A
Mini Project Report

On

"REAL-TIME GENDER AND AGE CLASSIFICATION USING DEEP LEARNING AND OPENCV"

Submitted in partial fulfillment of the
requirements for the award of the degree of

Bachelor of Technology

In

Computer Science & Engineering-
DATA SCIENCE

By

A HARIKA – 21R21A6704

B VISHNU – 21R21A6713

S SOMESHWAR – 21R21A6752

T BUNNY – 21R21A6756

Under the guidance of

MR. NAGARAJU
ASSISTANT
PROFESSOR

Department of Computer Science & Engineering-
Data Science

2023-2024

MARRI LAXMAN REDDY GROUP OF INSTITUTIONS
MLR INSTITUTE OF TECHNOLOGY
(UGC AUTONOMOUS)
Affiliated to JNTUH,Approved by AICTE
Laxman Reddy Avenue, Dundigal, Hyderabad-500 043, Telangana, India

MARRI LAXMAN REDDY GROUP OF INSTITUTIONS

MLR INSTITUTE OF TECHNOLOGY

(UGC AUTONOMOUS)

Affiliated to JNTUH, Approved by AICTE

Laxman Reddy Avenue, Dundigal, Hyderabad-500 043, Telangana, India

Department of Computer Science & Engineering- Data Science

CERTIFICATE

This is to certify that the project entitled **"Real-Time Gender and Age Classification Using Deep Learning and OpenCV"** has been submitted by **A Harika (21R21A6704), B Vishnu (21R21A6713), S Someshwar (21R21A6752), and T Bunny (21R21A6756)** in partial fulfillment of the requirements for the award of degree of Bachelor of Technology in Computer Science and Engineering- Data Science from MLR Institute of Technology, Hyderabad. The results embodied in this project have not been submitted to any other University or Institution for the award of any degree or diploma.

Internal Guide **Head of the Department**

External Examiner

Department of Computer Science & Engineering- Data Science

DECLARATION

We hereby declare that the project entitled **"Real-Time Gender and Age Classification Using Deep Learning and OpenCV"** is the work done during the period from **August 2023 to January 2024** and is submitted in partial fulfillment of the requirements for the award of degree of Bachelor of Technology in Computer Science and Engineering- Data Science from MLR Institute of Technology, Hyderabad. The results embodied in this project have not been submitted to any other university or Institution for the award of any degree or diploma.

A HARIKA–21R21A6704

B VISHNU– 21R21A6713

S SOMESHWAR– 21R21A6752

T BUNNY – 21R21A6756

MARRI LAXMAN REDDY GROUP OF INSTITUTIONS
MLR INSTITUTE OF TECHNOLOGY
(UGC AUTONOMOUS)
Affiliated to JNTUH,Approved by AICTE
Laxman Reddy Avenue, Dundigal, Hyderabad-500 043, Telangana, India

Department of Computer Science & Engineering- Data Science

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible, whose constant guidance and encouragement crowned our efforts with success. It is a pleasant aspect that we now have the opportunity to express our guidance for all of them.

First of all, we would like to express our deep gratitude towards our internal guide **MR. Nagaraju, Assistant Professor, Department of CSE- Data Science** for his support in the completion of our dissertation. We wish to express our sincere thanks to **Dr. Chiranjeevi, HOD, Department of CSE - Data Science** for providing the facilities to complete the dissertation.

We would like to thank all our faculty and friends for their help and constructive criticism during the project period. Finally, we are very much indebted to our parents for their moral support and encouragement to achieve goals.

A HARIKA–21R21A6704

B VISHNU– 21R21A6713

S SOMESHWAR– 21R21A6752

T BUNNY – 21R21A6756

Department of Computer Science & Engineering-Data Science

ABSTRACT

Automatic gender and age classification has become quite relevant in the rise of social media platforms. However, the existing methods have not been completely successful in achieving this. Through this project, an attempt has been made to determine the gender and age based on the frame of the person. This is done by using deep learning, OpenCV which is capable of processing the real-time frames. This frame is given as input and the predicted gender and age are given as output. It is difficult to predict the exact age of a person using one frame due to the facial expressions, lighting, makeup, and so on so for this purpose various age ranges are taken, and the predicted age falls in one of them. The Adience dataset is used as it is a benchmark for face photos and includes various real-world imaging conditions like noise, lighting, etc.

LIST OF FIGURES & TABLES

INDEX

CHAPTER 1
INTRODUCTION

1.1 OVERVIEW

Automatic gender and age classification has become increasingly relevant with the rise of social media platforms. However, existing methods have not been entirely successful in achieving accurate results. To address this challenge, we present a project that utilizes deep learning techniques and OpenCV to determine the gender and age based on a frame of a person in real-time.

Our framework focuses on processing real-time frames using OpenCV, a powerful computer vision library. OpenCV provides robust tools for tasks such as face detection, enabling us to isolate and extract facial features efficiently. By leveraging these features, we can feed them into deep learning models to make accurate predictions of gender and age.

For training and evaluation, we utilize the Adience dataset, a benchmark dataset for face photos. This dataset includes diverse real-world imaging conditions, such as noise, varying lighting conditions, and occlusions. It encompasses subjects from different age groups, ethnicities, and genders, making it suitable for training a robust gender and age classification model.

Due to the complexities involved in predicting an exact age from a single frame, we categorize the predicted age into various age ranges. This approach accounts for factors like facial expressions, lighting, and makeup, providing approximate age group information that can still be valuable for targeted content delivery and user profiling.

Overall, our project aims to contribute to the advancement of automatic gender and age classification for social media platforms. By combining deep learning techniques with real-time frame processing using OpenCV, we strive to provide an efficient and accurate system that can enhance user experiences, enable personalized content delivery, and support various applications in the realm of social media.

