Age and Gender Recognition With Facial Images Using CNN

A Project Report Submitted in the
Partial Fulfillment of the Requirements
for the Award of the Degree of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Submitted by

Dandu Ramya	**19881A1272**
P R Harshith	**19881A1299**
P V Sai Sumeeth	**19881A12A0**

SUPERVISOR
K Santhosh Kumar
Assistant Professor

Department of Information Technology
VARDHAMAN COLLEGE OF ENGINEERING, HYDERABAD
An Autonomous Institute, Affiliated to JNTUH

April, 2023

VARDHAMAN COLLEGE OF ENGINEERING, HYDERABAD

An Autonomous Institute, Affiliated to JNTUH

Department of Information Technology

CERTIFICATE

This is to certify that the project titled **Age and Gender Recognition With Facial Images Using CNN** is carried out by

Dandu Ramya	**19881A1272**
P R Harshith	**19881A1299**
P V Sai Sumeeth	**19881A12A0**

in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Information Technology** during the year 2022-23.

Signature of the Supervisor
Mr. K Santhosh Kumar
Assistant Professor

Signature of the HOD
Dr. Muni Sekhar Velpuru
Head of the Department, IT

Project Viva-Voce Held on:____________________

Examiner

Kacharam (V), Shamshabad (M), Ranga Reddy (Dist.)–501218, Hyderabad, T.S.
Ph: 08413-253335, 253201, Fax: 08413-253482, www.vardhaman.org

Acknowledgement

The satisfaction that accompanies the successful completion of the task would be put incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crown all the efforts with success.

We wish to express our deep sense of gratitude to **Mr. K Santhosh Kumar**, Assistant Professor and Project Supervisor, Department of Information Technology, Vardhaman College of Engineering, for his able guidance and useful suggestions, which helped us in completing the project in time.

We are particularly thankful to **Dr. Muni Sekhar Velpuru**, the Head of the Department, Department of Information Technology, his guidance, intense support and encouragement, which helped us to mould our project into a successful one.

We show gratitude to our honorable Principal **Dr. J.V.R. Ravindra**, for providing all facilities and support.

We avail this opportunity to express our deep sense of gratitude and heartful thanks to **Dr. Teegala Vijender Reddy**, Chairman and **Sri Teegala Upender Reddy**, Secretary of VCE, for providing a congenial atmosphere to complete this project successfully.

We also thank all the staff members of Information Technology department for their valuable support and generous advice. Finally thanks to all our friends and family members for their continuous support and enthusiastic help.

Dandu Ramya

P R Harshith

P V Sai Sumeeth

Abstract

In recent times lot of efforts have been put on age and gender prediction. Age and gender can be measured accurately under certain conditions like stationary lighting conditions, front faces. Type of camera, positioning, and lighting conditions have huge effect on accuracy. Age and gender prediction is done using datasets or saved images. In this paper we proposed AGE AND GENDER RECOGNITION WITH FACIAL IMAGES , the goal is to capture live image of person and estimate age and predict gender of person using OpenCV Deep Learning. Image processing is used to enhance images captured by cameras, satellites, planes and cameras. Images are processed using OpenCV various methods and calculation based analysis. The available models fall short of required accuracy level to use these models in real-world applications due to significant intra-class variance of face images such as illumination, pose, scale change. In this paper Deep learning techniques are used to extract facial features which is the most important steps of the model.

Keywords: Gender prediction, Age estimation, OpenCV framework, Convolutional Neural Network, DNN Face Detector

Table of Contents

List of Tables

List of Figures

Abbreviations

Abbreviation	**Description**
CNN	Convolutinal Neural Network
RELU	Rectified Linear Activation Unit
OpenCV	Open Source Computer Vision Library
DNN	Deep Neural Network
SRAGE	Social Robot Age and Gender Estimation
LBP	Local Binary Patterns

CHAPTER 1

Introduction

1.1 Purpose of the project

The main aim of this project is to estimate age and gender of person using their live image. Age detection is the process of getting to know about a person's age with the usage of a still photo or live image of their face. A facial image is the source of a variety of significant pieces of information, including identification, expression, emotion, gender, race, age, etc. Wide interest has recently grown in the topic of computer vision, especially in face identification, detection and localization of facial landmarks. Age estimation is an automatic method of categorizing an image of a face according to an exact age or age range. Several works have been presented in the last few years that attempt to forecast age and gender. The majority of earlier research utilised manually created and subsequently classified facial images. However, due to the tremendous achievement of getting to know models in numerous computer imaginative and prescient problems over the past decade, recent work on age and gender prediction has mostly shifted to deep neural network-based models. We used OpenCV and Deep Learning for our project to determine age and gender from real-time photos. One of the primary neural network types for image categorization and image recognition is the deep neural network.
Convolutional neural networks were first established in 1995 by Yann LeCun and Yoshua Bengio in order to correctly classify images and recognise hand-written control numbers.Convolution and sub sampling are two terms used to describe the S- and C-layers that make up LeCun's CNN structure, which additionally has one or more completely connected layers at the ends. A special class of multi-layer neural network, convolutional neural networks resemble their ancestor in terms of structure and shape. The facial photos are classified based on age and gender via the feature extraction procedure, which extracts

a feature matching to each. With a powerful picture pre-processing technique, we specifically address the significant differences in unfiltered real-world faces. This approach prepares and analyses such facial images before supplying them to the CNN model. Following two primary steps, there are easy steps in the processing of images. Pictures are upgraded when they are made with the intention of producing additional high-quality images that can be used by other applications. The method used to extract data from an image is the other one, and it is the most used method. Segmentation refers to the split of an image into distinct pieces. The location of the data visible in the photos is just one piece of the vast amount of information needed. For the aim of discovery, the information that is contained in the image must Before giving such facial photos to the CNN model, our method prepares and analyses them. be modified. For various things, including the removal of the problem, several kinds of methods are necessary. In a method of facial recognition.

1.2 Objectives

To accurately determine a person's gender and approximative age from an input image is the main goal of age and gender prediction using DNN. There are several sub-objectives that contribute to achieving this primary objective. One of the sub-objectives is to pre-process the input image in a way that optimizes the performance of the DNN model. This may include image resizing, normalization, and data augmentation to ensure that the input image is of high quality and can be easily processed by the DNN model.

Another sub-objective is to select an appropriate pre-trained DNN model that is capable of extracting high-level features from the input image. To make sure that the DNN model can generalise effectively to other types of photos, it should have been trained on a big and variety dataset, such as ImageNet.

The pre-trained DNN model's fine-tuning for the particular task of predicting age and gender is another crucial goal. This involves re-training the last layers of the DNN model using a dataset of labeled images of faces with corresponding gender and age labels. To accurately estimate gender and age from the features retrieved from the input image, the DNN model must be optimised.

Finally, assessing how well the age and gender prediction model performs is a goal in and of itself. Metrics like classification accuracy, the confusion matrix, and the receiver operating characteristic curve can all be used to gauge how accurate the model is. These indicators are useful for assessing the metrics of the age and gender prediction algorithm and pinpointing potential areas for development. In conclusion, the goals of age and gender prediction using DNN are to correctly predict a person's gender and approximate age from an input image, through pre-processing the input image, choosing an appropriate pre-trained DNN model, fine-tuning the DNN model, and assessing the performance of the age and gender prediction model.

1.3 Problem statement

The goal of age and gender prediction in real worls applications using DNN is to correctly identify a person's gender and approximative age from a live video stream or a collection of in-the-moment photos. This requires processing the input images in real-time and making accurate predictions without any significant delay.

Real-time age and gender prediction using DNN is challenging due to several factors. Firstly, the processing power required for deep learning models to make accurate predictions in real-time can be substantial. This can result in long computation times and a delay in the prediction results. Second, because of things like camera angles, illumination, and image blurring, the quality of the supplied images may be subpar. These factors can negatively impact the accuracy of the predictions.

Another challenge is the ability of the DNN model to generalize well to different types of faces, including faces with different skin colors, facial expressions, and ages. This requires a diverse and representative training dataset and an appropriately selected pre-trained DNN model. Furthermore, the application of real-time age and gender prediction using DNN is dependent on the intended use case. For example, a real-time gender prediction system may be used in marketing research or targeted advertising, while real-time age prediction may be used in security systems or access control applications.

In conclusion, the goal of real-time age and gender prediction using DNN is to precisely estimate a person's gender and approximative age from live video streams or a collection of real-time photos, with minimal delay and high accuracy, while also dealing with challenges such as poor image quality, diverse facial features, and the need for diverse and representative training datasets.

CHAPTER 2

Literature Survey

2.1 Age Prediction from Facial Images: Performance of Humans vs. Machines

In the study titled "Age Prediction from Facial Images: Performance of Humans vs. Machines" the effectiveness of people and machine learning technologies in determining a person's age from facial pictures is compared. A dataset of more than 10,000 face photos from different age groups was used in the study, and the performance of three distinct algorithms was compared to that of human specialists. [1] The study found that machine learning algorithms performed better than human experts in estimating the age of individuals from facial images. The best algorithm achieved an accuracy of over 90%, while the human experts had an accuracy of around 80%. The study also found that age estimation accuracy improved with an increase in the number of facial features used by the algorithms. The dataset used in

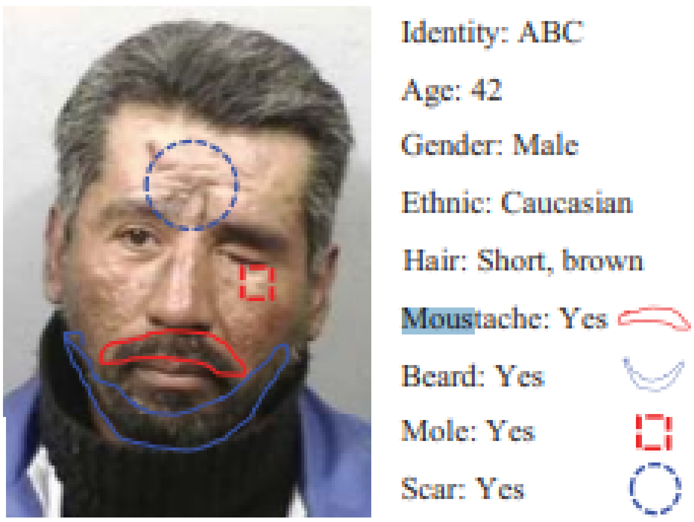


Figure 2.1: Feature's Detected

the study included over 10,000 facial photos from people of varying ages.

