PES UNIVERSITY
(Established under Karnataka Act No. 16 of 2013)
100-ft Ring Road, Bengaluru – 560 085, Karnataka, India

A Project Report
On

Classification of images based on BMI

Submitted in fulfillment of the requirements for the Project phase -1

Submitted by

SHAMANTH HS
SRN: PES2PGE20DS391

Under the guidance of
Mr. Hayavadan G Nargund
Professor

FEBRUARY-APRIL 2022

FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

PROGRAM M.TECH

FACULTY OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

PROGRAM M.TECH

CERTIFICATE

Classification of image based on BMI

This is to certify that the Dissertation entitled

is a bonafide work carried out by

SHAMANTH H S
SRN: PES2PGE20DS391

In partial fulfillment for the completion of 3rd semester course work in the Program of Study MTECH in Computer Science and Engineering under rules and regulations of PES University, Bengaluru during the period February 2022 – April 2022. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report. The project report has been approved as it satisfies the 3rd semester academic requirements in respect of project work.

Signature with date & Seal	*Signature with date & Seal*	*Signature with date & Seal*
Internal Guide	*Chairperson*	*Dean of Faculty*

Name and Signatures of the Examiners

1.

2.

3.

TABLE OF CONTENTS

TABLE OF FIGURES and TABLES

1 INTRODUCTION

The human face reveals information about a person's identity, personality, manner, as well as traits such as gender, age, and ethnicity. Facial recognition has received the most attention in the field of biometrics. Attributes or soft biometrics such as gender, age, height, and weight have lately acquired appeal because to their semantic interpretation, i.e., they may offer a description that people can easily understand; for example, "young, female, tall." There has been little focus on the relationship between the human face and bodily features such as body height and weight, and much less on the automated extraction of such.

Estimating body height, weight, and BMI is necessary for numerous reasons. For starters, height and weight are often utilised features in surveillance, forensics, re-identification applications, and image retrieval systems. Second, height and weight are key and evident traits used by humans to verbally characterise a person, and are frequently utilised in police reports, as opposed to standard biometrics, which may be insufficient, as Klontz and Jain demonstrated in the case of the 2013 Boston bombs. Third, in automated biometric systems, body weight and height have been considered as soft biometric features. Fourth, weight is an important health indicator, and being overweight has been linked to obesity, diabetes, and cardiovascular disease. The given technique contributes to the current trend of image-based automated self-diagnosis in this area. Finally, this work can help to further human metrology studies in psychology.

A background on the project is provided in Section 1.1. A concise problem statement is described in Section 1.2, Section 1.3 briefly describes the steps involved during object tracking, and Section 1.4 introduces the scope and methods used in this study.

1.1 Background

Body mass index (BMI) is a number calculated from a person's weight and height. The BMI is calculated by dividing the body mass by the square of the body height and is given in kilogrammes per square metre (kg/m2).

A table[a] or chart that presents BMI as a function of mass and height using contour lines or colours for different BMI categories and may utilise other units of measurement can be used to calculate BMI (converted to metric units for the calculation).

The BMI is a simple rule of thumb that may be used to classify a person as underweight, normal weight, overweight, or obese based on tissue mass (muscle, fat, and bone) and height. Underweight (less than 18.5 kg/m2), normal weight (18.5 to 24.9), overweight (25 to 29.9), and obese are the four major adult BMI classifications (30 or more). When used to forecast an individual's health rather than as a statistical measurement for groups, the BMI has limits that make it less effective than other alternatives, particularly when used to persons with abdominal obesity, low height, or extremely high muscle mass.

BMIs less than 20 and greater than 25 have been linked to increased all-cause mortality, with the risk rising as one moves away from the 20-25 range. Belgian astronomer, mathematician, statistician, and sociologist Adolphe Quetelet. Between 1830 and 1850, he constructed what he dubbed "social physics," which served as the foundation for the BMI.

[3] Ancel Keys and colleagues established the present term "body mass index" (BMI) for the ratio of human body weight to squared height in a study published in the July 1972

issue of the Journal of Chronic Diseases. Keys stated in this work that the BMI was "if not completely excellent, at least as good as any other relative weight index as a predictor of relative obesity.

"The interest in a body fat index arose as a result of rising obesity in rich Western nations." Keys specifically said that BMI is useful for population research but not for individual evaluation. Nonetheless, due to its ease of use, it has become commonly utilised for preliminary diagnostics. Additional measurements, such as waist circumference, might be more helpful.

The BMI is calculated by multiplying mass in kilogrammes by height in metres. When using pounds and inches, a conversion ratio of 703 (kg/m2)/(lb/in2) is utilised. The units are generally removed when the word BMI is used casually.

BMI is a simple numerical assessment of a person's thickness or thinness, allowing health practitioners to address weight issues with their patients in a more objective manner. As of 2014, the BMI value guidelines for such persons are as follows: 18.5 to 24.9 kg/m2 may be considered ideal weight, less than 18.5 may be considered underweight, 25 to 29.9 may be considered overweight, and 30 or more may be considered obese. Lean male athletes frequently have a high muscle-to-fat ratio, resulting in a BMI that is misleadingly high in comparison to their body-fat proportion.A concise statement of the problem is provided in the following section.

