HIGH LEVEL DESIGN (HLD)

Face image BMI prediction

Revision Number:
Last date of revision:

Document Version Control

Date Issued	Version	Description	Author
23-03-2024	1	Initial HLD-V1.0	Ranjith V

Contents

2.Abstract

The Face Image BMI Prediction project aims to leverage deep learning techniques to estimate Body Mass Index (BMI) from facial images. This interdisciplinary endeavor integrates computer vision and machine learning to provide a novel solution for non-invasive health monitoring.

The core of the project is a Convolutional Neural Network (CNN) model, which is trained on a diverse dataset of facial images and corresponding BMI values. The images undergo preprocessing steps, including resizing and normalization, to ensure consistency and accuracy in model predictions. The model architecture is designed to extract meaningful features from the images, translating these features into accurate BMI predictions through a series of convolutional and fully connected layers.

To facilitate easy access and practical application, a Flask-based web API is developed. This API allows users to upload facial images and receive real-time BMI predictions, making the model accessible and user-friendly. The API is designed to handle image input, process it through the trained model, and return the BMI result efficiently. For data storage and management, the system integrates with a Cassandra database, ensuring scalable and reliable handling of the image data and prediction results.

The project is deployed on an Ubuntu server, using Gunicorn as the WSGI server for handling web requests. This setup ensures robust performance and scalability for real-world applications.

Additionally, remote server management is facilitated using Putty, streamlining deployment and maintenance processes. The project not only demonstrates the potential of AI in health monitoring but also provides a scalable framework for further enhancements and integration into broader health diagnostic systems. By offering a non-invasive, easily accessible method for estimating BMI, this project contributes to the evolving landscape of health technology, promoting proactive health management and awareness.

3.Introduction

- This documentation serves as a comprehensive guide for the Face Image BMI Prediction project, detailing the development process, technical components, and implementation strategies.
- The aim of this documentation is to provide a clear and structured overview of the project, ensuring that both developers and stakeholders can understand the scope, methodologies, and technologies employed.

3.1 Purpose

- The primary purpose of this documentation is to outline the steps taken to create a machine learning model capable of predicting Body Mass Index (BMI) from facial images.
- This includes a detailed explanation of data collection, preprocessing, model development, and deployment. By documenting each phase of the project, we aim to ensure reproducibility, facilitate collaboration, and provide a foundation for future enhancements.

3.2 Objectives

- Clarity and Transparency: To offer a transparent view of the project's workflow, from initial data collection to final deployment, ensuring that each step is well-documented and easy to follow.
- Technical Reference: To serve as a technical reference for developers, providing detailed code snippets, configuration settings, and explanations of the tools and libraries used.
- Facilitate Collaboration: To provide a common understanding among team members and stakeholders, enabling effective collaboration and communication.
- Future Development: To create a foundation for future improvements and extensions of the project, allowing new contributors to quickly understand the existing work and build upon it.

By providing this documentation, we aim to ensure that the Face Image BMI Prediction project is not only successful in its current implementation but also accessible and understandable for future developments and enhancements.

4. General description

4.1 Product Perspective

- The Face Image BMI Prediction project aims to provide a non-invasive, rapid, and user-friendly solution for estimating Body Mass Index (BMI) using facial images. This approach leverages advancements in deep learning and computer vision to simplify the BMI calculation process, which traditionally requires manual input of height and weight.

4.2 Problem Statement

- Traditional methods of calculating BMI require physical measurements, which can be inconvenient, time-consuming, and prone to errors. There is a need for an alternative method that provides accurate BMI estimates without requiring manual measurements, making health monitoring more accessible and efficient.

4.3 Proposed Solution

- The Face Image BMI Prediction project proposes developing a Convolutional Neural Network (CNN) model to predict BMI from facial images. The process begins with collecting and preprocessing a dataset of facial images and corresponding BMI values. The CNN model is then trained to extract features from these images and accurately predict BMI.
- A Flask-based web API is developed to allow users to upload images and receive real-time BMI predictions. The system uses a Cassandra database for scalable data storage and management. Finally, the application is deployed on an Ubuntu server, ensuring robust performance and accessibility.