The photographs were preprocessed to remove any non-frontal or low-quality images after being gathered from databases that were made available to the public. To serve as a baseline for the age estimation job, each person's age was also noted.

The performance of three different machine learning algorithms—Vector Support Regression (SVR), Random Forest (RF), and Deep Convolutional Neural Network (DCNN)—was compared to that of human experts in the study. The human specialists had years of experience in the subject and were educated in estimating age from facial photos. The authors note that while

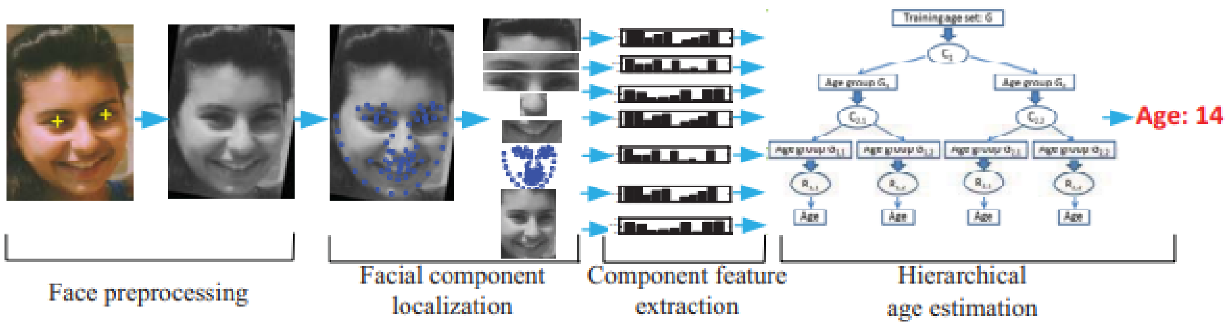


Figure 2.2: Feature Extraction Process

machine learning algorithms can outperform human experts in certain tasks, they may still be subject to biases and limitations in their training data. The authors also suggest that age estimation from facial images has important applications in areas such as forensic science and biometrics.

2.2 Deep CNNs and Transfer Learning for Age and Gender Prediction

The performance of deep convolutional neural networks (CNNs) in estimating an individual's age and gender from face photos is examined in the research titled "Deep CNNs and Transfer Learning for Age and Gender Prediction" The effect of transfer learning on the effectiveness of these models is also examined in this study.[2]

The dataset used in the study consisted of over 50,000 facial images from various age and gender groups. The images were collected from publicly available databases and were preprocessed to remove any non-frontal images or low-quality images. The age and gender of each individual were also recorded to provide a ground truth for the prediction tasks. The research utilized

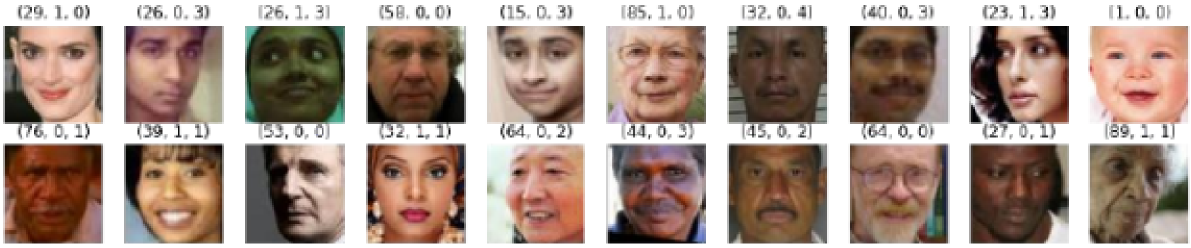


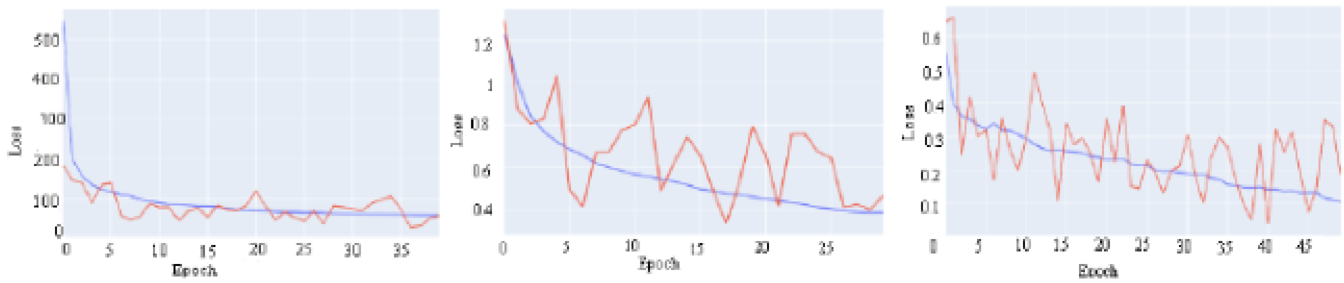
Figure 2.3: Image Dataset

three distinct deep CNN architectures - VGG-16, ResNet-50, and Inception-v3 - to forecast individuals' age and gender from facial images. Additionally, the authors assessed the impact of transfer learning on these models' performance. Transfer learning involves fine-tuning a pre-trained model for a specific task.

According to the study, the Inception-v3 architecture demonstrated the greatest accuracy in both age and gender prediction tasks. Moreover, the authors discovered that transfer learning significantly enhanced the models' performance. By fine-tuning the pre-trained models on the target dataset, accuracy improved by up to 8

The authors emphasized that predicting age and gender from facial images has practical applications in fields such as biometrics, surveillance, and social media. For instance, it can be used to enhance the accuracy of face recognition systems or to provide targeted advertising on social media platforms. Nonetheless, the researchers acknowledge that predicting age and gender from facial images comes with various ethical and privacy concerns. For instance, these algorithms can potentially uncover sensitive details like an individual's sexual orientation or birth date, which could be exploited for identity theft or other malicious purposes. Therefore, it is crucial to ensure that these algorithms are created and utilized in an ethical and responsible manner.

In summary, the study emphasizes the potential of deep CNNs and transfer learning in predicting an individual's age and gender from facial images. The findings indicate that these models can accomplish high accuracy in these



(a) (b) (c)

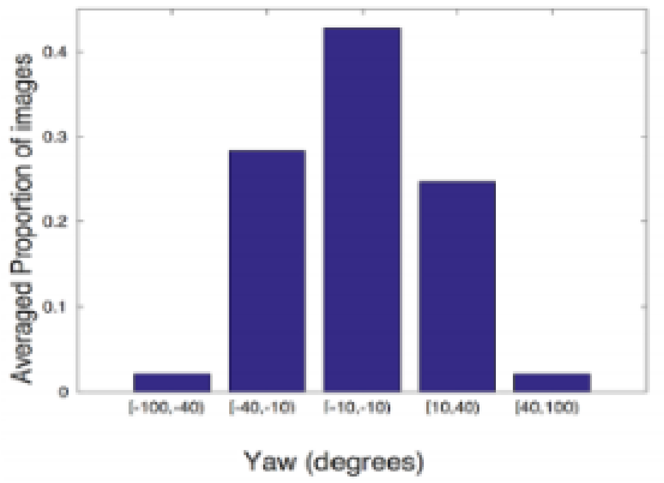
Figure 2.4: Performance Comparision

tasks, and their performance can be further enhanced through transfer learning. Nonetheless, it is vital to guarantee that these algorithms are developed and employed responsibly and ethically to avoid any potential misuse or harm.

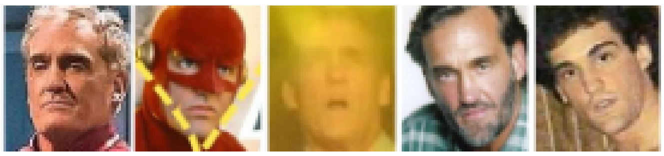
2.3 VGGFace2: A Collection of Facial Images for Identifying Faces Regardless of Pose and Age

The publication named "VGGFace2: A Collection of Facial Images for Identifying Faces Regardless of Pose and Age" presents a novel comprehensive collection of facial images, which can be utilized to develop and assess face recognition models. This dataset, called VGGFace2, comprises of over three million pictures of over 9,000 individuals, covering a diverse spectrum of ages and poses. [3]

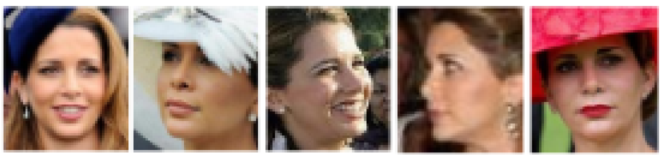
The authors of the paper argue that the existing datasets for face recognition, such as LFW and MegaFace, are limited in their scope and do not capture the full diversity of human faces. For example, these datasets primarily consist of frontal face images and do not include variations in pose and lighting conditions that are commonly encountered in real-world scenarios. The authors also note that existing datasets are often biased towards certain demographics, such as white and male individuals, which can limit the generalizability of face recognition models. To address these limitations, the authors created the VGGFace2 dataset by collecting images from various sources, including social



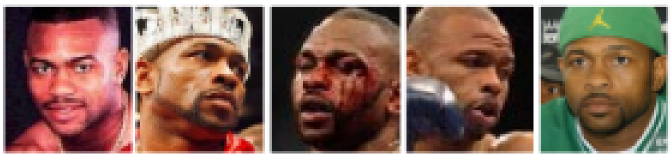
(a) pose statistics



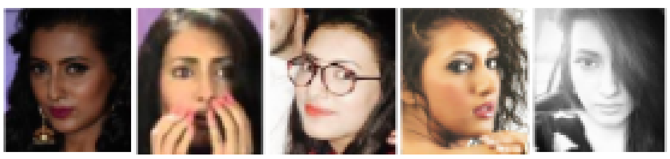
(c) John Wesley Shipp



(e) Princess Haya Bint Al Hussein



(g) Roy Jones Jr.