1.2 Problem Statement

BMI (body mass index) is a measurement of the relationship between weight and height of an individual. It is commonly used to assess how far an individual's weight deviates from normal or desirable for the individual's height. This excess or deficiency in weight may be due to a variety of factors. It is believed that body fat (adipose tissue) accounts for part of the excess or deficiency in weight although other factors, such as muscle mass, also affect BMI significantly.

The technique is used in many applications such as surveillance, re-identification, and image retrieval, as well as healthcare. Previous papers have concentrated on calculating BMI for images that are of a complete human size, or 3D images.

So, we present here a novel dataset consisting of more than 1000(both male and female images) subjects and show results calculating BMI just from facial images. The proposed method relies on the RESNET, VGG-19,INCEPTION NET AND EFFICIENTNET B0.A good BMI algorithm would use the various features of facial images to get an accurate representation of the body's size. The best algorithm would take into account the various features of facial images to obtain the appropriate BMI. This would suggest that facial images contain inequitable information comparable to that of body-images and videos.

2 LITERATURE SURVEY

Use of Convolutional Neural Networks for Computer Vision is fairly new. Once begun, the most important aspect from their use for image processing was automatic feature extraction. Earlier, features were mostly hand-crafted, and as such, rarely transferable across different applications. There were some algorithms and operations that were generic, such as edge, corner and contour detection, but these were only building blocks, and higher-level feature extraction mechanisms, through the use of CNNs made image processing more easily accessible to the research community.

2.1 What else does your biometric data reveal? A survey on soft biometrics

Recent research has explored the possibility of extracting ancillary information from primary biometric traits, viz., face, fingerprints, hand geometry and iris. This ancillary information includes personal attributes such as gender, age, ethnicity, hair color, height, weight, etc. Such attributes are known as soft biometrics and have applications in surveillance and indexing biometric databases. These attributes can be used in a fusion framework to improve the matching accuracy of a primary biometric system (e.g., fusing face with gender information), or can be used to generate qualitative descriptions of an individual (e.g., "young Asian female with dark eyes and brown hair"). The latter is particularly useful in bridging the semantic gap between human and machine descriptions of biometric data. In this paper, we provide an overview of soft biometrics and discuss some of the techniques that have been proposed to extract them from image and video data. We also introduce a taxonomy for organizing and classifying soft biometric attributes, and enumerate the strengths and limitations of these attributes in the context of an operational biometric system. Finally, we discuss open research problems in this field. This survey is intended for researchers and practitioners in the field of biometrics

2.2 Deciphering faces: Quantifiable visual cues to weight,

Body weight plays a crucial role in mate choice, as weight is related to both attractiveness and health. People are quite accurate at judging weight in faces, but the cues used to make these judgments have not been defined. This study consisted of two parts. First, we wanted to identify quantifiable facial cues that are related to body weight, as defined by body mass index (BMI). Second, we wanted to test whether people use these cues to judge weight. In study 1, we recruited two groups of Caucasian and two groups of African participants, determined their BMI and measured their 2-D facial images for: width-to-height ratio, perimeter-to-area ratio, and cheek-to-jaw-width ratio. All three measures were significantly related to BMI in males, while the width-to-height and cheek-to-jaw-width ratios were significantly related to BMI in females. In research 2, Caucasian viewers graded the perceived weight of these pictures. We demonstrated that these viewers employ all three signals to estimate weight in both African and Caucasian faces. These three facial signals, width-to-height ratio, perimeter-to-area ratio, and cheek-to-jaw-width ratio, are thus not only associated with real weight but also serve as a foundation for perceptual qualities.

2.3 A computational approach to body mass index prediction from face images

Human faces provide a wealth of information. Recent research in psychology and human perception has discovered that face traits are related to human weight or BMI (BMI). These research are looking for links between face characteristics and BMI. Motivated by current psychology studies, we create a computer technique for automatically predicting BMI from facial photos. We model BMI prediction from facial traits as a machine vision issue and test our method on a huge collection of over 14,500 face photos. A positive result was produced, demonstrating the viability of constructing a large-scale computing system for BMI prediction using facial photos.

2.4 Show me your face and I will tell you your height ,weight and body mass index

Body height, weight, and the related and composite body mass index (BMI) are important human traits since they are used in a variety of applications such as surveillance, re-identification, image retrieval systems, and healthcare. Previous work on automated height, weight, and BMI estimates has mostly focused on 2D and 3D fullbody photos and movies. The use of the face for evaluating such features has received little study. Motivated by the foregoing, we propose a regression approach based on the 50-layer ResNet-architecture for predicting height, weight, and BMI from single-shot face photos. Furthermore, we provide a fresh dataset of 1026 people and exhibit results indicating that facial photographs, like body photos and videos, contain discriminating information about height, weight, and BMI. Finally, we conduct a gender-based study of height, weight, and BMI prediction.

3 Analysis of the System

3.1 Study of the feasibility

The assessment of its attainability in terms of information, yield, projects, and procedures is dependent on the system prerequisite's design plan. Following the definition of a model system, the analysis will continue to propose the type of equipment required to develop the framework, as well as the approach required to operate the framework once it has been designed. The project is enlarged to the point where the essential capacities and execution are met within the constraints. The task is produced by the most recent invention. Despite the fact that the technology may become obsolete after a given period of time owing to the fact that it never creates more advanced versions in the same programming, the system is still in use. The system was created using the convolution neural networks technique. The system's formation is aided by price and benefit. Criteria for ensuring that effort is directed toward the most timely return, which will yield the best results. Because the system is developed as part of the venture activity, the suggested structure does not necessitate any manual expense. Indeed, as easily available as each of the qualities is, it indicates that the system is economically possible for improvement. The project would be useful because, if completed and deployed, it would achieve the goals. Each and every aspect of behaviour is carefully examined and suggests that the task is usually feasible..

