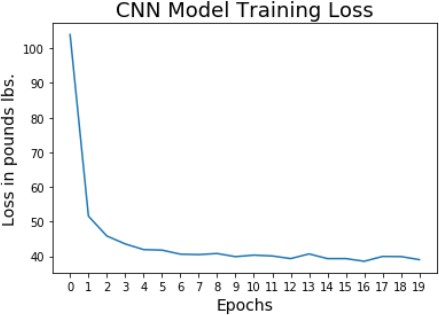
The [DL](#_bookmark8) and XGBoost models were experimented with three combinations of anthropometric features, with only torso features, with only facial features, and a combined model using both the features. This was necessary not only because the Face and Body Landmarking algorithms were different but also to validate if the face features enhance the predictions made.

Out of the three combinations, the model which used only face features performs the worst, this can be explained by the fact that the face has to be localized from the full-body image, and hence there is a reduction in image quality.

The models that work on only the body features using a combination of [SJD](#_bookmark7) and [BCD](#_bookmark6) perform comparably to [(Jiang and Guo, 2019).](#_bookmark101) The inclusion of face features im- proves the overall performance for both the [DL](#_bookmark8) and XGboost models. The XG- Boost model uses a combination of both features performing the best among all the model approaches with the lowest [MAE](#_bookmark10) of 16.65 Kilo Grams [(Kg).](#_bookmark13)

### CNN

The [BCD](#_bookmark6) required in the previous two approaches is computationally complex and slow to render an output given that there are two more stages to those approaches. Hence, the [CNN](#_bookmark9) models that work directly on the image itself experiment. Among the approaches that used the same Model input feature ref Table [5.1,](#_bookmark81) the [CNN](#_bookmark9) model had the least [MAE](#_bookmark10) of [18Kg](#_bookmark13) and had the best performance. Figure [5.1](#_bookmark84) shows the training loss for the [CNN](#_bookmark9) model.

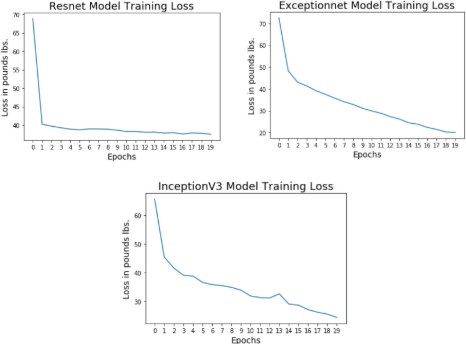


**Figure 5.1:** CNN model training loss

### Transfer Learning using Deep CNNs Trained on Imagenet

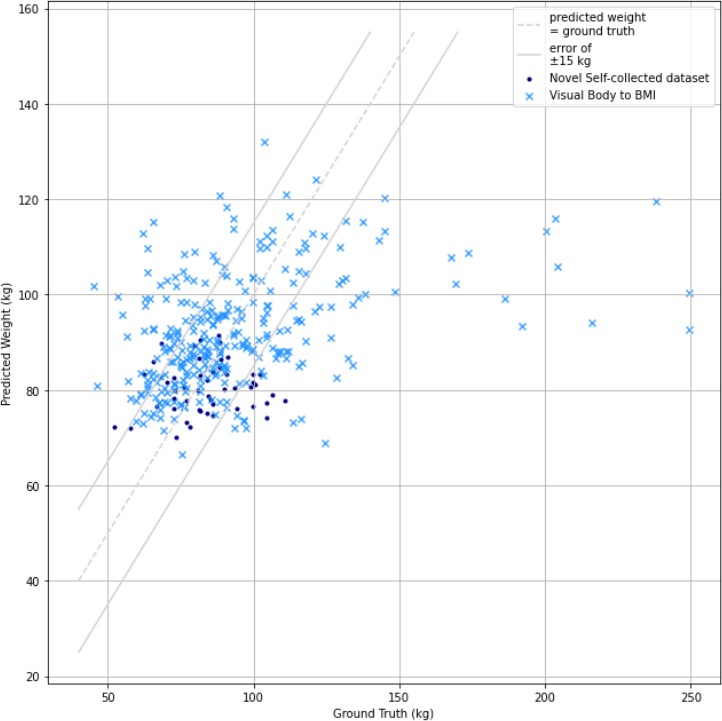
The work done by [Haritosh et al. (2019),](#_bookmark100) using Transfer learning for the estimation of weight on 2D Face Images, showed good results that were comparable to previous approaches that relied on additional features such as depth (from rgb-d sensors) [Barr et al.](#_bookmark97) [(2018).](#_bookmark97) This gave the motivation to experiment with the approach on full-body images, hoping that the increase in the features available to the model ie. the entire body instead of just the face would provide a trained model with an [MAE,](#_bookmark10) that is under the permissible limits for the applications proposed in the introduction section. Figure [5.5](#_bookmark94) shows the training loss for all three models that have been experimented with.