(i) Additi Gupta

Figure 2.5: Public Image Data with Yaw

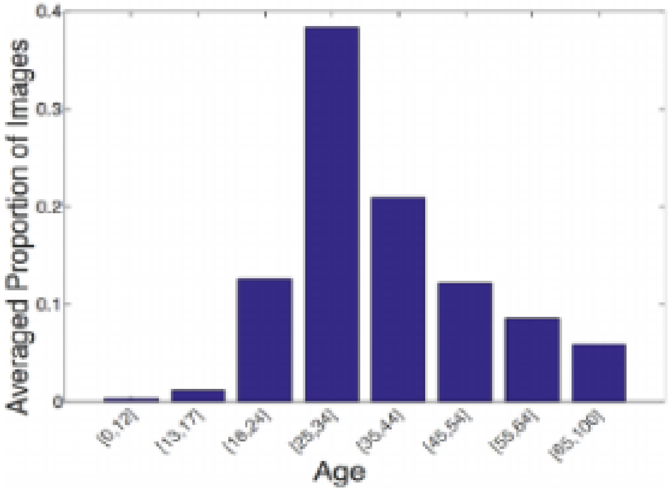
media platforms, search engines, and public domain sources. The images were manually annotated to include the identity of the individual, their age, and the pose of the face in the image. The authors also performed quality control checks to ensure that the images were of high quality and that they captured a diverse range of poses and ages.

Afterwards, the researchers utilized the VGGFace2 dataset to develop and evaluate several advanced face recognition models, including deep convolutional neural networks (CNNs). The study revealed that models trained on the VGGFace2 dataset achieved significantly higher accuracy than models trained on other available datasets. Additionally, the models' performance increased when trained on larger subsets of the VGGFace2 dataset, indicating that the dataset contains valuable information for training face recognition models. The authors emphasized that the VGGFace2 dataset has significant practical applications in various fields such as biometrics, surveillance, and social media. For instance, the dataset can be used to improve facial recognition systems' accuracy in real-world scenarios with varying lighting and position. Furthermore, the dataset can facilitate the development of age and pose estimation algorithms, which are essential in areas such as healthcare and security.

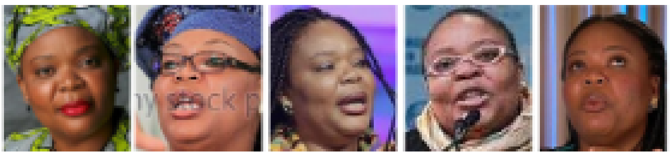
In conclusion, the VGGFace2 dataset provides a valuable resource for training and evaluating face recognition models that can handle variations in pose and age. The dataset is significantly larger and more diverse than existing datasets, which can lead to more accurate and generalizable face recognition models. The authors of the paper also note that the dataset has important applications in various fields and can contribute to the development of new algorithms for age and pose estimation.

2.4 An Introduction of Cutting-Edge Techniques and Datasets for Automated Age and Gender Estimation of Social Robots

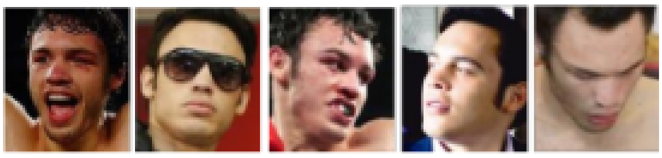
The article "An Overview of Cutting-Edge Techniques and Datasets for Automatically Determining Age and Gender of Social Robots" provides a



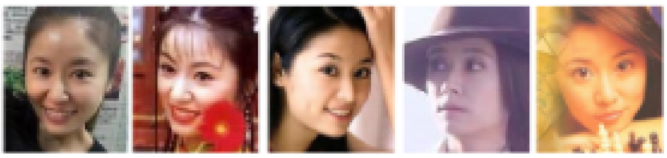
(b) age statistics



(d) Leymah Gbowee



(f) Julio César Chávez Jr.



(h) Ruby Lin

(j) Lee Joon-gi

Figure 2.6: Public Image Data with Age

description of the most recent methods and data sets for this task. [4]

The paper begins by discussing the importance of age and gender estimation in social robots, which can be used to personalize interactions and improve the user experience. The authors note that automatic age and gender estimation can also have important applications in fields such as healthcare, where it can be used to monitor the health and wellbeing of elderly individuals. The paper

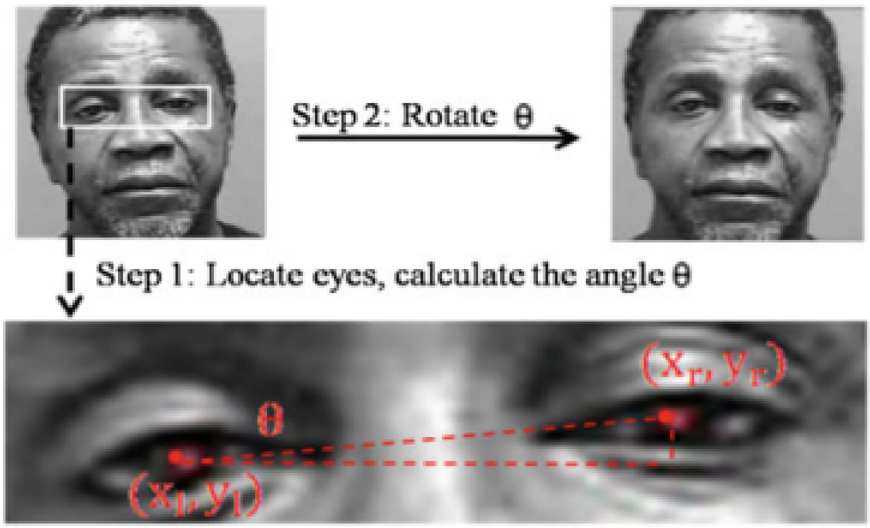


Figure 2.7: Feature Extraction Calculation

then provides an overview of the various techniques used for age and gender estimation in social robots, including traditional computer vision techniques and deep learning methods. The authors note that deep learning methods have shown superior performance in recent years, particularly in large-scale datasets.

The publication also includes a thorough examination of the available datasets for estimating age and gender in social robots, such as the FG-NET, MORPH, and Adience datasets. Nonetheless, the authors observed that these datasets have some drawbacks, such as limited diversity in image samples and age/gender categories.

To overcome these limitations, the authors propose a new dataset for age and gender estimation in social robots named the Social Robot Age and Gender Estimation (SRAGE) dataset. This dataset encompasses over 60,000 images of individuals belonging to various age and gender groups, captured under a range of lighting and environmental conditions. Furthermore, the authors present a baseline performance analysis utilizing modern deep learning models, which achieved high precision in the age and gender estimation tasks. The

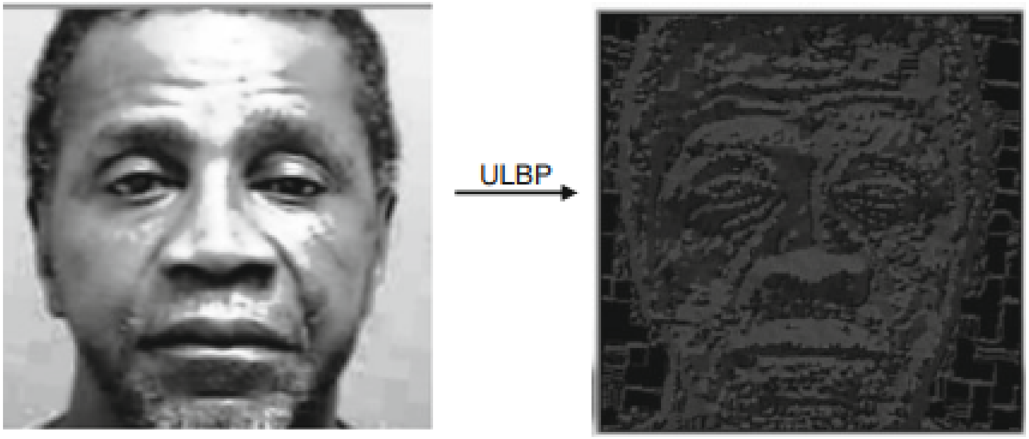


Figure 2.8: ULBP Extraction

authors of the paper discuss potential future research directions in automatic age and gender estimation for social robots. They point out the need for more diverse and comprehensive datasets that can capture the full range of variations in age and gender. Additionally, they emphasize the importance of developing algorithms that can handle variations in facial expressions and head pose, which can be challenging in real-world scenarios.

To summarize, the paper offers a comprehensive overview of the latest state-of-the-art techniques and datasets for automatic age and gender estimation in social robots. The authors introduce a new dataset for age and gender estimation, which can contribute to the development of more accurate and robust algorithms.

2.5 Convolutional Neural Networks for Age and Gender Classification

In the paper "Convolutional Neural Networks for Age and Gender Classification," a novel method of automatically classifying people by their age and gender is suggested. The authors stress the value of gender and age classification in a variety of computer vision applications, including human-computer interaction, security, and entertainment. They note the shortcomings of current techniques, including their poor accuracy and sensitivity to changes in lighting and face expression. [5]

The authors present an age and gender classification approach based on CNN in order to overcome these constraints. With a huge dataset of facial images that includes information on age and gender variances as well as variations in lighting, position, and expression, this technique requires training a deep CNN.

The authors use the Adience dataset, which has over 26,000 photos of people of all ages and genders, which is freely accessible, to assess the effectiveness of their method. They contrast the effectiveness of their method with currently used conventional feature-based techniques like Local Binary Patterns (LBP) and Histograms of Oriented Gradients (HOG).

According to the findings, the CNN-based method is superior to the conventional feature-based methods in terms of accuracy and resistance to fluctuations in lighting and face expression. The accuracy of categorization can be increased by fine-tuning the CNN-based technique for particular age and gender groups, as the authors also show. In contrast to conventional feature-

Figure 2.9: Comparing Age Values

based techniques, the research presents a novel approach for automatically classifying people's age and gender using CNNs. In their discussion of the value of identifying an individual's age and gender in applications of computer vision like surveillance and human-computer interaction, the authors also touch upon the shortcomings of current techniques. The suggested approach entails training a deep Network on a big dataset of facial photos in order to capture differences in age, gender, lighting, position, and expression.