3.2 FUNCTIONAL REQUIREMENTS

- Useful requirements define the internal actions of the product, such as technological nuances, data monitoring and management, and other particular functions demonstrating how to support the use cases. They are supported by non-functional requirements that compel the planning or execution of imperatives.
- System should Process the data.
- System should detect the lesion.
- System should predict the lesion.
- System should check the level of Melanoma.

Data collection: The datasets used in this project is the data collected from reliable websites and sources. The sources of the datasets are from government entity for crop yielddata, Kaggle for crop recommendation and pest identification. The crop recommendation consists of names of the crops, temperature, soil ph, humidity average rainfall (mm), soil nutrient contents such as nitrogen, phosphorus and potassium. The other dataset contains State name, season, soil type, area(in hectares),crop and Production(in tons).

Data Preprocessing: The purpose of preprocessing is to convert raw data into a form thatfits machine learning. Structured and clean data allows a data scientist to get more preciseresults from an applied machine learning model. The technique includes data formatting, cleaning, and sampling. Here, data pre-processing focuses on finding the attributes with null values or invalid values and finding the relationships between various attributes as well. Data Pre-processing also helps in finding out the impact of each parameter on the target parameter. To preprocess our datasets, we used EDA methodology. All the invalid and null values were handled by removing that record or giving the default value of that particular attribute based on its importance.

Dataset splitting: A dataset used for machine learning should be partitioned into two subsets —

training and test sets. We split the dataset into two with a split ratio of 80%i.e., in 100 records 80 records were a part of the training set and remaining 20 records were a part of the test set.

Model training may begin when a data scientist has preprocessed the obtained data and divided it into train and test sets. This procedure comprises supplying training data to the algorithm. An algorithm will analyse data and produce a model capable of locating a target value (attribute) in fresh data to provide the answer you want in a predictive analysis. The goal of model training is to create a model. We used the random forest approach to train our model. When the model is trained, it predicts the yield when given the other properties of the dataset as input.

Model review and testing: The purpose of this stage is to create the simplest model capable of formulating a target value quickly and accurately. This may be accomplished by a data scientist through model adjustment. That is the optimization of model parameters to attain the optimal performance of an algorithm.

3.3 NON-FUNCTIONAL REQUIREMENTS

- Unnecessary prerequisites are requirements that indicate parameters to be utilised to evaluate the operation of a framework rather than particular actions. This is distinct from useful needs that indicate explicit behaviour or capability. Non-practical requirements include dependability, adaptability, and cost. System ileitis is a term used to describe non-practical preconditions. Non-practical needs are sometimes known as limits, "quality traits," and "administrative requirements." If any unusual circumstances arise during product execution, they should be obtained and kept the framework from crashing in this manner. The architecture should be designed such that additional modules and features may be added, supporting application development. Because programming packages are publicly available, the cost should be low.
- Usability
- Reliability
- Performance
- Supportability

3.4 System Environment

The entire system will be run on a laptop computer

3.4.1 Hardware Requirements

- The hardware requirements specify the physical computer resources required for a system to function properly. The hardware requirements may serve as the foundation for a contract for system implementation and should thus be a comprehensive and consistent definition of the entire system. The following are the hardware requirements:
- • **Processor:** An integrated electronic circuit that conducts the computations that allow a

computer to function. A processor executes arithmetic, logical, input/output (I/O), and other fundamental commands sent by an operating system (OS). Most other processes are dependent on processor activities. A minimum 1 GHz processor should be used, although we would recommend S2GHz or more. A processor includes an arithmetical logic and control unit (CU), which measures capability in terms of thefollowing:

Ability to process instructions at a given time

Maximum number of bits/instructions

Relative clock speed

The proposed system requires a 2.4 GHz processor or higher.

- **Memory (RAM):** Random-access memory RAM is a type of computer data storage that holds current data and machine code. A random-access memory device enables data objects to be read or written in about the same amount of time, regardless of their physical position within the memory. Random-access memory is now available in the form of integrated circuits. RAM is typically linked with volatile kinds of memory (such as DRAM modules), in which recorded information is lost when power is disconnected, while non-volatile RAM has also been produced. For the suggested system, a minimum of 4 GB RAM is recommended.

The minimum requirements to run the software developed as part of this study issummarized in Table 3-1.