Out of the three approaches, the Resnet model performed the best with an [MAE](#_bookmark10) of



**Figure 5.2:** Transfer Learning training loss curves

18.7 [Kg](#_bookmark13) and the XceptionNet model performing the worst among all the models and approaches that were experimented with.



**Figure 5.3:** Ground Truth vs Prediction Scatterplot

# Graphical analysis

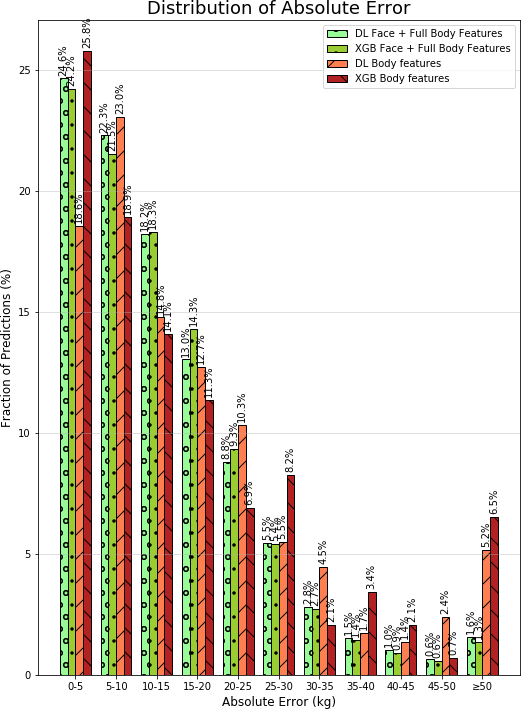
### Ground Truth vs Prediction Scatterplot

The data points in Figure [5.3,](#_bookmark88) show the distribution of the ground truth vs predicted body weight of both the test split of the Visual-Body-to-BMI dataset as well as on the Novel test dataset that was created for the purpose of testing the models. The best performing XGBoost is the model of choice for testing. The dotted line represents that *predicted weight = ground truth weight*. Since most of the data points are clustered within close proximity of the line on both the testing datasets, it can be said that the model is quite accurate at the estimation of body weight. The two solid lines depict an error of *±* 15 [Kg](#_bookmark13) which is considered within permissible limits for the applications mentioned in the Introduction Section.

However, from Figure [5.3](#_bookmark88) it is clear that the model fails in cases of heavily overweight

and underweight subjects whose weight is greater than 120 [Kg](#_bookmark13) and less than 50 [Kg.](#_bookmark13)

### Error Distribution Histogram



**Figure 5.4:** Error Distribution Histogram

The Absolute error distribution on testing with the Visual-Body-to-BMI dataset is illustrated in Figure [5.4.](#_bookmark91) Since both the [DL](#_bookmark8) and XGBoost models had a similar performance on the Visual-Body-to-BMI dataset.

The Histograms for both types of models

* + - * Using only Body features
      * Using the combined Body and face features

show how the inclusion of Face features drastically decreases the Fraction of Predictions that have an error over 35 [Kg](#_bookmark13) this is mostly in the case of overweight subjects and can be explained by the wider nature of the face the contributes to the higher face adiposity in the feature extraction phase which would aid determining the bodyweight of subjects that are overweight.