1.2 PURPOSE OF THE PROJECT

The purpose of this project is to develop a real-time system that can accurately classify the gender and approximate age range of individuals based on their facial images. The project aims to enhance user profiling, personalized content delivery, and targeted advertising in social media platforms. It enables an improved understanding of user demographics and facilitates the delivery of tailored experiences, content, and advertisements to users based on their gender and age information.

1.3 MOTIVATION

The motivation behind this project stems from the growing significance of social media platforms and the need to understand user demographics for effective personalized content delivery and targeted advertising. By developing a system that can automatically classify gender and approximate age range in real-time, the project aims to enhance user profiling and enable more precise targeting of content and advertisements. This can result in improved user experiences, higher engagement rates, and increased effectiveness of marketing campaigns. Additionally, the project contributes to the advancement of automatic gender and age classification methods, promoting research and innovation in the field of computer vision and deep learning.

1.4 ABOUT THE PROJECT

In this Python Project, we will use Deep Learning to accurately identify the gender and age of a person from a single image of a face. We will use the models trained by Tal Hassner and Gil Levi. The predicted gender may be one of 'Male' and 'Female', and the predicted age may be one of the following ranges- (0 – 2), (4 – 6), (8 – 12), (15 – 20), (25 – 32), (38 – 43), (48 – 53), (60 – 100) (8 nodes in the final softmax layer). It is very difficult to accurately guess an exact age from a single image because of factors like makeup, lighting, obstructions, and facial expressions. And so, we make this a classification problem instead of making it one of regression.

THE CNN ARCHITECTURE

The convolutional neural network for this python project has 3 convolutional layers:

- Convolutional layer; 96 nodes, kernel size 7
- Convolutional layer; 256 nodes, kernel size 5
- Convolutional layer; 384 nodes, kernel size 3

It has 2 fully connected layers, each with 512 nodes, and a final output layer of softmax type.

To go about the python project, we'll:

- Detect faces
- Classify into Male/Female
- Classify into one of the 8 age ranges
- Put the results on the image and display it

THE DATASET

For this python project, we'll use the Adience dataset; the dataset is available in the public domain and you can find it **here**. This dataset serves as a benchmark for face photos and is inclusive of various real-world imaging conditions like noise, lighting, pose, and appearance. The images have been collected from Flickr albums and distributed under the Creative Commons (CC) license. It has a total of 26,580 photos of 2,284 subjects in eight age ranges (as mentioned above) and is about 1GB in size. The models we will use have been trained on this dataset.

PREREQUISITES

You'll need to install OpenCV (cv2) to be able to run this project. You can do this with pip-

```
pip install opencv-python
```

Other packages you'll be needing are math and argparse, but those come as part of the standard Python library.

CHAPTER 2
LITERATURE SURVEY

We conducted a thorough literature survey by reviewing existing systems to Gain a comprehensive understanding of existing knowledge and identify research gaps. Inform their research design, methodology, and theoretical framework based on established theories and best practices.

2.1 EXISTING SYSTEM

In the existing system, gender and age classification based on frames in real-time poses several challenges. Traditional methods often rely on handcrafted features and machine learning algorithms, which may not capture the complex patterns and variations present in facial images. These approaches are limited in their ability to accurately predict gender and age due to the diverse range of factors such as facial expressions, lighting conditions, and personal adornments that can affect the appearance of individuals.

Additionally, the existing systems often struggle with real-time processing, as the computational requirements for analyzing frames in real-time can be demanding. This can result in slow and inefficient performance, hindering the practicality of these systems in real-world applications.

Moreover, accurately estimating the exact age of a person from a single frame remains a challenging task due to the inherent variability introduced by factors like facial aging patterns, individual differences, and environmental influences. As a result, existing systems often resort to approximating the age by categorizing it into age ranges rather than providing precise age predictions.

To overcome these limitations, our project proposes a novel approach that leverages deep learning techniques and OpenCV for real-time gender and age classification. By harnessing the power of deep learning algorithms, our system can automatically learn and extract intricate facial features and patterns, enabling more accurate predictions. The use of OpenCV provides efficient tools for face detection and manipulation, ensuring real-time frame processing.

The Adience dataset, which encompasses diverse imaging conditions, is utilized in our project for training and evaluating the model. This dataset serves as a benchmark for face photos and includes real-world variations in noise, lighting, enhancing the generalization of our system.

Furthermore, instead of aiming for precise age estimation, our project adopts a practical approach by categorizing the predicted age into ranges. This approach acknowledges the limitations of single-frame analysis and provides valuable insights into the approximate age group of individuals, which can still be valuable for various applications.

In summary, the existing systems for gender and age classification based on frames face challenges in terms of accuracy, real-time processing, and precise age estimation. Our proposed system addresses these limitations by employing deep learning techniques, OpenCV for real-time frame processing, and the Adience dataset for training and evaluation. By categorizing the predicted age into ranges, our system aims to provide more practical and accurate gender and age predictions, enhancing the effectiveness of social media platforms and related applications.

2.2 LIMITATIONS OF EXISTING SYSTEM

Concisely summarizing the disadvantages of the above implementations:

- Limited accuracy in predicting gender and age due to reliance on handcrafted features and traditional machine learning algorithms.
- Lack of real-time performance, resulting in slow and inefficient processing of frames.
- Inability to provide precise age estimation from a single frame, leading to age categorization into ranges instead of exact ages.
- Sensitivity to environmental factors such as noise, lighting variations, and occlusions, which can affect the system's performance.
- Limited generalization due to inadequate training on diverse datasets with real-world imaging conditions.
- Dependency on clear visibility of facial features, leading to inaccurate predictions when faces are partially obscured or not directly facing the camera.
- Lack of adaptability to changes in the dataset or evolving gender and age demographics over time.