Processor	AMD Ryzan-5, 3.2GHz
No. of CPUs	4 (2 physical CPUs, with multi-threading enabled)
RAM	8GB
Cache	L1 Cache: 32KB, L2 Cache: 256KB, L3 Cache: 3MB

Table 3-1 Hardware requirements for the proposed system

3.5 TOOLS AND TECHNOLOGY REQUIRED HARDWARE AND SOFTWARE REQUIREMENTS HARDWARE

- System : intel i3/i5 2.4 GHz.
- Hard Disk : 500 GB
- Ram : 4/8 GB SOFTWARE
- Operating system : Windows XP / 7/10
- Coding Language : Python
- Software : Anaconda
- Language : Python3

3.6 SOFTWARE IMPLEMENTATION

- OPEN CV
- OpenCV-Python could be a Python linking library designed to unravel issues with pc vision. Python could be a general artificial language started by Guido van Rossum, which quickly became very talked-about, primarily thanks to its simplicity and readability of code. It permits the computer user to precise ideas in fewer code lines while not decreasing readability. Open CV application areas include: • Toolkits 2D and 3D • Recognition of the ego phenomenon • Recognition of the face • Recognition of leadership • Robot mobile • Segmentation and Recognition • Enhanced Perception and Movement Detection
- PYTHON 3.5.0
- It is an enormous programming language of deciphered standard of generally useful programming. Python has organized thinking that emphasizes the lucidity of code and syntax that empowers designers to express their thoughts in fewer lines of code, using the basic void zone prominently. It gives assemblies that participate in both small and large scales of simple programming. Python combines the board's awesome type of system and personalized memory. It supports different programming, ideal models, including organized objects, fundamental, useful and procedural, and has a large and wide standard library. For some operating systems, Python Interpreters are available. Generally, Python use joins a read-eval-print hover, allowing the customer to function as an immediate line interpreter for which declarations are entered successively and \ results are immediately obtained. Some things that Python is often used for are:
- • Creation of the web page.
- • Scientific timetable.
- • GUIs for the desktop.
- • Programming of the network.
- • Programming of the game.

3.7 Software Requirements

- The software requirements are description of features and functionalities of the target system. Requirements convey the expectations of users from the software product. The requirements can be obvious or hidden, known or unknown, expected or unexpected fromclient ‘s point of view.
- **Jupyter Notebook:** The Jupyter Notebook is an open- source web application that you canuse to create and share documents that contain live code, equations, visualizations, and text. Jupyter ships with the IPython kernel, which allows you to write your programs in Python, but there are currently over 100 other kernels that you can also use.

The notebook web application is an interactive online application that allows you to write and run code while also producing notebook papers.

Kernels: Independent processes launched by the notebook web application that run users' code in a specified language and provide results to the notebook web application. The kernel also supports interactive widget calculations, tab completion, and introspection.

Notebook documents: Self-contained documents that contain a representation of all material viewable in the notebook web application, such as computation inputs and outputs, narrative prose, equations, photos, and rich media object representations. Each laptop has its own kernel.

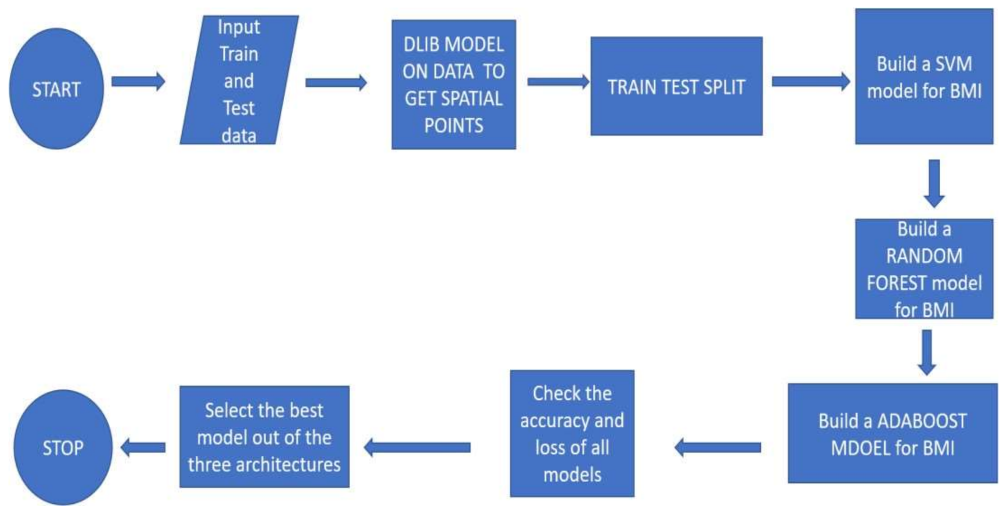
Fig 3-2 Jupyter Notebook

- **Python:** It is an object-oriented, high-level programming language with dynamic semantics that is designed largely for web and app development. It is quite appealing in the field of Rapid Application Development since it supports dynamic type and dynamic binding. Python is a reasonably basic language that requires a distinctive syntax that focuses on readability. Python code is significantly easier to understand and translate for developers than code in other languages. As a result, the cost of programme maintenance and development is reduced since teams may collaborate without large language and experience barriers. Furthermore, Python permits the usage of modules and packages, allowing applications to be developed in a modular fashion and code to be reused across a number of projects.

Fig 3-3 Python

4 Proposed Methodology

Flowchart:



The above flow chart is very self explanatory.

Firstly input the train and test data. Perform the DLIB model on it and get the facial points. Preprocessing is not required. So continue with train test split. Build the respective models and find the accuracy of it. Perform the optimisations on each model.