# Testing on Novel Self-created dataset for external model validation

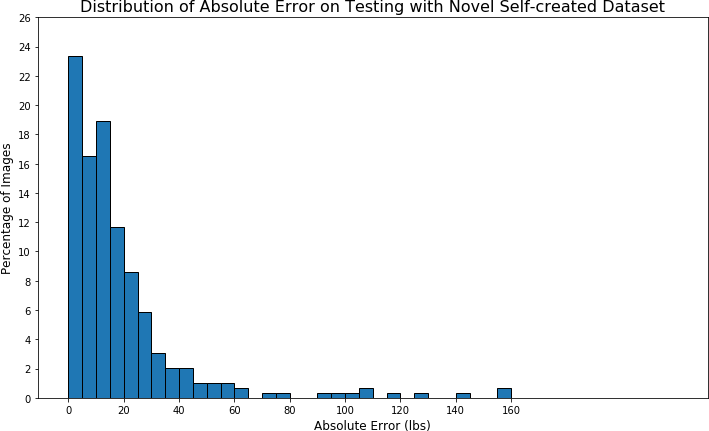
In addition to testing on the Visual-Body-to-BMI dataset using a train-test split. Test- ing is also performed on a Novel dataset that was created by collecting Photos from Volunteers and labeling them with the Volunteer’s body weight. This is done to check the robustness of the model on external real-world data. The details are provided in Table [5.2.](#_bookmark93)

The best performing model from Table [2.1,](#_bookmark31) XGBoost on a Combination of the face and full body features are chosen for testing and a test [MAE](#_bookmark10) of 17.8 [Kg](#_bookmark13) is obtained which is comparable to the test on the test split of the Visual-Body-to-BMI dataset. Figure [5.3](#_bookmark88) shows the ground truth versus predicted value in the form of a scatterplot with the ’.’ legend showing the distribution of the Novel dataset data points. Figure [5.5](#_bookmark94) shows that most of the test dataset had errors within 20 pounds or 9 [Kg](#_bookmark13) which is a good estimate for most use cases.

This comprehensively validates that the model will work on real-world data and can be used in various real-world applications.

Table 5.2: Model Results on Novel Self-Collected Test dataset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset | Approach | Model Input Feature | Model | MAE(kg) |
| Novel Self-Collected  Full Body Image Dataset | Regression ML model XGBoost | Combined Extracted Features from  Full Body and Face | XGBoost Regressor n\_estimators = 6 | 17.8 |



**Figure 5.5:** Distribution of Absolute error on Testing with Novel dataset

# Chapter 6

## CONCLUSION

By obtaining a mean absolute error of about 16.65 [Kg](#_bookmark13) in finding the weight of a subject from the best case model using full-body images, it can be safely said that this method of weight estimation is a viable alternative to find the weight of a person in situations where the actual weight of the person cannot be physically measured. Therefore, it can hence be useful to have a rough estimate of a person’s weight in several use cases such as in airports to detect the approximate total weight of all the passengers in a flight, forensics to identify people by adding more features than just an image such as a fat man, thin man, etc. and also for remotely practicing physicians to measure such a key human metric.

Challenges and limitations:

* Currently, API provided by MediaPipe for SJ[D](#_bookmark7) only provides landmarks for up- per body as it is still a work in progress with the first version released in 2020 [(Bazarevsky et al., 2020a).](#_bookmark98) Hence features such as the width of the thigh, which is a good indicator of adiposity (fatness) could not be extracted.
* The model fails under extreme bodyweight conditions which include extreme overweight (over 100 [Kg)](#_bookmark13) and extreme underweight (under 50 [Kg)](#_bookmark13)
* Since age is not taken as a feature the model doesn’t work well on images of Children below the age of 14 as their body adiposity (fatness), which though correlates to bodyweight belongs to a different range of body weight.
* The images must contain the image of a single person as person recognition hasn’t been incorporated as it added too much complexity.

# Chapter 7

## FUTURE ENHANCEMENT

* BlazePose - Joint Detection

Currently, API provided by MediaPipe only provides landmarks for the upper body as it is still a work in progress with the first version released in 2020 [(Bazarevsky](#_bookmark98) [et al., 2020a).](#_bookmark98) MAE might improve when it is updated to provide other landmarks.

* Improvement in dataset quality

Since the Visual Body to BMI dataset is scraped from the internet, images are extremely varying and a lot of them have low resolution, resulting in relatively high MAE compared to face images that are taken at a close range with a higher resolution.

* Anthropometric features

Once the BCD and BlazePose are improved, a full set of anthropometric measurements can be extracted better.

* Incorporating more soft-biometric features

Since all experimentation was done using only the 2D image with the body weight as a target. Since features like age, gender and ethnicity correlate with body weight and can create a range for a possible bodyweight of a person. Incorporating prediction models that are publicly available, which can predict

* + age
  + gender
  + ethnicity

from the 2D image. can serve as additional features to the bodyweight estimation model.