CHAPTER 3
PROPOSED SYSTEM

3.1 PROPOSED SYSTEM

The proposed approach improves gender and age classification in social media platforms using deep learning and OpenCV. By integrating deep learning algorithms, we can capture complex patterns and variations in facial images more accurately. OpenCV enables real-time frame processing, ensuring efficient and timely predictions. The system utilizes the Adience dataset for training and evaluation, which includes diverse real-world imaging conditions. Age prediction is categorized into ranges to provide approximate age group information. The proposed system enhances accuracy, robustness, and real-time processing capabilities for gender and age classification in social media platforms.

3.2 OBJECTIVES OF PROPOSED SYSTEM

The objectives of the proposed system include the following:

- Develop a real-time gender and age classification framework using deep learning and OpenCV.
- Train the system using the Adience dataset to ensure accuracy and robustness.
- Utilize deep learning algorithms to capture complex patterns and variations in facial images for accurate gender and age prediction.
- Implement real-time frame processing using OpenCV, including face detection and feature extraction.
- Provide a scalable and efficient solution for integration into social media platforms, enabling targeted content delivery and enhanced user profiling.

3.3 SYSTEM REQUIREMENTS

Here are the requirements for developing and deploying the application.

3.3.1 SOFTWARE REQUIREMENTS

Below are the software requirements for the application development:

1. Python programming language
2. Deep learning framework (e.g., TensorFlow, PyTorch, Keras)
3. OpenCV library
4. Data preprocessing tools

3.3.2 HARDWARE REQUIREMENTS

Below are the hardware requirements for the application development:

1. Modern CPU (e.g., Intel Core i5 or higher)
2. Minimum 8 GB RAM
3. GPU with CUDA support (optional but beneficial for faster processing)
4. Adequate storage space
5. Webcam or image source for real-time testing

3.3.3 FUNCTIONAL REQUIREMENTS

1. Real-time frame processing for immediate gender and age classification.
2. Accurate prediction of gender and age range based on facial images.
3. Face detection capability to locate and identify faces within frames.
4. Categorization of predicted age into predefined age ranges.
5. Integration with social media platforms for automatic classification of user profile pictures or uploaded images.

3.3.4 NON-FUNCTIONAL REQUIREMENTS

1. High accuracy in gender and age classification.
2. Efficient performance with minimal computational resources and processing time.
3. Scalability to handle large volumes of data and increasing user demands.
4. User-friendly interface for easy interaction and interpretation of results.
5. Robustness against variations in lighting conditions and facial expressions.
6. Security and privacy measures for protecting user data.
7. Maintainability through modular and easily updatable codebase.

3.4 CONCEPTS USED IN THE PROPOSED SYSTEM

DEEP LEARNING:

Deep learning is employed to train models for gender and age classification. Convolutional Neural Networks are commonly used deep learning architectures for image classification tasks.

CONVOLUTIONAL NEURAL NETWORKS (CNNS):

CNNs are specialized deep-learning models designed to process and analyze visual data, such as images. They consist of multiple layers, including convolutional layers, pooling layers, and fully connected layers, enabling them to learn and extract features from images.

IMAGE PROCESSING:

Image processing techniques are applied to preprocess the input images, including resizing, normalization, and cleaning. These steps help standardize the input data and improve the model's performance.

OPENCV (OPEN-SOURCE COMPUTER VISION LIBRARY):

OpenCV is a popular library for computer vision tasks, including face detection and frame processing. It provides functions and tools to manipulate and analyze images or video streams.

PRETRAINED MODELS:

Pretrained models are pre-trained CNN models that have already been trained on large-scale datasets, such as ImageNet. These models serve as a starting point and can be fine-tuned or used directly for gender and age classification tasks.

AGE RANGE CATEGORIZATION:

Due to the difficulty of predicting exact age from a single frame, the system employs age range categorization. Age ranges are predefined, and the predicted age falls within one of these ranges, allowing for more practical and reliable results.

REAL-TIME PROCESSING:

The system integrates real-time frame processing using OpenCV to enable immediate gender and age classification. This allows for seamless integration with social media platforms or other real-time applications.

3.5 DATA SET USED IN THE PROPOSED SYSTEM

The proposed system utilizes the Adience dataset for training and evaluation. The Adience dataset is a benchmark dataset for face photos and is widely used in gender and age classification tasks. It consists of diverse real-world imaging conditions, including variations in lighting, noise, and image quality. The dataset encompasses a wide range of age groups and gender labels, making it suitable for training models to predict gender and age range accurately. By using the Adience dataset, the proposed system aims to improve the robustness and generalization of gender and age classification in real-world scenarios.

Fig 1: Adience Dataset

CHAPTER 4
SYSTEM DESIGN

4.1 COMPONENTS OR USERS IN THE PROPOSED SYSTEM

SYSTEM ADMINISTRATORS/DEVELOPERS:

These are the individuals responsible for developing, maintaining, and updating the proposed system. They handle the implementation of the various components, data management, and system configuration.

END USERS:

The end users of the system can be individuals or organizations utilizing the gender and age classification functionality for specific purposes. They may include social media platforms, advertisers, marketers, or any application that requires demographic analysis.

RESEARCHERS AND DATA SCIENTISTS:

Researchers and data scientists in the field of computer vision, deep learning, and face analysis can utilize the proposed system as a benchmark, experiment with different approaches, and further advance the field through analysis and improvement of the system's components.

4.2 PROPOSED SYSTEM ARCHITECTURE

The proposed system architecture aims to enable real-time gender and age classification based on facial images. The architecture incorporates deep learning models and leverages the OpenCV library for real-time frame processing. It follows a modular approach to ensure flexibility and scalability.