5 Proposed Solution

5.1 Machine Learning

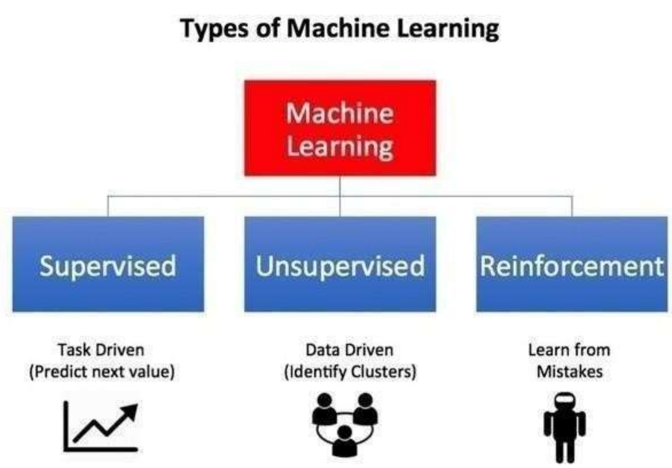
- Machine Learning is certainly one of today's most significant and powerful technologies. Machine learning is a technology for converting data into knowledge. There has been a data explosion in the last 50 years. This massive amount of data is meaningless until we analyse it and discover the underlying patterns. Machine learning techniques are used to identify useful underlying patterns in complicated data that would otherwise be difficult to find. Hidden patterns and information about a situation can be utilised to forecast future events and make difficult decisions. Machines must travel to understand the laws that govern a phenomena.

- Through a process of experimenting with new rules and learning from how effectively they perform. A dataset is a collection of data examples that have attributes critical to addressing the problem.

- Features: Important pieces of data that help us understand a problem. These are fed into a Machine Learning algorithm to help it learn.

- Model: A representation (internal model) of a phenomena learned by a Machine Learning algorithm. It learns this from the data presented to it during training. The model is the result of training an algorithm. A decision tree algorithm, for example, might be taught to generate a decision tree model..

5.2 Types of Machine Learning



Machine Learning can be supervised, unsupervised, semi-supervised, or reinforcement learning. Each technique to Machine Learning is different, yet they all follow the same fundamental process and philosophy.

Fig 5-1: Types of Machine Learning Algorithms

- **Supervised Learning:** It is the most widely used machine learning paradigm. Given data in the form of examples with labels, we may feed these example-label pairs to a learning algorithm one by one, enabling the system to predict the label for each case and providing feedback on whether it anticipated the correct answer or not. The algorithm will learn to estimate the exact nature of the link between instances and labels over time. When properly trained, the supervised learning algorithm will be able to recognise a new, previously unseen sample and predict a suitable label for it.

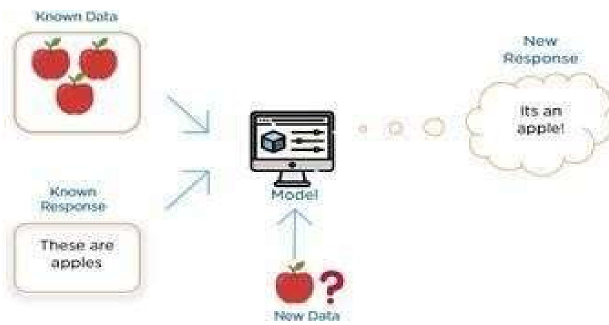


Fig 5-2 Supervised Learning

● **Unsupervised learning**: It is the polar opposite of supervised learning. There are no labels on it. Instead, the algorithm would be fed a large amount of data and given the skills to grasp the data's features. It can then learn to group, cluster, and arrange the data in such a manner that a human can enter and make sense of the newly structured data. Because unsupervised learning is dependent on data and its qualities, it may be described as data-driven. The data and how it is structured control the outcomes of an unsupervised learning activity.

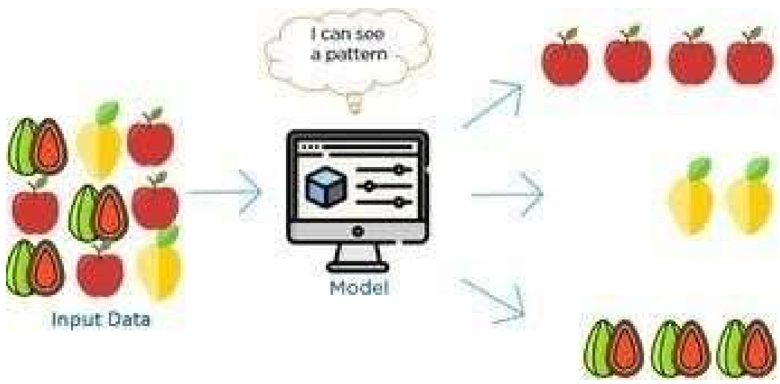


Fig 5-3 Unsupervised Learning

● **Reinforcement learning**: It differs significantly from supervised and unsupervised learning. Reinforcement learning is primarily motivated by behaviour. It draws inspiration from neuroscience and psychology. We require an agent and an environment, as well as a mechanism to connect the two via a feedback loop, for any reinforcement learning task. To link the agent to the environment, we provide it with a set of actions that influence the environment. To connect the environment to the agent, we make it provide two signals to the agent on a continuous basis: an updated status and a reward (our reinforcement signal for behavior).