4.4 Further Improvements

- To enhance the Face Image BMI Prediction project, several key improvements can be made. First, the model accuracy can be increased by experimenting with various Convolutional Neural Network (CNN) architectures and expanding the dataset in terms of size and diversity to improve robustness and generalizability.

4.5 Data requirements

The Face Image BMI Prediction project requires a well-structured and comprehensive dataset to ensure the accuracy and reliability of the machine learning model. The following data requirements are essential for the successful development and deployment of the model:

1. Facial Images: High-quality images of human faces in various lighting conditions and angles.

2. BMI Values: Accurate and reliable BMI values corresponding to each facial image.

3.Demographic Information (Optional but beneficial): Age, gender, and ethnicity of individuals to improve the model's accuracy and robustness.

4.Data Annotation: Proper labeling and annotation of images with their corresponding BMI values and demographic information.

5. Diversity and Variability: A diverse dataset that includes individuals from different age groups, ethnicities, and body types.

6.Data Preprocessing: Images should be preprocessed to ensure uniformity (e.g., resizing, normalization).

7.Data Storage and Management: Efficient storage solutions to handle large volumes of image data, such as using a Cassandra database for scalability and reliability.

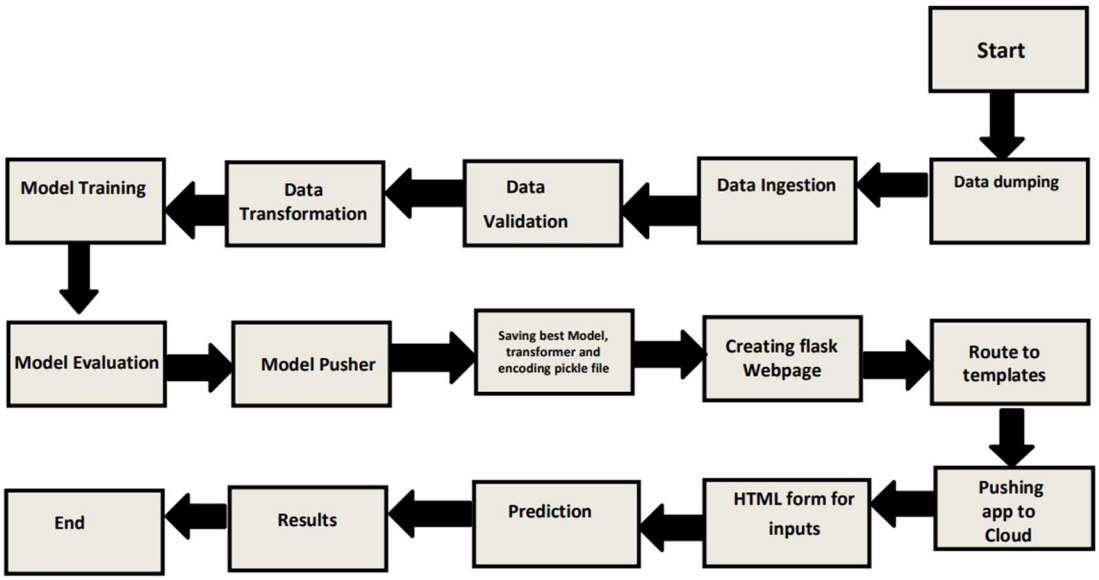
By meeting these data requirements, the Face Image BMI Prediction project can develop a robust and accurate model capable of predicting BMI from facial images, ultimately contributing to better health monitoring and management.