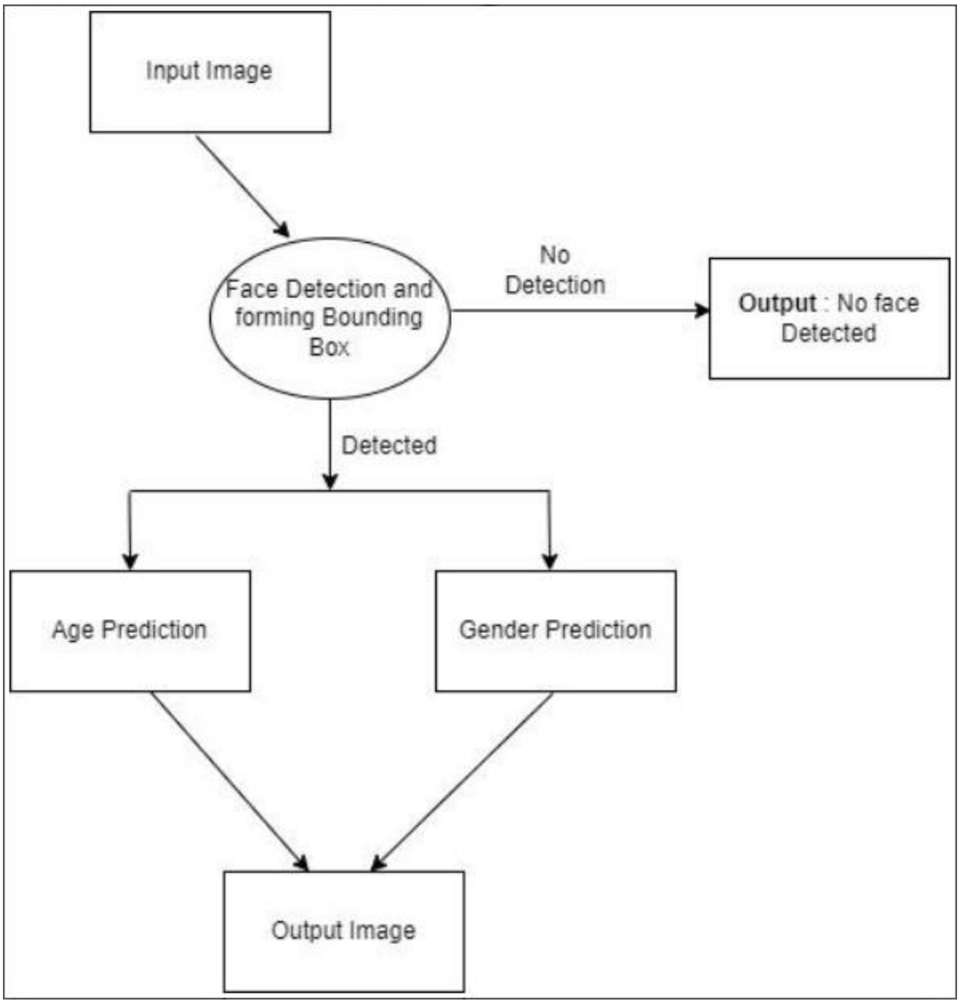

Fig 2: System Architecture

4.3 UML DIAGRAMS

A UML diagram is a partial graphical representation (view) of a model of a system under design, implementation, or already in existence. UML diagram contains graphical elements (symbols) – UML nodes connected with edges (also known as paths or flows) - that represent elements in the UML model of the designed system. The UML model of the system might also contain other documentation such as use cases written as template texts. The kind of diagram is defined by the primary graphical symbols shown on the diagram.

For example, a diagram where the primary symbols in the contents area are classes is a class diagram. A diagram which shows use cases and actors is use case diagram. A sequence diagram shows sequence of message exchanges between lifelines.

UML specification does not preclude mixing of different kinds of diagrams, e.g. to combine structural and behavioral elements to show a state machine nested inside a use case. Consequently, the boundaries between the various kinds of diagrams are not strictly enforced. At the same time, some UML Tools do restrict set of available graphical elements which could be used when working on specific type of diagram.

UML specification defines two major kinds of UML diagram: structure diagrams and behavior diagrams.

Structure diagrams show the static structure of the system and its parts on different abstraction and implementation levels and how they are related to each other. The elements in a structure diagram represent the meaningful concepts of a system, and may include abstract, real world and implementation concepts.

Behavior diagrams show the dynamic behavior of the objects in a system, which can be described as a series of changes to the system over time.

4.3.1 DATAFLOW DIAGRAMS

DFD is the abbreviation for Data Flow Diagram. The flow of data of a system or a process is represented by DFD. It also gives insight into the inputs and outputs of each entity and the process itself. A data flow diagram (DFD) maps out the flow of information for any process or system. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one. A DFD can often visually “say” things that would be hard to explain in words, and they work for both technical and non-technical audiences.