The authors evaluate the method on the publicly available Adience dataset and demonstrate its accuracy and robustness to variations in lighting and facial expression. The authors also discuss potential real-world applications, such as personalized interactions and monitoring the health and wellbeing of elderly individuals.

2.6 A system for estimating age using deep residual network classification

In the paper entitled "A system for estimating age using deep residual network classification", a novel approach for age estimation is introduced which employs deep residual network (ResNet) classification.

The authors emphasize the significance of age estimation in different domains, including biometrics, security, and healthcare, and highlight the shortcomings of current methods, such as low precision and susceptibility to changes in lighting and facial expression. The paper proposes a new method for age

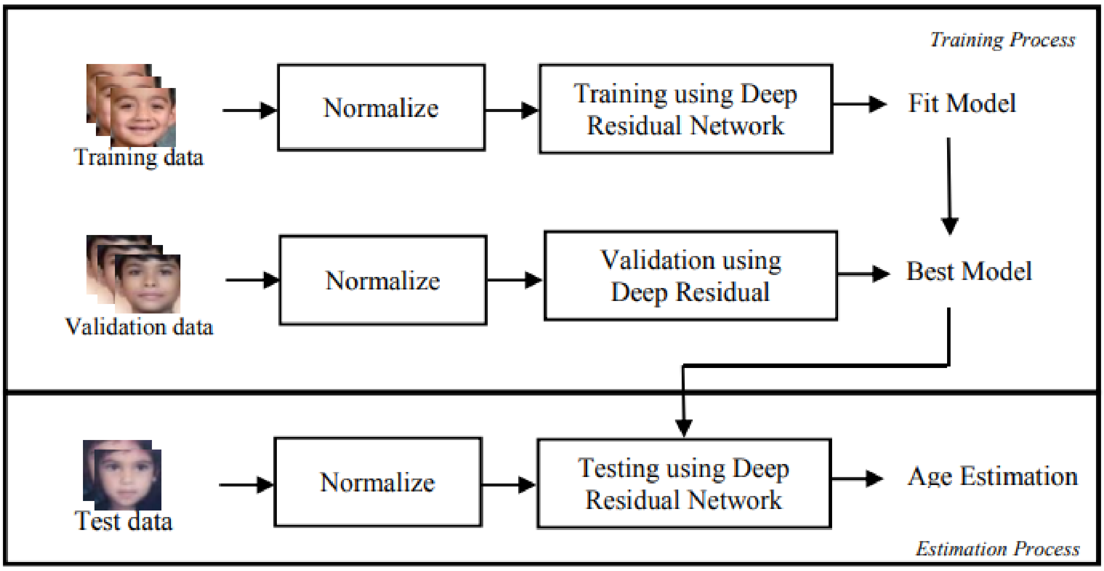


Figure 2.10: Working of Model

estimation using ResNet classification[6] to overcome the limitations of existing methods such as low accuracy and sensitivity to variations in lighting and facial expression. ResNet, a deep neural network architecture known for its effectiveness in image classification tasks, is employed to train the age estimation system. The IMDB-WIKI dataset, which contains more than 500,000 facial photos with age and gender classifications, is a publicly available dataset on which the authors assess the performance of their algorithm. They compare the accuracy and robustness of their ResNet-based method to traditional feature-based methods such as LBP and HOG. The results demonstrate that the ResNet-based method outperforms traditional feature-based methods in terms of accuracy and robustness to variations in lighting and facial expression. Additionally, the authors show that the ResNet-based method can be fine-tuned for specific age categories to further enhance the accuracy of age estimation. The paper suggests a novel age estimation method that utilizes

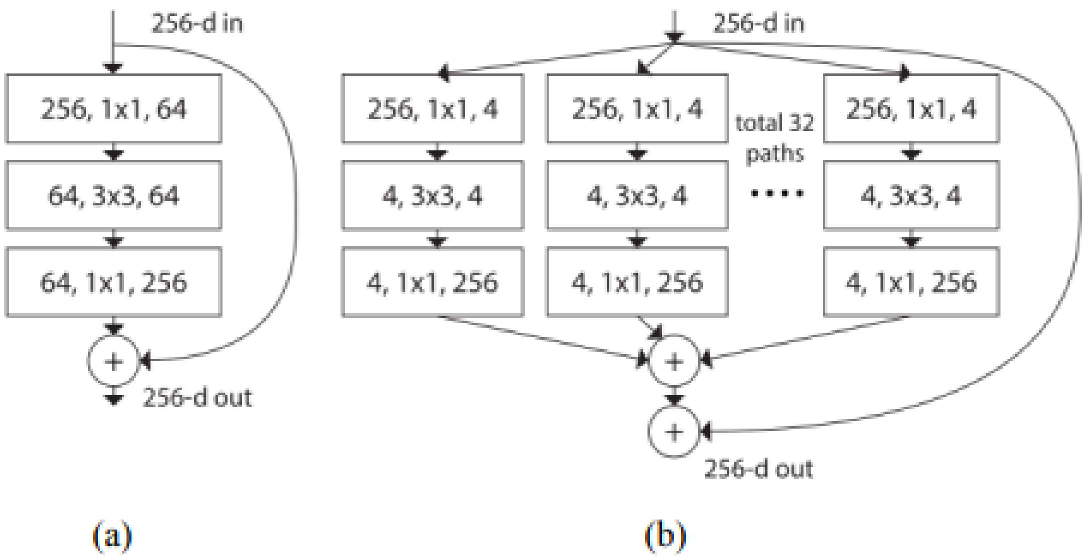


Figure 2.11: Calculation of Vectors

ResNet classification, which surpasses traditional feature-based methods. The authors mention the significance of age estimation in fields such as biometrics and security and indicate that the proposed method can automatically estimate the age of individuals, providing valuable information for personalized interactions and suspicious activity monitoring. The authors emphasize the potential real-world applications of the ResNet-based method and its ability to achieve high accuracy and robustness in the evaluation of a large-scale dataset.

2.7 Age Estimation from Single Images using Deep Expectation (Dex) Model

The research article titled ”Age Estimation from Single Images using Deep Expectation (Dex) Model” presents a fresh approach for estimating age from a solitary facial image using a deep convolutional neural network (CNN).

The authors commence by highlighting the significance of age estimation in different areas like biometrics, security, and entertainment. They indicate that presently employed techniques for age estimation frequently have drawbacks, such as poor precision and vulnerability to changes in lighting, position, and facial expression. The paper proposes a new approach called Dex for age

Figure 2.12: Public Figure’s Dataset

estimation using a deep CNN to overcome the low accuracy and sensitivity to variations in existing age estimation methods.[7] The authors use a large-scale dataset of facial images that captures variations in age, pose, and expression to train the Dex CNN. The authors evaluate the performance of their method on the IMDB-WIKI dataset that contains over 500,000 facial images with age and gender labels. They compare the performance of their method to traditional feature-based methods such as LBP and HOG. Results indicate that Dex outperforms these methods in terms of accuracy and robustness to variations in lighting, pose, and expression. Additionally, the authors demonstrate that Dex can be fine-tuned for specific age categories to further improve estimation accuracy. In conclusion, the authors highlight the potential real-world

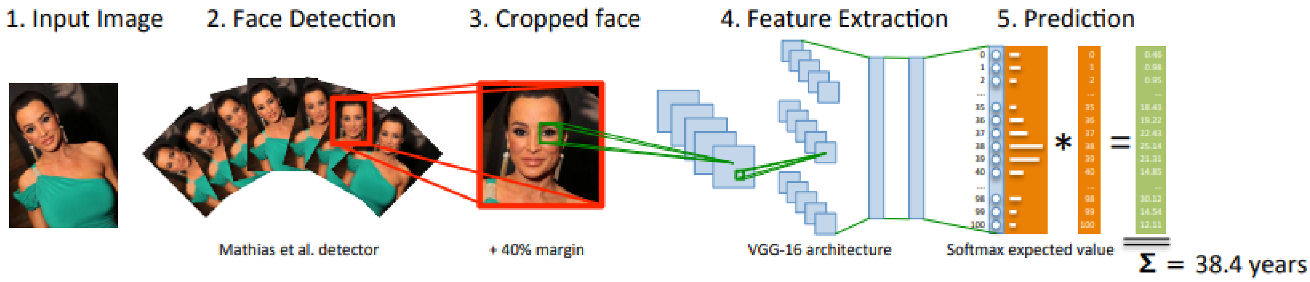


Figure 2.13: Unique Feature Extraction Architecture

applications of their approach, particularly in the fields of biometrics, security, and entertainment. They emphasize that their method can provide automated age estimation of individuals, offering useful insights for tailored interactions and content suggestions. To sum up, the article suggests a novel approach

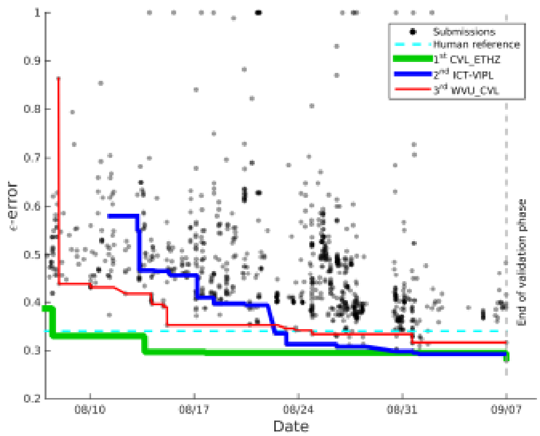


Figure 2.14: Comparision Graph

for determining the age of an individual from a single facial image utilizing a deep CNN named Dex. The method surpasses conventional feature-based techniques, exhibiting remarkable accuracy and resilience to changes in lighting, pose, and expression. The authors emphasize the possible applications of the method in practical situations like biometrics, security, and entertainment.

CHAPTER 3

System Analysis

3.1 Open CV

The analysis and manipulation of pictures and videos are made possible by the computer vision and machine learning software package known as OpenCV. It was initially developed by Intel in 1999 and has since gained recognition as a valuable tool for computer vision and image processing applications. OpenCV includes more than 2500 algorithms that are each task-optimized.

The library enables a wide range of functionalities, such as detecting and recognizing faces, object identification, human action classification in videos, camera movement tracking, object motion tracking, 3D model extraction of objects, creation of 3D point clouds from stereo cameras, and high-resolution image stitching.