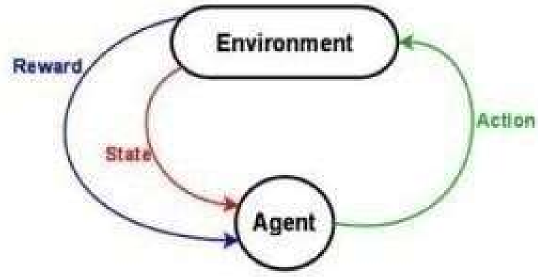


Fig 5-4 Reinforcement Learning

- **Neural Networks:** An artificial neural network is a network of linked nodes similar to the huge network of neurons in the human brain. Each circular node represents an artificial neuron, and each arrow indicates a link from one artificial neuron's output to another's input. Artificial neural networks (ANNs), also known as connectionist systems, are computing systems that are loosely based on the biological neural networks that comprise animal brains. Such systems "learn" to execute tasks by evaluating examples, typically without any task-specific rules being implemented. An ANN is a model that is built from a collection of linked units or nodes known as "artificial neurons," which are roughly modelled after the neurons in the human brain. Each link, like synapses in the actual brain, may send information, or a "signal," from one artificial neuron to another. When an artificial neuron gets a signal, it can process it and then signal other artificial neurons that are linked to it. The signal at a link between artificial neurons is a real number in most ANN implementations, and the output of each artificial neuron is generated by some non-linear function of the sum of its inputs. Edges are the connections between artificial neurons. The weight of artificial neurons and edges is often adjusted as learning progresses. The weight changes the intensity of the signal at a connection. Artificial neurons may have a threshold that is crossed by the aggregate signal before the signal is transmitted. Artificial neurons are often organised into layers. Different layers may apply various modifications to their inputs. Signals go from the first layer (the input layer) to the last layer (the output layer), sometimes many times. The ANN approach's original purpose was to solve problems in the same way as a human brain would. However, as time passed, emphasis shifted to certain activities, resulting in departures from biology. Artificial neural networks have been used to perform a wide range of tasks, including computer vision, voice recognition, machine translation, social network filtering, board and video game play, and medical diagnosis. Deep learning is made up of several hidden layers of an artificial neural network. This method attempts to simulate how the human brain converts light and sound into vision and hearing. Deep learning has been used successfully in a variety of applications, including computer vision, speech recognition, and picture categorization.

5.3 Algorithms Used

- Machine Learning provides a diverse set of algorithms from which to pick. These are commonly classified as classification, regression, grouping, and association. Classification and regression algorithms are examples of supervised learning, whereas clustering and association are examples of unsupervised learning.
- Classification: A classification problem is when the output variable is a category, such as

  —red or —blue or —disease and —no disease. Example: Decision Trees
- Regression: A regression problem is when the output variable is a real value, such as —dollars or —weight. Example: Linear Regression
- Clustering: A clustering problem is where you want to discover the inherent groupings in the data, such as grouping customers by purchasing behavior. Example: k means clustering
- Association: An association rule learning problem is where you want to discover rules that describe large portions of your data, such as people that buy X also tend to buy Y.

  Example: Apriori Algorithm
- The most appropriate type of algorithm is employed for each prediction based on the issue description and the intended result of the project. Many algorithms were investigated before selecting one and working with it further. The error rates and accuracy of each were evaluated.

5.3.1 Random Forest

Random forest is a versatile, user-friendly machine learning method that provides excellent results most of the time even without hyper-parameter adjustment. Because of its simplicity and diversity, it is also one of the most often used algorithms. It may be used for classification as well as regression applications. Random forest combines many decision trees to produce a more accurate and consistent forecast. Random forest has a significant benefit in that it can be used for both classification and regression tasks, which comprise the majority of contemporary machine learning systems. Another outstanding aspect of the random forest method is its ease of measuring the relative relevance of each variable on prediction. Sklearn has a wonderful tool for this that quantifies the relevance of a feature by examining how much the tree nodes that utilise that feature reduce impurity across all trees in the forest. After training, it computes this score automatically for each feature and adjusts the findings so that the total of all importance equals one.

In random forest, hyper parameters are used to either boost the prediction power of the model or to make the model quicker. Python has various random Forest methods using the following hyper arguments.

To increase the predictive power:

First, there is the n estimators hyper parameter, which is just the number of trees the algorithm generates before taking the maximum voting or prediction averages.

Another crucial hyper parameter is max features, which is the maximum amount of features considered by random forest while splitting a node.

The final critical hyper parameter is min sample leaf. This specifies how many leaves are necessary to separate an internal node.To Increase the model's speed:

The n jobs hyper parameter instructs the engine on how many processors it is permitted to use. It can only use one processor if it has a value of one. A value of -1 indicates that no restriction exists.

The random state hyper parameter allows the model's output to be replicated. When given the same hyper parameters and training data and a certain value of random state, the model will always deliver the same outputs.

The oob score (also known as oob sampling), a random forest cross-validation approach. Approximately one-third of the data in this sampling is not utilised to train the model but can be used to evaluate its performance. These are referred to as out-of-bag samples.

The steps for Random forest are:

Step 1 − First, start with the selection of random samples from a given dataset.

Step 2 − Next, this algorithm will construct a decision tree for every sample. Then it will get the prediction result from every decision tree.

Step 3 − In this step, voting will be performed for every predicted result.

Step 4 − At last, select the most voted prediction result as the final prediction result.