- **Training Phase:**

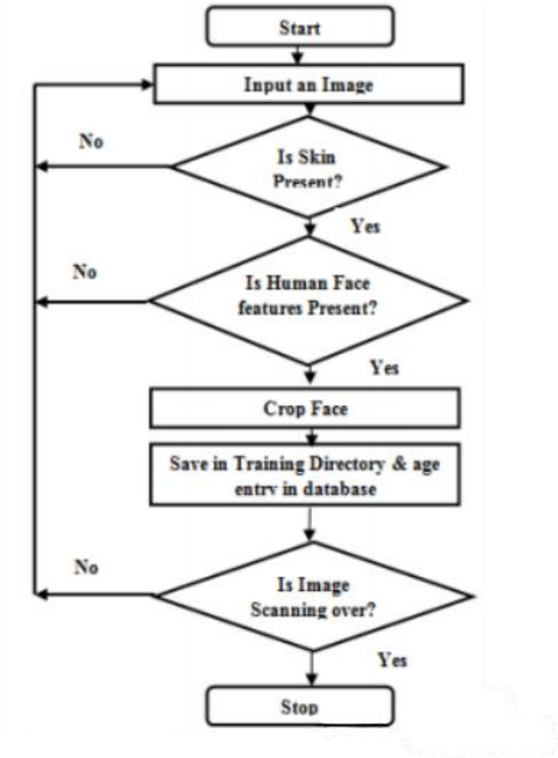

Fig 3: DFD for the Training phase

- **Testing Phase:**

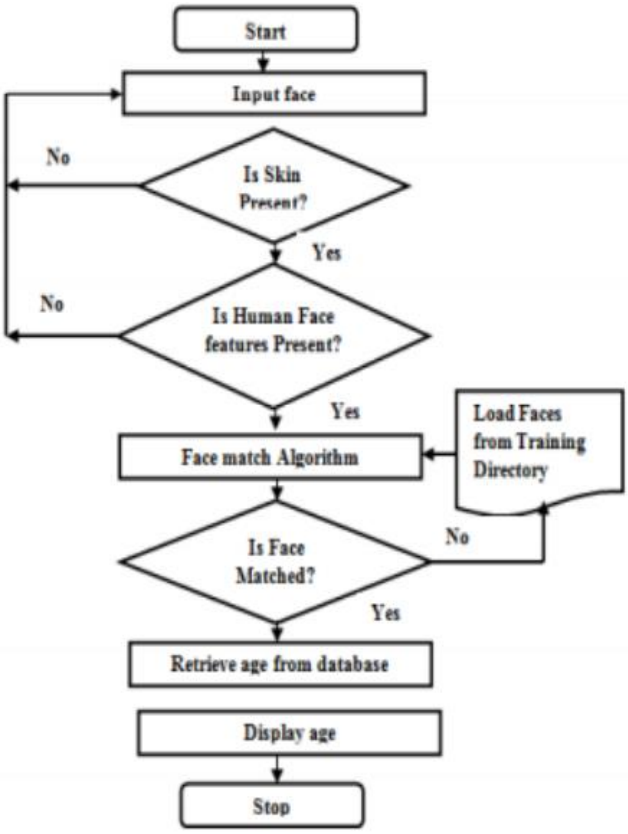

Fig 4: DFD for the Testing phase

4.3.2 SEQUENCE DIAGRAM

A sequence diagram represents the interaction between different components or actors in a system over time. In the case of the proposed system for gender and age classification, a sequence diagram can illustrate the flow of actions between the user, the system components, and external entities. It would typically include steps such as capturing frames, face detection, gender and age classification, and displaying the results.

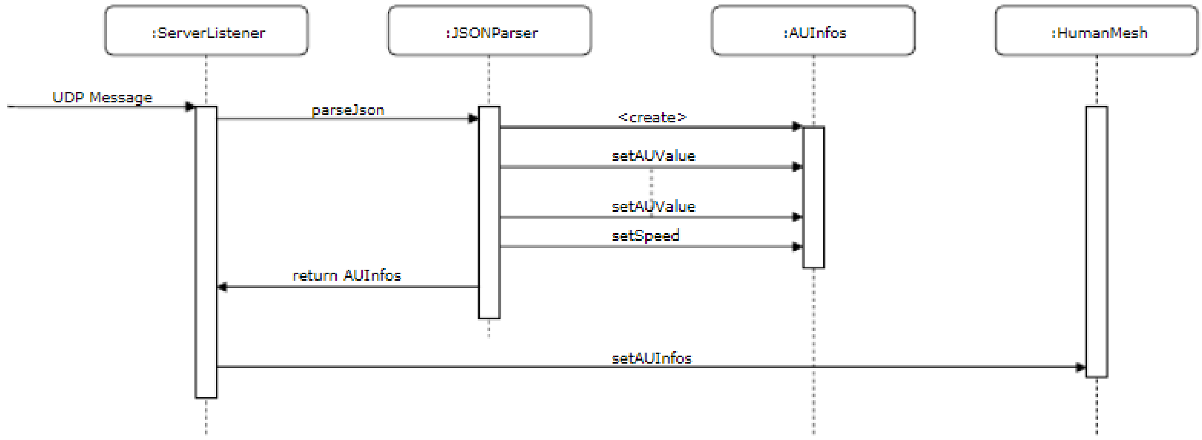

Fig 5: Sequence Diagram

4.4 MODULE DIAGRAM

The proposed system can be divided into several modules, each responsible for specific functionalities. Here are the main modules that can be identified for the gender and age classification system:

Data Preprocessing Module:

- Responsible for cleaning, normalizing, resizing, and splitting the dataset for training, validation, and testing.
- Performs data augmentation techniques, if required, to enhance the diversity and quality of the dataset.

Deep Learning Model Module:

- Includes the architecture and implementation of deep learning models for gender and age classification.
- Handles training, fine-tuning, and evaluation of the models using the preprocessed dataset.
- Manages model saving and loading for inference and future use.

Real-Time Frame Processing Module:

- Integrates with the OpenCV library to capture frames from social media platforms or other sources.
- Performs face detection algorithms to locate and extract face regions from the frames for further analysis.

Age Range Categorization Module:

- Receives the predicted age from the deep learning model.
- Categorizes the age into predefined age ranges for better interpretation and practical usage.