* Including More people in Image, The images that have been experimented with contain the image of only a single person. This limitation can be alleviated by labeling unique people in an image using person recognition and thus allow multiple people in the image frame/

# REFERENCES

1. Barr, M., Guo, G., Colby, S., and Olfert, M. (2018). “Detecting body mass index from a facial photograph in a lifestyle intervention.” *Technologies*, 6(3), 83.
2. Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang, F., and Grundmann,

M. (2020a). “Blazepose: On-device real-time body pose tracking.

1. Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang, F., and Grundmann,

M. (2020b). “Blazepose: On-device real-time body pose tracking.” *arXiv preprint arXiv:2006.10204*.

1. Haritosh, A., Gupta, A., Chahal, E. S., Misra, A., and Chandra, S. (2019). “A novel method to estimate height, weight, and body mass index from face images.” *2019 Twelfth International Conference on Contemporary Computing (IC3)*. 1–6.
2. Jiang, M. and Guo, G. (2019). “Bodyweight analysis from human body images.” *IEEE Transactions on Information Forensics and Security*, 14(10), 2676–2688.
3. Molarius, A. and Seidell, J. (1998). “Selection of anthropometric indicators for classification of abdominal fatness - a critical review.” *International journal of obesity and related metabolic disorders: journal of the International Association for the Study of Obesity*, 22, 719–27.
4. Nguyen, T. V., Feng, J., and Yan, S. (2014). “Seeing human weight from a single rgb-d image.” *Journal of Computer Science and Technology*, 29(5), 777–784.
5. Pfitzner, C., May, S., and Nüchter, A. (2017). “Evaluation of features from rgb-d data for human body weight estimation.” *IFAC-PapersOnLine*, 50(1), 10148 – 10153. 20th IFAC World Congress.
6. Santner, K., Rüther, M., Bischof, H., Skrabal, F., and Pichler, G. (2009). “Human body volume estimation in a clinical environment.
7. Schneider, T. M., Hecht, H., and Carbon, C.-C. (2012). “Judging body weight from faces: The height-weight illusion.” *Perception*, 41(1), 121–124. PMID: 22611670.
8. Seidell, J., Oosterlee, A., Thijssen, M., Burema, J., Deurenberg, P., Hautvast, J., and Ruijs, J. (1987). “Assessment of intra-abdominal and subcutaneous abdominal fat: Relation between anthropometry and computed tomography.” *The American journal of clinical nutrition*, 45, 7–13.
9. Sánchez Hernández, S., Romero, H., and Morales, A. (2020). “A review: Comparison of performance metrics of pre-trained models for object detection using the TensorFlow framework." *IOP Conference Series: Materials Science and Engineering*, 844, 012024.
10. Tai, C. and Lin, D. (2015). “A framework for healthcare everywhere: Bmi prediction using Kinect and data mining techniques on mobiles.” *2015 16th IEEE International Conference on Mobile Data Management*, Vol. 2. 126–129.
11. Velardo, C. and Dugelay, J.-L. (2010). “Weight estimation from visual body appearance. 1 – 6.
12. Wang, L., Wan, T., Tang, W., Zhu, Y., and Wu, T. (2017). “An efficient human body contour extraction method for mobile apps. 173–181.
13. Wen, L. and Guo, G. (2013). “A computational approach to body mass index prediction from face images.” *Image and Vision Computing*, 31(5), 392 – 400.
14. Windhager, S., Bookstein, F. L., Millesi, E., Wallner, B., and Schaefer, K. (2017). "Patterns of correlation of facial shape with physiological measurements are more integrated than patterns of correlation with ratings.” *Scientific Reports*, 7(1), 45340.
15. Zhang, L., Li, H., Shen, P., Zhu, G., Song, J., Shah, S. A. A., Bennamoun, M., and Zhang, L. (2018). “Improving semantic image segmentation with a probabilistic superpixel-based dense conditional random field.” *IEEE Access*, 6, 15297–15310.
16. Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., and Torr, P. H. S. (2015a). “Conditional random fields as recurrent neural networks.” *2015 IEEE International Conference on Computer Vision (ICCV)*. 1529–1537.
17. Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., and Torr, P. H. S. (2015b). “Conditional random fields as recurrent neural networks.” *International Conference on Computer Vision (ICCV)*.