5.3.2 SVD:

In theory, the SVM algorithm, aka the support vector machine algorithm, is linear. What makes the SVM algorithm stand out compared to other algorithms is that it can deal with classification problems using an SVM classifier and regression problems using an SVM regressor. However, one must remember that the SVM classifier is the backbone of the support vector machine concept and, in general, is the aptest algorithm to solve classification problems.

Being a linear algorithm at its core can be imagined almost like a Linear or Logistic Regression. For example, an SVM classifier creates a line (plane or hyper-plane, depending upon the dimensionality of the data) in an N-dimensional space to classify data points that belong to two separate classes. It is also noteworthy that the original SVM classifier had this objective and was originally designed to solve binary classification problems, however unlike, say, linear regression that uses the concept of line of best fit, which is the predictive line that gives the minimum Sum of Squared Error (if using OLS Regression), or Logistic

Regression that uses Maximum Likelihood Estimation to find the best fitting sigmoid curve, Support Vector Machines uses the concept of Margins to come up with predictions.

Before understanding how the SVM algorithm works to solve classification and regression-based problems, it's important to appreciate the rich history. SVM was developed by Vladimir Vapnik in the 1970s. As the legend goes, it was developed as part of a bet where Vapnik envisaged that coming up with a decision boundary that tries to maximize the margin between the two classes will give great results and overcome the problem of overfitting. Everything changed, particularly in the '90s when the kernel method was introduced that made it possible to solve non-linear problems using SVM. This greatly affected the importance and development of neural networks for a while, as they were extremely complicated. At the same time, SVM was much simpler than them and still could solve non-linear classification problems with ease and better accuracy. In the present time, even with the advancement of Deep Learning and Neural Networks in general, the importance and reliance on SVM have not diminished, and it continues to enjoy praises and frequent use in numerous industries that involve machine learning in their functioning

5.3.3 Ada Boost

AdaBoost algorithm, short for Adaptive Boosting, is a Boosting technique used as an Ensemble Method in Machine Learning. It is called Adaptive Boosting as the weights are re-assigned to each instance, with higher weights assigned to incorrectly classified instances. Boosting is used to reduce bias as well as variance for supervised learning. It works on the principle of learners growing sequentially. Except for the first, each subsequent learner is grown from previously grown learners. In simple words, weak learners are converted into strong ones. The AdaBoost algorithm works on the same principle as boosting with a slight difference.

5.3.4 Steps:

Step 1 – Creating the First Base Learner

Step 2 – Calculating the Total Error (TE)

Step 3 – Calculating Performance of the Stum

Step 4 – Updating Weights

Step 5 – Creating a New Dataset

In Python, coding the AdaBoost algorithm takes only 3-4 lines and is easy. We must import the AdaBoost classifier from the **sci-kit learn library**. Before applying AdaBoost to any dataset, one should split the data into train and test. After splitting the data into train and test, the training data is ready to train the AdaBoost model. This data has both the input as well as output. After training the data, our algorithm will try to predict the result on the test data. Test data consists of only the inputs. The output of test data is not known by the model. Accuracy can be checked by comparing the actual output of the test data and the output predicted by the model. This can help us conclude how our model is performing and how much accuracy can be considered, depending on the problem statement. If it's a medical problem, then accuracy should be above 90%. Usually, 70% accuracy is considered good. Accuracy also depends on factors apart from the type of model.

Summary:

Summary Among the different networks, it is difficult to determine beforehand which of them can act as the best extractor of accuracy. So we perform the methodology on each of the architecture and determine which is the best of them all based on accuracy.

6 Implementation Details:

The software and tools used are described in 5.1. Different modules used in the project are described in 4.2

6.1 Software and Tools

The project operates on a laptop with a Ryzen Core 5400 Processor and Windows 10, 64bit installed. The machine comes with 8GB of RAM. Python 3.6 was used to create the complete project. Table 6-1 summarises the tools and modules used in Python for various tasks.

Task	Library / Framework	Version
Reading Data from Source File	Pandas	1.2.4
Finding Statistics about Crops	Ipywidgets	7.6.3
Scaling Operations	Scikit (Sklearn)	2.0
Reading Images	OpenCV	4.5.4
Neural Network Operations	Keras	2.6.0

Table 6-1 Python libraries and their version for the various tasks used in the project

- **Data Preparation**

Most machine learning algorithms require data to be formatted in a very specific way, so datasets generally require some amount of preparation before they can yield useful insights.Not much work is needed for our dataset as all points are useful for our model. No columns have been removed or rows as they were prepared from raw data by us .So no preprocessing is required and have been directly passed to the model. We have used different models such as SVD, RandomForest and AdaBoost. We have also used DLIB library to get to to know the different spatial points of the image. The data set we have used contains over 1000 images equally divide between male and female.

- **Model Selection**

We require two datasets when building a machine learning model: one for training and one for testing. But now we just have one. So, let's divide this in half with an 80:20 ratio. We will also split the data frame is divided into feature and label columns We used the sklearn train test split function here. Then, split the dataset with it. When test size = 0.2, the split is 80% train dataset and 20% test dataset. The random state option starts a random number generator, which aids in splitting the dataset. Four datasets are returned by the method. They were labelled as train x, train y, test x, and test y. We can observe the split of the dataset if we look at the form of the dataset. Four datasets are returned by the method. They were labelled as train x, train y, test x, and test y. We can observe the split of the dataset if we look at the form of the dataset. To fit several decision trees to the data, we utilised RandomForestRegressor, SVM, and AdaBoost. Finally, I train the model by using the fit function with train x and train y parameters.