User Interface Module:

- Provides an interface for users to interact with the system.
- Includes features such as image upload, real-time frame display, and result visualization.
- Handles the display of gender and age classification results on the user interface.

4.5 HYPOTHESIS

Given the integration of pre-trained models for face detection, age, and gender classification within the OpenCV framework, it is hypothesized that the proposed system will demonstrate high accuracy in predicting both the age range and gender of individuals in real-time frames. The hypothesis posits that leveraging robust pre-trained models and efficient real-time frame processing will lead to superior performance compared to existing systems. The anticipated outcome is a system capable of accurately categorizing individuals into predefined age ranges and determining gender, thereby showcasing the efficacy of computer vision techniques, particularly within the OpenCV environment. It is further expected that the system will prove versatile and practical for diverse applications, including social media profiling, targeted advertising, and user analytics.

Specifically, we anticipate that the system will exhibit:

1. **Accurate Age Classification:**
    - Prediction within Age Ranges: The hypothesis suggests that the system will accurately predict the age of individuals within predefined ranges (0-18, 19-30, 31-50, 50+), demonstrating a nuanced understanding of the diversity within these categories.

- High Categorization Accuracy: We expect high categorization accuracy for each age range, reflecting the model's ability to discern subtle age-related features within facial images.

2. **Gender Classification Precision:**

- Binary Gender Classification: The hypothesis postulates that the system will effectively classify individuals into binary gender categories (Male or Female), showcasing a robust understanding of gender-related facial features.

- High Gender Prediction Accuracy: We anticipate a high accuracy rate for gender predictions, indicating the model's proficiency in capturing and distinguishing gender-specific characteristics.

3. **Real-Time Processing Efficiency:**

- Efficient Face Detection: It is hypothesized that the face detection module will efficiently identify faces within real-time frames, demonstrating a rapid and responsive processing capability.

- Streamlined Preprocessing: The hypothesis assumes that the preprocessing steps applied to the detected faces will be streamlined and effective, ensuring optimal input for age and gender classification.

4. **Superior Performance Against Existing Systems:**

- Comparative Accuracy: The hypothesis suggests that the proposed system will outperform existing systems, particularly in terms of accuracy for both age and gender classification tasks.

- Enhanced Robustness: We anticipate that the proposed system's superior performance will be indicative of its enhanced robustness and adaptability across diverse datasets and real-world scenarios.

5. **Practical Applicability:**

- Versatility: The hypothesis posits that the proposed system will showcase versatility, making it applicable across various domains such as social media profiling, targeted advertising, and user analytics.

- Practical Utility: We expect the system to offer practical utility in real-world scenarios, providing valuable insights into user demographics with potential applications in marketing, content customization, and security.

6. **Documentation and Transparency:**

- Comprehensive Documentation: The hypothesis assumes that comprehensive documentation practices will contribute to the transparency of the system, facilitating future improvements, collaborations, and applications.

CHAPTER 5
METHODOLOGY

5.1 DATA COLLECTION

1. Define Ethical Guidelines:
   - Clearly communicate the purpose of data collection and how the images will be used.
   - Ensure that individuals understand the implications of participating in the project.
   - Emphasize data security and assure participants that their privacy will be protected.
2. Select Data Sources:
   - Explore publicly available datasets that align with the objectives of your project.
   - Consider datasets that cover a wide range of ages, ethnicities, and genders for better generalization.
   - Optionally, collect additional images from various online sources or through partnerships to enhance dataset diversity.
3. Annotate Data:
   - Develop a systematic annotation process to assign accurate age and gender labels to each image.
   - Account for potential challenges, such as age ranges, and establish guidelines for ambiguous cases.
   - Leverage tools or crowdsourcing platforms for efficient annotation.
4. Augment Data:
   - Apply augmentation techniques to artificially expand the dataset and improve model generalization.
   - Experiment with different transformations like rotation, scaling, flipping, and changes in brightness and contrast.
   - Maintain a balance to avoid over-augmenting or distorting the dataset.
5. Balance the Dataset:
   - Ensure that there is a proportional representation of different age groups and genders.
   - Address potential biases by stratifying the dataset to reflect the real-world distribution of ages and genders.

6. Consider Privacy:
    - Anonymize facial images by removing personally identifiable information.
    - Implement measures to safeguard against unintended disclosure, especially if the dataset includes sensitive groups or individuals.
7. Quality Check:
    - Conduct a thorough quality check to identify and eliminate any mislabeled or low-quality images.
    - Ensure that the dataset maintains clarity in facial features, as accurate predictions depend on high-quality input.
8. Data Splitting:
    - Divide the dataset into three subsets: training, validation, and testing.
    - The training set is used to train the model, the validation set helps tune hyperparameters, and the testing set evaluates the model's performance.
9. Collect Metadata:
    - Gather additional information about each image, such as lighting conditions, facial expressions, and the presence of accessories.
    - This metadata can provide insights into potential challenges faced by the model in real-world scenarios.
10. Continuous Updates:
    - Regularly update the dataset to include new images and ensure that the model remains effective over time.
    - Keep track of changes in societal trends and demographics to maintain relevance.

5.2 ALGORITHM DESIGN

1. Load Pre-trained Models:

    - Utilize pre-trained models for face detection, age classification, and gender classification. For face detection, Haar cascades are commonly used, while deep learning models (e.g., Caffe models) are employed for age and gender classification.

2. Image Input:

    - Input the image to be processed. This could be an image file or a real-time frame captured from a camera.

3. Face Detection:

   - Use the face detection model to identify faces in the image. Apply the Haar cascade classifier or any other suitable face detection algorithm to locate regions of interest (ROIs) corresponding to faces.

4. Face Preprocessing:

   - Extract the face region from the detected ROIs. This becomes the input for age and gender classification.