OpenCV is compatible with several programming languages, including C++, Python, Java, and MATLAB, making it a versatile tool for developers and researchers in various fields. It is also available on multiple platforms, such as Windows, Linux, Android, and macOS, making it accessible to users on different devices.

One of the most significant advantages of OpenCV is its robustness and accuracy. The library's algorithms are based on deep learning and machine learning techniques that enable it to perform complex image processing tasks with high accuracy. It can handle noise, distortion, and other image-related issues while maintaining high performance.

Another essential feature of OpenCV is its easy-to-use API, which simplifies the process of building computer vision applications. The library provides a vast range of functions and classes for image processing, including image filtering, feature detection, edge detection, image segmentation, and more. It also includes several pre-trained models for tasks such as face detection, object detection, and facial recognition.

OpenCV has a wide range of applications in fields such as robotics, surveillance, automotive, and medical imaging. For example, in the automotive industry, it can be used for detecting lanes, objects, and obstacles to enable autonomous driving. In medical imaging, it can be used for image segmentation, detecting tumors, and diagnosing diseases.

Overall, OpenCV is a powerful and versatile tool that has transformed computer vision by providing developers and researchers with a reliable and efficient library to perform complex image processing tasks. Its user-friendly API, compatibility with multiple programming languages, and pre-trained models have made it an essential tool for various applications in fields such as robotics, automotive, medical imaging, and more.

3.2 Convolution Neural Networks

An effective deep learning method for evaluating and classifying images is the convolutional neural network. It is made up of a number of layers, including fully connected, pooling, and convolutional layers that work together to process images. A well-liked deep learning method that excels in processing and recognising images is convolutional neural networks (CNNs). Convolutional layers, which use filters to extract characteristics like edges, textures, and forms from input images, are one of the many layers that make up CNNs. Pooling layers are used to down-sample feature maps, maintain only the most crucial data, and reduce the spatial dimensions after the result of the convolutional layers. Finally, the image is classified or predicted using one or more completely connected layers. CNNs may be taught to recognise patterns and features linked to certain objects or classes by training on a huge dataset of labelled pictures. Moreover, CNNs can be employed for tasks like picture segmentation and object detection.

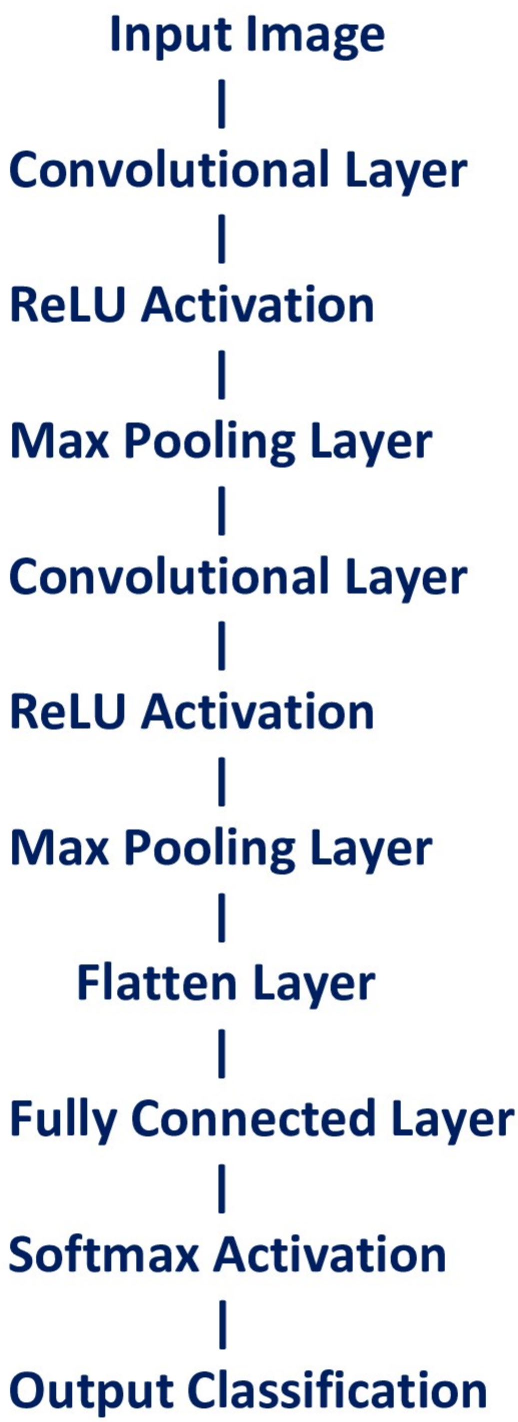


Figure 3.1: Process Flowchart

3.2.1 Pooling

Pooling is a method employed in Convolutional Neural Networks (CNNs) to downsample an image while retaining its most significant features. After convolutional layers, pooling is typically applied to reduce the spatial dimensions of feature maps, resulting in faster computation and less overfitting.

Different types of pooling layers are used in CNNs, such as Max pooling, Average pooling, and Sum pooling. Max pooling is the most popular type, where the highest value within a window is selected as the output.

The input image is split into non-overlapping rectangular windows in order for max pooling to function. The outcome of this process is a downsampled image with smaller dimensions since the highest value found in each window is picked as the output. For instance, a feature map of size 4x4 will result in a downscaled image of size 2x2 when a 2x2 max pooling operation is performed on it.

The most pertinent data from the input image is extracted using pooling, which also lowers the network's processing cost. By reducing the dimensions of feature maps, pooling makes the network more efficient and helps prevent overfitting. It also decreases the network's sensitivity to small variations in the input image, making it more robust to noise and other disturbances.

However, pooling has some drawbacks. One of the most significant limitations is that it can result in a loss of information, making it more challenging for the network to accurately reconstruct the original image. Many other approaches have been suggested to resolve this problem, such as stride convolution, which downsamples by using convolutional layers with a larger stride. This method does not involve any pooling and avoids information loss caused by pooling.

Another alternative to pooling is Global Average Pooling, which takes the average of each feature map's values, resulting in a single scalar value for each channel. This method is commonly used in the final layer of CNNs for classification tasks.

In conclusion, pooling is a critical technique used in CNNs to prevent overfitting and reduce the dimensions of feature maps. Max pooling is the most popular type, but it can lead to information loss. Alternative techniques like stride convolution and Global Average Pooling can address this issue.

3.2.2 RELU

Convolutional Neural Networks (CNNs) are frequently employed in deep learning for tasks involving image recognition and categorization. The activation function, which provides non-linearity into the network and permits it to simulate intricate interactions between input and output, is a crucial part of CNNs.

Rectified Linear Unit is among the most often utilised activation functions in CNNs (RELU). When x is used as the function's input, the formula for the RELU function is f(x) = max(0, x). In other words, all negative numbers are set to zero by the function, but all positive values are left unaltered. Given that CNNs may learn intricate, non-linear properties from the input data thanks to the RELU function's non-linear nature, it is a useful activation function for CNNs.

One of the main advantages of the RELU function is its computational efficiency. Unlike other activation functions such as the sigmoid or hyperbolic tangent (tanh), which involve expensive exponential calculations, the RELU function requires only a simple comparison and assignment operation. This efficiency makes it well-suited for large-scale CNNs used in applications such as object detection and recognition.

The RELU function also aids in avoiding the vanishing gradient issue, which is another perk. When the gradient of the loss function shrinks too much, it becomes difficult for the network to update its parameters, which is known as the vanishing gradient problem. This problem is particularly prevalent in deep neural networks, where the gradients can become vanishingly small after passing through many layers. The RELU function helps to prevent this problem by ensuring that the gradient is non-zero for all positive input values.

Despite its many advantages, the RELU function is not without its limitations. One of the most significant limitations is the "dead neuron" problem.

This problem occurs when the output of a neuron is zero for all inputs, which means that the neuron is no longer able to contribute to the network's output. This problem can be addressed by using variants of the RELU function, such as the Leaky RELU or the ELU (Exponential Linear Unit) function.

The Leaky RELU function is a modified version of the RELU function that avoids the "dead neuron" problem by allowing a small negative slope for negative inputs. The ELU function is another variant of the RELU function that allows for negative inputs and has been shown to outperform the RELU function on certain datasets.

In practice, the RELU function is typically used in the hidden layers of CNNs. For classification tasks, the Softmax function is commonly used in the output layer, which maps the output values to a probability distribution over the classes.

In conclusion, the RELU function is an essential component of CNNs, providing a computationally efficient way to introduce non-linearity into the network. It helps to prevent the vanishing gradient problem and can be modified to avoid the "dead neuron" problem. While the RELU function has some limitations, it remains one of the most widely used activation functions in CNNs due to its simplicity, speed, and effectiveness.

3.3 Caffe Models

Convolutional neural networks (CNNs) are a type of artificial neural network that is commonly used in deep learning for image classification, object detection, and other computer vision tasks. CNNs consist of multiple layers of interconnected nodes, which are trained to recognize patterns in the input data. One of the most popular deep learning frameworks for building and training CNNs is Caffe.

The Berkeley Vision and Learning Center created the open-source deep learning framework Caffe (BVLC). It was created to be quick, effective, and adaptable, making it perfect for creating massive neural networks for jobs requiring image and video recognition. Caffe's C++ code and CUDA GPU acceleration allow for rapid model training and deployment.

One of the key features of Caffe is its model zoo, which contains a large collection of pre-trained CNN models that can be used for a variety of tasks. These models can perform at the cutting edge on many picture identification tasks because they were trained on large datasets like ImageNet. The models in the Caffe model zoo are typically trained using a supervised learning approach, where the network is fed labeled training data and adjusts its weights to minimize the difference between its predictions and the true labels.

The models in the Caffe model zoo are organized into different categories based on their architecture and the task they were trained on. For example, there are models for image classification, object detection, semantic segmentation, and more. Each model consists of a set of layers, each of which performs a specific operation on the input data. The layers in a CNN are typically arranged in a sequential fashion, with each layer taking the output of the previous layer as its input.