We must test the model after it has been trained. To do this, we will provide test x to the predict function.Finding the BMI classified values:

7 Finding the BMI certified values

So firstly we had a dataset of 1000 plus images.
We had convert them into an excel file for our machine learning algorithm. We started with converting each of the images into each of the faces into facial points. Each image has been converted to 68 facial points .So together we have both 128 columns ,each point giving 2 columns,x axis and y axis respectively.
Each of this conversion is done using the D-Lib module.
After this we have to split the model in the ratio 4 :1. The models SVM, Random Forest and AdaBoost is used.
All the three models will be run on the the models and checked on accuracy.

The best accuracy model will further used, the accuracy can be next found in the results model.

8 INTERMEDIATE RESULTS AND DISCUSSION:

This chapter presents the intermediate results found for the procedures elaborated in the previous section, and also describes the datasets used for the project, and the evaluation metrics considered. The results presented here are intended only as a baseline, and will be further worked upon to improve them in the next phase of the project.

8.1 Datasets

There are 6 datasets based on BMI values.

They are normal, underweight, overweight, obese class 1, obese class 2 and obese class 3.

All these datasets together contain 1000+ images.

A converted csv file containing all the facial points.

8.2 Performance Metrics

There are three performance metrics used to evaluate the effectiveness of the system proposed in this thesis – accuracy of identification and loss value

8.3 Accuracy:

MODELS	TRAIN ACCURACY	TEST ACCURACY
Random Forest	0.966188	0.841823
SVM	0.9958100558659218	0.8273615635179153
AdaBoost	0.7860635696821516	0.7658536585365854

8.4 Results:

Promising results have been shown by the model compared to previous models.

So based on the results we can conclude that if you don't want any over fitting or underfitting u can go with Adaboost.If accuracy is the only measure then Random forest gives the highest accuracy.

9 CONCLUSIONS AND FUTURE WORK

So previously by using deep learning models we could get an highest accuracy of 70 percent which is not that great. So this work can be considered as a significant improvement in accuracy by using DLIB and getting the spatial points and converting the above model into a machine learning problem and dealing with it.

Future work might include improving the model even more or using another DLIB model with more facial points for each image.

10 REFERENCES:

[1].Dantcheva, P. Elia, and A. Ross, "What else does your biometric data reveal? a survey on soft biometrics," IEEE Transactions on Information Forensics and Security, pp. 1–26, 2015

[2].V. Coetzee, J. Chen, D. I. Perrett, and I. D. Stephen, "Deciphering faces: Quantifiable visual cues to weight," Perception, vol. 39, no. 1, pp. 51–61, 2010

[3].L. Wen and G.-D. Guo, “A computational approach to body mass index prediction from face images,” Image and Vision Computing, vol. 31, no. 5, pp. 392–400, 2013

[4].Dantcheva, P. Bilinski, F. Bremond, "Show me your face and I will tell you your height, weight and body mass index," Proc. of 24th IAPR International Conference on Pattern Recognition (ICPR),
(Beijing, China), August 2018.

[5]. Chen, A. Dantcheva, and A. Ross, “Impact of facial cosmetics on automatic gender and age estimation algorithms,” in Proceedings of IEEE International Conference on Computer Vision Theory and Applications (VISAPP), vol. 2, 2014, pp. 182–190.

[6] J. Bekios-Calfa, J. M. Buenaposada, and L. Baumela, “Robust gender recognition by exploiting facial attributes dependencies,” PRL, 2013.

[7] J. C. Klontz and A. K. Jain, “A case study on unconstrained facial recognition using the boston marathon bombings suspects,” Technical Report, no. MSU-CSE-13-4, 2013.

[8] A. Dantcheva, C. Velardo, A. D’angelo, and J.-L. Dugelay, “Bag of soft biometrics for person identification,” Multimedia Tools and Applications, vol. 51, no. 2, pp. 739–777, 2011.

[9] V. Coetzee, J. Chen, D. I. Perrett, and I. D. Stephen, “Deciphering faces: Quantifiable visual cues to weight,” Perception, vol. 39, no. 1, pp. 51–61, 2010.

[10] D. D. Pham, J.-H. Do, B. Ku, H. J. Lee, H. Kim, and J. Y. Kim, “Body mass index and facial cues in sasang typology for young and elderly persons,” Evidence-Based Complementary and Alternative Medicine, vol. 2011, 2011.

[11] D. Adjeroh, D. Cao, M. Piccirilli, and A. Ross, “Predictability and correlation in human metrology,” in Proc. of the IEEE WIFS, 2010.

[12] L. Wen and G.-D. Guo, “A computational approach to body mass index prediction from face images,” Image and Vision Computing, vol. 31, no. 5, pp. 392–400, 2013.

[13] E. A. Yates, A. K. Macpherson, and J. L. Kuk, “Secular trends in the diagnosis and treatment of obesity among us adults in the primary care setting,” Obesity, vol. 20, no. 9, p. 19091914, 2012. [14] C. Velardo and J.-L. Dugelay, “Weight estimation from visual body appearance,” in Proc. of IEEE BTAS, 2010.