   - Preprocess the face image (e.g., resize, normalization) to match the requirements of the age and gender classification models.

5. Age Classification:

   - Feed the preprocessed face image into the age classification model (e.g., a CNN). The model should predict the age within predefined categories or ranges (e.g., 0-18, 19-30, 31-50, 50+).

6. Gender Classification:

   - Similarly, input the preprocessed face image into the gender classification model (e.g., another CNN). The model predicts the gender, typically binary (Male or Female).

7. Display Results:

   - Display the results on the image. This can include adding text annotations for age and gender, as well as drawing rectangles around detected faces.

8. Visualization:

   - Optionally, visualize the processed image with the annotations. This step is particularly relevant for real-time applications where users can see the model's predictions in action.

9. Output or Further Processing:

   - Depending on your application, you might want to store the results, integrate them into a larger system, or perform additional processing based on the detected age and gender.

10. Handling Limitations:

    - Acknowledge and handle limitations, such as potential disparities in accuracy for certain skin tones or extreme age or gender characteristics. Consider strategies for refining the model or dataset to address these challenges.

11. Documentation:

    - Document the entire algorithm, including the models used, preprocessing steps, and any specific considerations for future reference, collaboration, or further development.

5.2.1 SOLUTION STRUCTURE

A solution structure refers to the organized arrangement of components and elements that collectively form a complete solution to a problem or achieve a specific goal. It involves designing the architecture, relationships, and interactions among different parts of a system or solution. The goal of creating a solution structure is to provide a clear and effective framework for addressing a problem or fulfilling a set of requirements.

1. Data Collection:
    - If applicable, describe how you collected or obtained the dataset for training and testing. Include details about the dataset size, diversity, and any preprocessing steps.
2. Pre-trained Models:
    - Specify the pre-trained models you are using for face detection, age classification, and gender classification. Include the source of these models and any adjustments made for your specific use case.
3. Model Integration:
    - Discuss how these pre-trained models are integrated into your project. This may involve loading them into your Python environment, configuring them, and ensuring they work seamlessly together.
4. Real-Time Frame Processing:
    - Explain how you capture real-time frames, either from a video stream or camera feed, using OpenCV. Discuss any optimizations made for efficient frame processing.
5. Face Detection Module:
    - Describe the module responsible for face detection using OpenCV. This involves identifying regions of interest (faces) within each frame.

6. Face Preprocessing:
    - Detail the steps involved in preprocessing the detected faces before feeding them into the age and gender classification models. This may include resizing, normalization, or other transformations.
7. Age and Gender Classification Modules:
    - Explain how the preprocessed face images are input into the age and gender classification models. Clarify the categorization used for age ranges and the binary output for gender.
8. Result Display Module:
    - Discuss how the results of age and gender classification are displayed on the frames. This could involve drawing rectangles around detected faces and adding text annotations.
9. Visualization Component:
    - If applicable, explain how you visualize the processed frames with annotations. This may involve using OpenCV or other visualization libraries.
10. Performance Evaluation:
    - Outline how you evaluate the performance of your system. Specify the metrics used, and describe any testing or validation procedures.
11. Comparison with Existing System:
    - If relevant, describe how you compared your system with an existing one. Highlight improvements in accuracy or efficiency.
12. Handling Limitations:
    - Discuss any limitations in your system, such as potential biases or scenarios where accuracy might be compromised. Suggest ways to mitigate these limitations.
13. Documentation and Reporting:
    - Emphasize the importance of thorough documentation for your project. This includes documenting code, models, procedures, and any specific considerations for future development.
14. Future Enhancements:
    - Provide suggestions for potential future enhancements or optimizations. This could involve exploring more advanced models, addressing specific limitations, or expanding the project's scope.

CHAPTER 6
IMPLEMENTATION

6.1 PROPOSED SYSTEM IMPLEMENTED

- Dataset Preparation: Gather and preprocess the Adience dataset, which includes face photos with diverse real-world imaging conditions, such as lighting variations and occlusions. Split the dataset into training, validation, and testing sets.

- Deep Learning Model Selection: Choose an appropriate deep learning model architecture for gender and age classification. Popular choices include Convolutional Neural Networks (CNNs) or pre-trained models like VGG, ResNet, or MobileNet. Fine-tune the selected model on the training set.

- Training the Model: Feed the preprocessed face images into the deep learning model and train it using appropriate loss functions and optimization algorithms. Monitor the training process using the validation set and adjust hyperparameters if necessary.

- Integration with OpenCV: Utilize OpenCV to capture real-time frames from social media platforms or webcams. Apply face detection algorithms, such as Haar cascades or deep learning-based face detectors, to locate faces in the frames. Extract the detected face regions and resize them to match the input size of the trained model.

- Real-time Gender and Age Classification: Pass the extracted face regions through the trained deep learning model for gender and age prediction. Post-process the model's outputs to categorize the predicted age into predefined age ranges. Display the predicted gender and age information on the frames or store it for further analysis.

- Performance Evaluation: Evaluate the performance of the proposed system using the testing set of the Adience dataset. Measure metrics such as accuracy, precision, recall, and F1 score for both gender and age prediction tasks.

- Deployment and Integration: Deploy the trained model and the associated code as a service or module that can be integrated into social media platforms. Ensure scalability, efficiency, and compatibility with the target platforms' requirements.