4.5 Tools used

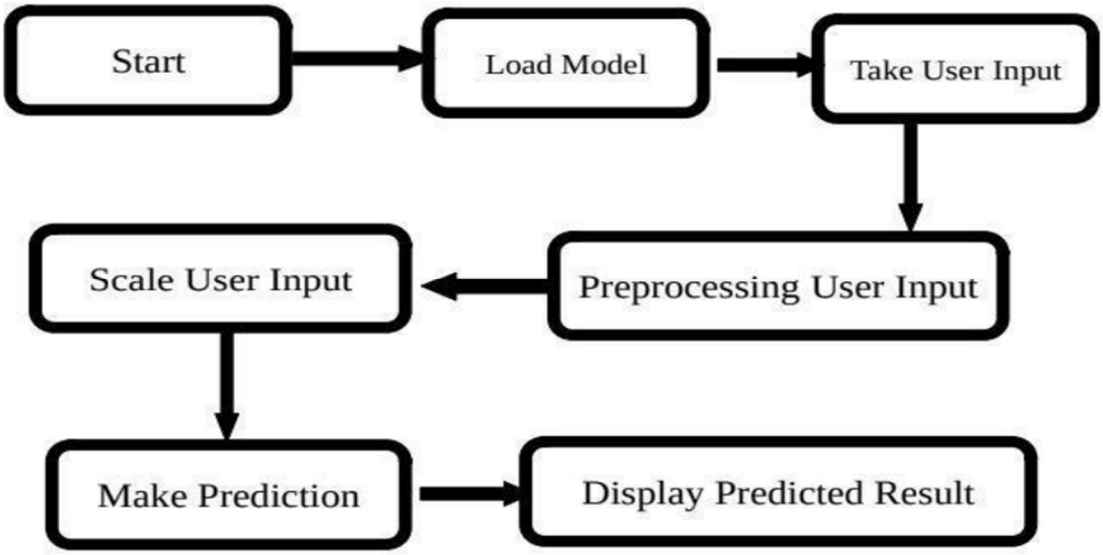
Python programming language and frameworks such as NumPy, Pandas, Scikit-learn, Matplotlib, Plotly, Flask etc. are used to build the whole model.

- PyCharm is used as IDE.
- Virtual Studio Code is also used as IDE.
- For visualization of the plots, Matplotlib, Seaborn and Plotly are used.
- Python, Flask is used for backend development.
- Azure is used for deployment of the model.
- GitHub is used as Version Control System.


5. Design Details



5.1 Deployment Process



5.2 Event log

The system should log every event so that the user will know what process is running internally.

Initial Step-By-Step Description:

1. The System identifies at what step logging is required.
2. The System should be able to log each system flow.
3. System should not hang even after handling so many users.

6. Performance

The machine learning based Thyroid Disease Detection solution will be used for detection of thyroid disease in patients having symptoms of thyroid. So that necessary action will be taken ASP. Also, model retraining is very important to improve performance.

6.1 Reusability

The code written and the components used should have the ability to be reused with no problems.

6.2 Application Compatibility

The different components for this project will be using python as an interface between them. Each component will have its own task to perform, and it is the job of the Python to ensure proper transfer of information.

6.3 Resource utilization

When any task is performed, it will likely use all the processing power available until that function is finished.

6.4 Deployment

7. Conclusion

- The code effectively addresses the task of predicting BMI from face images. It starts by loading and preprocessing image data and associated BMI values from a CSV file and a designated image directory. The preprocessing steps include resizing images to a standard size, normalizing pixel values, and converting images to arrays.
- A Convolutional Neural Network (CNN) is then built using Keras, with convolutional, pooling, and dense layers for feature extraction and BMI prediction. The model is compiled with the Adam optimizer and trained on the dataset. A Flask API is developed to serve the model, allowing for BMI predictions based on image inputs.
- Instructions for deploying the API on cloud platforms are provided, along with documentation requirements and performance optimization considerations. Additionally, logging is implemented for monitoring and debugging purposes.
- The final steps include maintaining code on GitHub, creating a project report, recording a demo video, and sharing the project on LinkedIn.

8. Reference

https://ieeexplore.ieee.org/document/9398733