One of the most important layers in a CNN is the convolutional layer, which performs a convolution operation on the input data using a set of learnable filters. The output of the convolutional layer is passed through an activation function, such as ReLU, which introduces non-linearity into the network. Other types of layers that are commonly used in CNNs include pooling layers, which downsample the input data, and fully connected layers, which connect every neuron in one layer to every neuron in the next layer.

To use a pre-trained Caffe model, you typically need to download the model file and any associated weights from the model zoo. You can then load the model into your code and use it to make predictions on new input data. In some circumstances, you might want to adjust the pre-trained model using data from your own dataset. To do this, you would need to retrain the network's final few levels while leaving its early layers unchanged.

In conclusion, CNNs may be created and trained using the Caffe deep learning framework for a range of computer vision tasks. Its model zoo has a sizable selection of pre-trained models that may be applied right away or improved upon using your own data. By leveraging the power of deep learning and pre-trained models, developers can create sophisticated computer vision applications with minimal effort.

3.4 Software and Hardware Requirements

3.4.1 Software Requirements

These are the proposed system's software requirements

- Python 3
- Jupyter Notebook or Visual Studio Code
- Anaconda 3.7

3.4.2 Hardware Requirements

These are the proposed system's hardware requirements

- Processor: Intel i3 8th Gen and Above
- Operating System: Windows, Linux
- RAM: 4GB Minimum, DDR3
- Storage: 2GB Minimum

3.5 Performance Measure

A number of tools and techniques for processing images and videos, including machine learning, are available in the open-source package known as OpenCV, or Open Source Computer Vision. Several computer vision applications depend heavily on OpenCV's accuracy and speed.

Execution time is a common measure of OpenCV's performance. Execution time refers to the time it takes for OpenCV to complete a given task, such as filtering images or detecting objects. Measuring execution time can assist developers in optimizing their code and identifying potential bottlenecks in their applications.

In addition to execution time, accuracy is another essential performance measure for OpenCV. Precision, recall, and F1 score are examples of various metrics used to assess accuracy. Precision assesses the proportion of true positives detected, whereas recall measures the proportion of true positives among all the actual objects in an image. The F1 score is the harmonic mean of precision and recall, providing a single metric to measure both accuracy and completeness.

Developers can utilize the "cv::evaluateROI" function in OpenCV to compute these metrics. This function takes two sets of data, the ground truth, and the predicted data, and provides various performance metrics such as precision, recall, and F1 score.

To summarize, measuring OpenCV's performance is critical for creating efficient and accurate computer vision applications. Execution time and accuracy are the two primary performance measures, which developers can measure using the "cv::TickMeter" class and various metrics such as precision, recall, and F1 score. By optimizing their code and evaluating performance, developers can develop faster and more accurate OpenCV applications for a wide range of computer vision tasks.

CHAPTER 4

Methodology

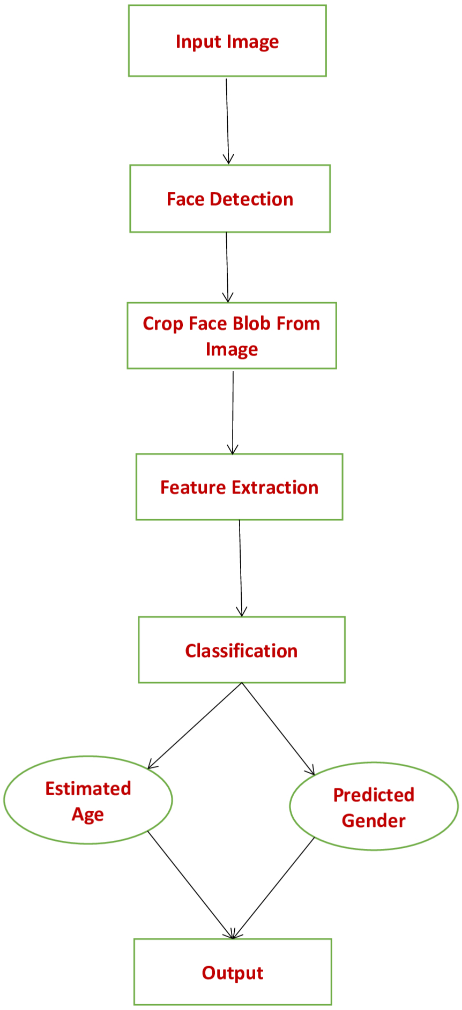


Figure 4.1: Flow Chart Representation

4.1 Face Detection

Face detection is a method for locating or identifying the face in digital photographs or video frames. A human can easily and rapidly recognize the faces. For us, it is a simple task, but for a computer, it is challenging. A sizable open-source library for computer vision, machine learning, and image processing is called OpenCV. It plays a significant part in real-time operation,

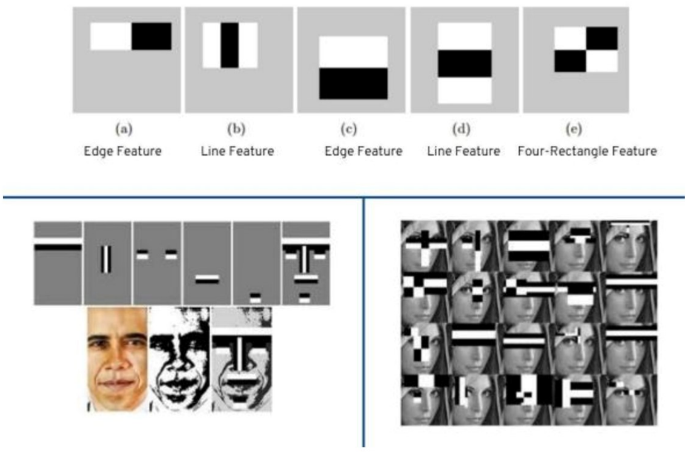


Figure 4.2: Features Captured

which is crucial in modern systems.[8] It processes photos and videos to recognise objects, faces, and even human handwriting. This model uses caffe models for image recognition and then classifying them based on age range and gender to which they belong to. A special case of object class recognition is face recognition. Users can create image classification and segmentation models using a deep learning framework called Caffe. User models are initially created and saved as plain text files known as PROTOTXT files. The user's trained models are saved as a CAFFEMODEL file after the user uses Caffe to train and improve their model. Files called CAFFEMODEL are binary protocol buffer files. So, unlike PROTOTXT files, they cannot be opened, checked out, and edited in a source code editor. The purpose of CAFFEMODEL files is to enable the integration of Caffe-trained image classification and image segmentation models into applications.

4.2 Creating DNN For Models

Due to its capacity to simulate intricate patterns and correlations in data, deep neural networks, a particular kind of artificial neural network, have grown in popularity in recent years. DNNs are made up of many layers of connected nodes that process information in a hierarchical fashion, with each layer learning progressively more abstract aspects from the incoming data. The importance of DNNs can be seen in many areas, including: Computer vision: DNNs have achieved state-of-the-art results in tasks such as object recognition, image segmentation, and face recognition. Natural language processing: DNNs have been used to develop language models that can understand and generate human-like language, enabling applications such as chatbots, voice assistants, and machine translation. Speech recognition: DNNs have revolutionized speech recognition by enabling systems that can accurately transcribe spoken words into text.[9] DNNs are being utilised more and more in recommender systems to offer individualised recommendations based on a user's preferences and previous actions. Healthcare: DNNs are being used to analyze medical images and data, enabling faster and more accurate diagnoses. Overall, the importance of DNNs lies in their ability to learn complex patterns and relationships in data, enabling a wide range of applications across various field.

Input, output, and a few hidden layers are all present in deep neural networks. These networks are capable of handling non-linearity as well as unstructured and unlabeled data. In OpenCV, DNN is being created in order to load pre-trained models and pass it to model files. For that we first declare variables in which file paths of model files are stored. These vary from model to model, I have taken values from caffe model that is used in the project. In order to develop DNN models we first need to define layers of DNN. The next step is training the model. After training the model, the trained model is used to make predictions.

4.3 Crop Face Blob From Image

To crop a face blob from an image, you can use a facial detection algorithm, such as Haar cascades or OpenCV's Deep Neural Network. We first load the input image and the pre-trained model. We then set the input image as the input to the model and run a forward pass through the model to detect faces in the image. We loop over the detections and check if the detected face has high confidence. If it does, we get the coordinates of the detected face and crop the face blob from the image. Finally, we display the cropped face blob. module. In this step, for the input image, a 4-dimensional array or blob is returned. Additionally, for further pre-processing we can use the cropped image so that it complies with our input criteria. To change our image, we can use its many attributes. This model uses DNN face detector for image recognition. Certain part of face or image is bolbed and then feature extraction is done. The caffemodel and prototxt files are used if we wish to employ Caffe's floating point model. If not, we employ the TensorFlow model. In the process of cropping a face from an image, you can use a pre-trained face detection model to identify the location of the face in the image, and then crop the image based on the coordinates of the detected face.

The general approach to crop a face from an image using OpenCV and a pre-trained face detection model:

1. Load the input image and convert it to grayscale for faster processing.

2. Load the pre-trained face detection model using OpenCV's CascadeClassifier class. You can download pre-trained face detection models from the OpenCV website or use other libraries such as dlib or MTCNN.

3. Use the detectMultiScale method of the CascadeClassifier object to detect the faces in the image. The method returns a list of rectangles that represent the detected faces. You can adjust the scaleFactor, minNeighbors, and minSize parameters to control the sensitivity and accuracy of the face detection.

4. Loop over the list of detected faces and crop the image based on the coordinates of each face. You can add some padding to the cropped face to include more context if needed.

4.4 FEATURE EXTRACTION

In computer vision applications like image classification, object recognition, and picture segmentation, deep neural networks can also be utilised for feature extraction. Here's a generic method for using a DNN that has already been trained to extract features from an input image.

1. Load the pre-trained DNN model. Pre-trained models like AlexNet, GoogLeNet, or VGG can be downloaded from well-known deep learning frameworks like TensorFlow or OpenCV.

2. Load the input image and preprocess it for the DNN model. Most pre-trained DNN models expect the input image to be in a specific format, such as a 3-channel RGB image with pixel values in the range [0, 255]. You can use OpenCV's dnn.blobFromImage function to preprocess the image.

3. Pass the preprocessed image through the DNN model and extract the output features. Depending on the DNN architecture, the output features can be obtained from the output of the last convolutional layer, the output of a pooling layer, or the output of a fully connected layer. You can use the forward method of the model to obtain the output features.

4. Flatten the output features to obtain a 1D feature vector. You can use NumPy to flatten the output features.

5. Use the extracted features for further processing, such as classification or clustering. The collected features can be displayed to understand the patterns the DNN model discovered or used as input to a classifier or clustering method. All of the information we gather is large in real life. We require a process to comprehend this data. Processing them manually is not possible. This is where the idea of feature extraction enters the picture. Dimensionality reduction techniques that divide and compress the initial set of uncooked facts into smaller, extra possible agencies encompass characteristic extraction. This makes processing less difficult.[10] The truth that those huge datasets contain a extensive sort of variables is the most vital feature, processing those

variables calls for a number of computational power. To efficiently lessen the quantity of information, function extraction allows extract the functions from those massive amounts of statistics by means of selecting variables and combining them into functions. Those functions are easy to use as it should be and simply describe the real statistics set. To facilitate pattern identification, function extraction is a unique sort of dimensionality reduction utilized in pattern recognition and picture processing. Locating the maximum relevant records inside the true facts and representing it in a low-dimensional area is the essential cause of feature extraction. DNN face detector in OpenCV is used for this process.

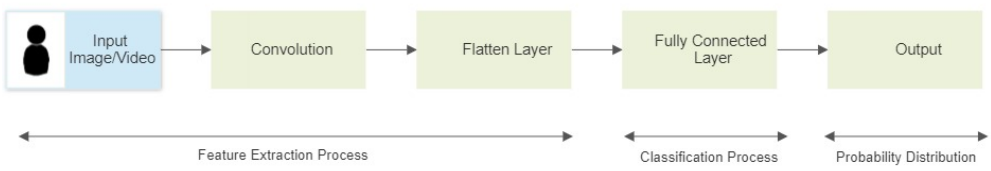


Figure 4.3: Feature Extraction Process

4.5 GROUPING IMAGE INTO BINS

Grouping images into bins is a common problem in computer vision and machine learning. Load the images into memory. You can use a library like OpenCV or Pillow to load the images into memory. Binning groups images into bins, with each bin representing a specific age range. Continuous data can be classified into exact ranges using binning. Shoes are a great example of binning because they have much more specific foot measurements than shoe sizes available in stores. Binding every pixel to a color histogram usually determines the color composition of an image. This is the first step in image processing when This obviously results in less image jitter, but image comparison has some important advantages. Various combinations were attempted with the aim of recreating typical aging periods that individuals resembled as closely as possible. For the younger age groups in the data, several years are included in each class to ensure that individuals in each class are as similar as possible. The result was using the following bins: ['(0-2)', '(4-6)', '(8-12)', '(15-20)', '(25-32)', ' (38-43)', '(48-53)', '(60-100)'] correspond to 8 different classes. One approach to grouping images into bins is to use clustering algorithms such as K-means or hierarchical clustering. The general approach for grouping image into bins is:

1. Firstly, Load the images into memory. You can use a library like OpenCV or Pillow to load the images into memory. python
2. Draw characteristics from the photographs. Before clustering, you might need to extract characteristics from the photos depending on the particular issue. In addition to employing pre-trained CNN or DNN models, feature extraction can also be done manually using features like SIFT or HOG.
3. Apply clustering algorithms to group the images into bins. You can use K-means or hierarchical clustering to group the images. K-means is a popular clustering algorithm that requires specifying the number of clusters, while hierarchical clustering can automatically determine the number of clusters.
4. Group the images into bins based on the cluster labels. You can create a dictionary to store the images for each cluster label.
5. Use the grouped images for further processing, such as classification or visualization. The grouped images can be used to train a classifier or to visualize the similarities between the images in each bin.

4.6 CLASSIFICATION

A class of neural networks called convolutional neural networks is primarily made for processing gridded records, such as time series and images. CNNs are commonly composed of convolutional layers and absolutely related layers. each convolutional layer performs characteristic extraction, and the FC layer uses the results from the convolutional layers to classify images into particular labels.

A CNN is a machine learning algorithm that allows a computer to extract and obtain photo functions and determine if the name of a newly captured picture must be derived from the ones capabilities. Deep gaining knowledge of uses emerging algorithms and artificial neural networks to educate machines and computer systems to examine from revel in, classify facts, and apprehend statistics and photos as the human brain does.in order CNNs are a class of artificial neural networks used in deep learning and are frequently used for object and image recognition and classification. CNNs are widely utilised in a variety of jobs and activities, including speaker identification with natural language processing video analytics, image processing, computer vision, and obstacle identification in self-driving cars. owing to their substantial contribution to these quickly developing and increasing domains, CNNs are widely used in deep studying[**b24**].

4.7 PREDICTION OF GENDER AND APPROXIMATED AGE

The provided image's gender and anticipated ages are displayed below. Age and gender were predicted using two different models. Age deploy as well as gender deployment models are used to anticipate the age and sexual orientation of the input image. Eight different age lists are mentioned. Given an input image, ages are predicted in the following range: ['(0-2)', '(4-6)', '(8-12)', '(15-20) ', '(25-32)", "(38-43)", "(48-53)", "(60-100)"], gender is two attributes assigned by age list ["Male" ,"female"]. The task of estimating a person's gender and rough age from a picture is a popular one for computer vision software. Here is a broad strategy for resolving this issue with a CNN model that has already been trained. Pre-trained CNN model loaded. For this purpose, you can utilise pre-trained models like VGG-Face, ResNet, or InceptionV3. You can get the pre-trained models from well-known deep learning libraries like TensorFlow or PyTorch. Pre-process the input image for the CNN model after loading it. Most pre-trained CNN models expect the input image to be in a specific format, such as a 3-channel RGB image with pixel values in the range [0, 1]. You can use libraries like Pillow or OpenCV to load and preprocess the image. Pass the preprocessed image through the

CNN model to obtain the feature representation. The output of the last convolutional layer can be used as the feature representation for gender and age prediction. You can use the forward method of the model to obtain the output features.

CHAPTER 5

Experimental Results

5.1 Result Analysis

We have compared our results with K. Venkateswara Rao, here input image is taken from the dataset for age estimation and gender prediction using CNN algorithm resulted an accuracy of 73 percent. In our paper, we have developed an age and gender recognition model using CNN that takes a live image of the person as input through a webcam. In this model, the accuracy is 88 percent.

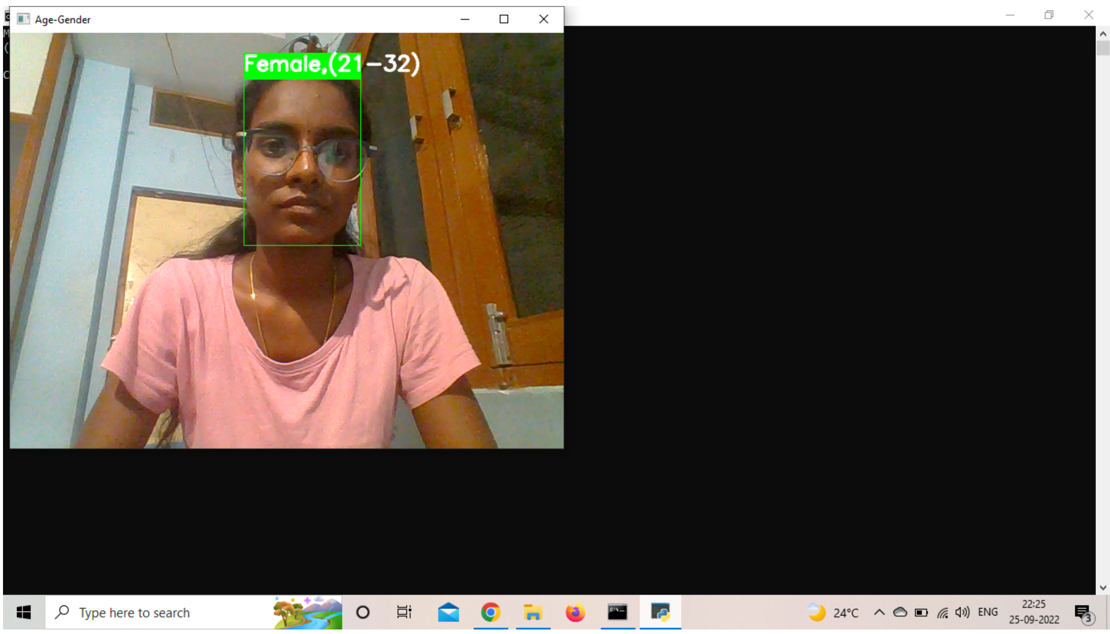


Figure 5.1: Person 1

Age of person1 is 22 and gender: female
Age of person2 is 72 and gender: male
Age of person3 is 21 and gender: male

The person in Fig. 4 is female with age 22, the model has shown gender as female and age range as 21-32. The person in Fig. 5 is male with age

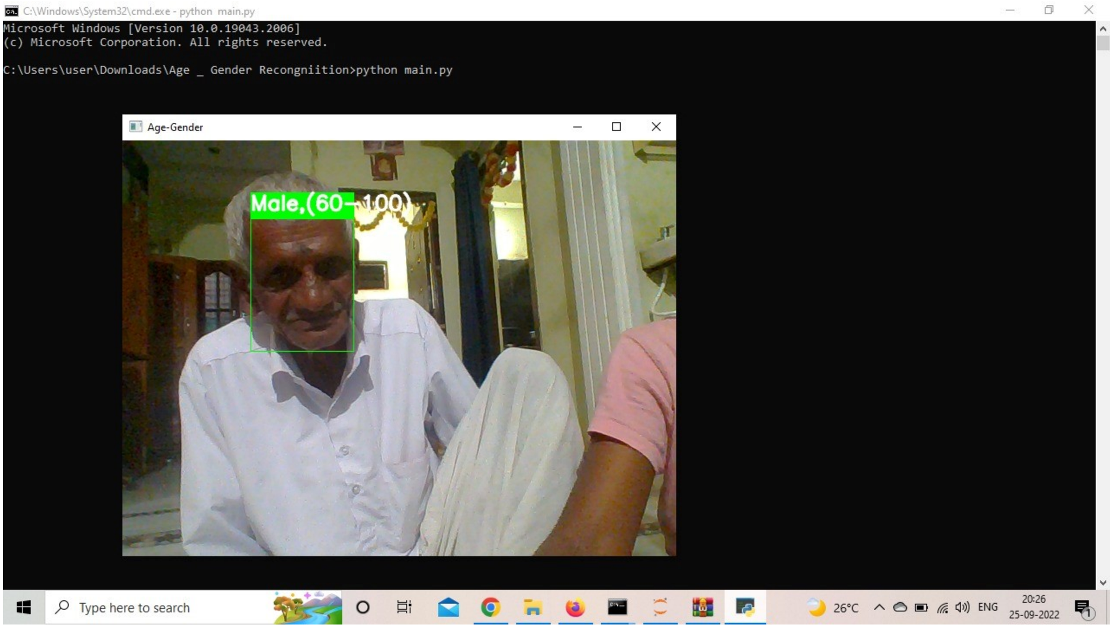


Figure 5.2: Person 2

72, the model predicted gender to be male and age range as 60-100. The age of the person in Fig. 6 is 21 with the gender male, the model predicted it as male and age range between 21-32. Therefore through this model, we can see that age and gender recognition is done accurately where the image is being taken from a webcam i.e., live image as input, and using CNN for classification.

Table 5.1: Actual age and gender

Sr	Actual age	Actual gender
1	22	female
2	72	male
3	21	male

Table 5.2: Predicted age and gender

Sr	Predicted age	Predicted gender
1	21-32	female
2	60-100	male
3	21-32	male

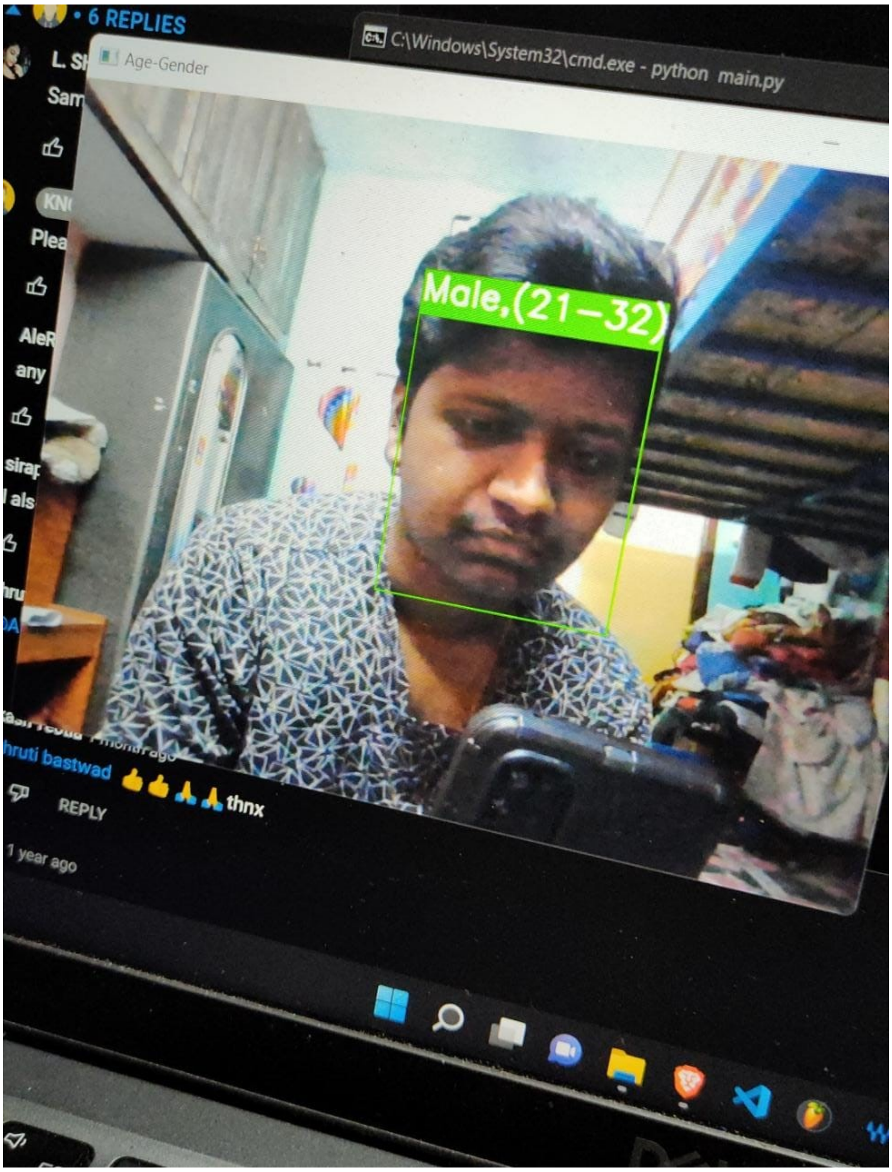


Figure 5.3: Person 3

CHAPTER 6

Conclusions and Future Scope

6.1 Conclusion

The study comes to the conclusion that age and gender research has recently gained popularity. In the past, many tactics have concentrated on age and sexuality-related difficulties, but more recently, this research has concentrated on spectacular photos obtained in lab settings. for supplying the appropriate and essential conditions for the testing environment. reflects how current real-world photography generally appears on social networking platforms and online archives. The intricacy of web images is not the only issue. In terms of saturation, the same is true. You can easily access a big library of excellent films that provide continual preparation with knowledge. CNN enables us to make age and age order suggestions without even glancing at the most simplistic diagram of age and sexuality.

6.2 Future Scope

We can extend the reach of our information by including incorrectly cropped image variants in our kit. The framework was tested on unfiltered images from the Audience benchmark and basically appeared to outperform Late-Sharp. Using the experimental results, we can draw two important conclusions.

First, CNN allows us to get better results when sorted by age and gender, even though the currently unbounded image set named by age and gender is of a very small size. Second, the correctness of our model suggests that a participation framework that uses more preparatory information could significantly improve results beyond those reported here.

Firstly, age and gender prediction can be used for marketing and advertising. Advertisers can use this technology to target specific age and gender groups with personalized ads. This will not only increase the effectiveness of the advertisement but also reduce the cost of advertising. Additionally, age and gender prediction can be used to analyze consumer behavior and preferences, which can be used to create new marketing strategies.

Secondly, age and gender prediction can be used in the healthcare industry. This information can be used to create preventive measures to reduce the risk of developing these diseases. Gender prediction can be used to develop personalized treatment plans for patients. For instance, women and men have different reactions to medications, and gender prediction can be used to create personalized treatment plans that are specific to the patient's gender.

Thirdly, age and gender prediction can be used in the entertainment industry. For instance, age prediction can be used to create personalized movie recommendations for viewers. Gender prediction can be used to recommend movies or TV shows that are more likely to be of interest to the viewer based on their gender. Additionally, age and gender prediction can be used to develop more personalized video games that are tailored to the preferences of the player.

Fourthly, age and gender prediction can be used in the field of security. For instance, age and gender prediction can be used in surveillance cameras to identify potential threats. This can help to prevent crimes before they happen. Additionally, age and gender prediction can be used in airport security to identify potential terrorists. This can help to prevent terrorist attacks and keep people safe.

Fifthly, age and gender prediction can be used in the field of education. For instance, age prediction can be used to identify the learning capabilities of students. This information can be used to create personalized learning plans that are specific to the student's age. Gender prediction can be used to create gender-specific teaching plans that are tailored to the needs of male and female students.

Lastly, age and gender prediction can be used in the field of social media. For instance, age and gender prediction can be used to create personalized social media recommendations for users. This can help to improve user engagement and retention. Additionally, age and gender prediction can be used to identify potential cyberbullies and prevent cyberbullying. This can help to create a safer and more positive social media environment.

In conclusion, age and gender prediction is a rapidly growing field that has a wide range of potential applications. The accuracy of age and gender prediction algorithms is improving, and the potential applications of this technology are expanding. It is likely that in the future, age and gender prediction will be used in many different industries to create more personalized products and services.

REFERENCES

[1] Hu Han, Charles Otto, and Anil K Jain. "Age estimation from face images: Human vs. machine performance". In: (2013), pp. 1–8.

[2] K Venkateswara Rao, G Harika, A Blessy, D Vishnu Vardhan Reddy, and B Rakesh. "Age Estimation and Gender prediction of Unfiltered Faces Using Deep Learning Techniques". In: ().

[3] Asif Ahmed, Soumitra Saha, Sudip Saha, Md Bipul, Nasif Muslim, Salekul Islam, et al. "Real-time face recognition based on IoT: A comparative study between IoT platforms and cloud infrastructures". In: *Journal of High Speed Networks* 26.2 (2020), pp. 155–168.

[4] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. "Vggface2: A dataset for recognising faces across pose and age". In: *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*. IEEE. 2018, pp. 67–74.

[5] MAH Akhand, Md Ijaj Sayim, Shuvendu Roy, and N Siddique. "Human age prediction from facial image using transfer learning in deep convolutional neural networks". In: *Proceedings of International Joint Conference on Computational Intelligence: IJCCI 2019*. Springer. 2020, pp. 217–229.

[6] Liang Hu, Zheyuan Li, and Hong Liu. "Age group estimation on single face image using blocking ULBP and SVM". In: *Proceedings of the 2015 Chinese Intelligent Automation Conference: Intelligent Information Processing*. Springer. 2015, pp. 431–438.

[7] Arna Fariza, Agus Zainal Arifin, et al. "Age estimation system using deep residual network classification method". In: *2019 International Electronics Symposium (IES)*. IEEE. 2019, pp. 607–611.

[8] Rasmus Rothe, Radu Timofte, and Luc Van Gool. "Dex: Deep expectation of apparent age from a single image". In: *Proceedings of the IEEE international conference on computer vision workshops*. 2015, pp. 10–15.

[9] Paul Viola and Michael Jones. "Rapid object detection using a boosted cascade of simple features". In: *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*. Vol. 1. Ieee. 2001, pp. I–I.

[10] Amit Kumar, Fahimuddin Shaik, Amit Kumar, and Fahimuddin Shaik. "Importance of image processing". In: *Image Processing in Diabetic Related Causes* (2016), pp. 5–7.