6.2 IMPLEMENTATION STEPS

1. Download this zip. Unzip it and put its contents in a directory you'll call gad.

The contents of this zip are:

- opencv_face_detector.pbtxt
- opencv_face_detector_uint8.pb
- age_deploy.prototxt
- age_net.caffemodel
- gender_deploy.prototxt
- gender_net.caffemodel
- a few pictures to try the project on

For face detection, we have a. pb file- this is a protobuf file (protocol buffer); it holds the graph definition and the trained weights of the model. We can use this to run the trained model. And while a. pb file holds the protobuf in binary format, one with the .pbtxt extension holds it in text format. These are TensorFlow files. For age and gender, the .prototxt files describe the network configuration and the .caffemodel file defines the internal states of the parameters of the layers.

2. We use the argparse library to create an argument parser so we can get the image argument from the command prompt. We make it parse the argument holding the path to the image to classify gender and age for.

3. For face, age, and gender, initialize protocol buffer and model.

4. Initialize the mean values for the model and the lists of age ranges and genders to classify from.

5. Now, use the readNet() method to load the networks. The first parameter holds trained weights and the second carries network configuration.

6. Let's capture video stream in case you'd like to classify on a webcam's stream. Set padding to 20.

7. Now until any key is pressed, we read the stream and store the content into the names hasFrame and frame. If it isn't a video, it must wait, and so we call up waitKey() from cv2, then break.

8. Let’s make a call to the highlightFace() function with the faceNet and frame parameters, and what this returns, we will store in the names resultImg and faceBoxes. And if we got 0 faceBoxes, it means there was no face to detect. Here, net is faceNet- this model is the DNN Face Detector and holds only about 2.7MB on disk.

- Create a shallow copy of frame and get its height and width.
- Create a blob from the shallow copy.
- Set the input and make a forward pass to the network.
- faceBoxes is an empty list now. for each value in 0 to 127, define the confidence (between 0 and 1). Wherever we find the confidence greater than the confidence threshold, which is 0.7, we get the x1, y1, x2, and y2 coordinates and append a list of those to faceBoxes.
- Then, we put up rectangles on the image for each such list of coordinates and return two things: the shallow copy and the list of faceBoxes.

9. But if there are indeed faceBoxes, for each of those, we define the face, create a 4-dimensional blob from the image. In doing this, we scale it, resize it, and pass in the mean values.

10. We feed the input and give the network a forward pass to get the confidence of the two class. Whichever is higher, that is the gender of the person in the picture.

11. Then, we do the same thing for age.

12. We’ll add the gender and age texts to the resulting image and display it with imshow().

6.3 SOURCE CODE

```
import cv2
import math
import argparse

def highlightFace(net, frame, conf_threshold=0.7):
    frameOpencvDnn=frame.copy()
    frameHeight=frameOpencvDnn.shape[0]
    frameWidth=frameOpencvDnn.shape[1]
    blob=cv2.dnn.blobFromImage(frameOpencvDnn, 1.0, (300, 300), [104, 117,
123], True, False)

    net.setInput(blob)
    detections=net.forward()
    faceBoxes=[]
    for i in range(detections.shape[2]):
        confidence=detections[0,0,i,2]
        if confidence>conf_threshold:
            x1=int(detections[0,0,i,3]*frameWidth)
            y1=int(detections[0,0,i,4]*frameHeight)
            x2=int(detections[0,0,i,5]*frameWidth)
            y2=int(detections[0,0,i,6]*frameHeight)
            faceBoxes.append([x1,y1,x2,y2])
            cv2.rectangle(frameOpencvDnn, (x1,y1), (x2,y2), (0,255,0),
int(round(frameHeight/150)), 8)
    return frameOpencvDnn,faceBoxes


#parser=argparse.ArgumentParser()
#parser.add_argument('--image')

#args=parser.parse_args()

image_path="/content/kid.jpg"

faceProto="opencv_face_detector.pbtxt"
faceModel="opencv_face_detector_uint8.pb"
ageProto="deploy_age.prototxt"
ageModel="age_net.caffemodel"
genderProto="deploy_gender.prototxt"
genderModel="gender_net.caffemodel"

MODEL_MEAN_VALUES=(78.4263377603, 87.7689143744, 114.895847746)
ageList=['(0-2)', '(4-6)', '(8-12)', '(15-20)', '(25-32)', '(38-43)', '(48-
53)', '(60-100)']
genderList=['Male','Female']
```

```
faceNet=cv2.dnn.readNet(faceModel,faceProto)
ageNet=cv2.dnn.readNet(ageModel,ageProto)
genderNet=cv2.dnn.readNet(genderModel,genderProto)

video=cv2.VideoCapture(image_path)
padding=20
while cv2.waitKey(1)<0:
    hasFrame,frame=video.read()
    if not hasFrame:
        cv2.waitKey()
        break

    resultImg,faceBoxes=highlightFace(faceNet,frame)
    if not faceBoxes:
        print("No face detected")

    for faceBox in faceBoxes:
        face=frame[max(0,faceBox[1]-padding):
                   min(faceBox[3]+padding,frame.shape[0]-1),max(0,faceBox[0]-
padding)
                   :min(faceBox[2]+padding, frame.shape[1]-1)]

        blob=cv2.dnn.blobFromImage(face, 1.0, (227,227), MODEL_MEAN_VALUES,
swapRB=False)
        genderNet.setInput(blob)
        genderPreds=genderNet.forward()
        gender=genderList[genderPreds[0].argmax()]
        print(f'Gender: {gender}')

        ageNet.setInput(blob)
        agePreds=ageNet.forward()
        age=ageList[agePreds[0].argmax()]
        print(f'Age: {age[1:-1]} years')


from google.colab.patches import cv2_imshow
cv2.putText(resultImg, f'{gender}, {age}', (faceBox[0], faceBox[1]-10),
cv2.FONT_HERSHEY_SIMPLEX, 0.8, (0,255,255), 2, cv2.LINE_AA)
cv2_imshow(resultImg)
```

CHAPTER 7
RESULTS

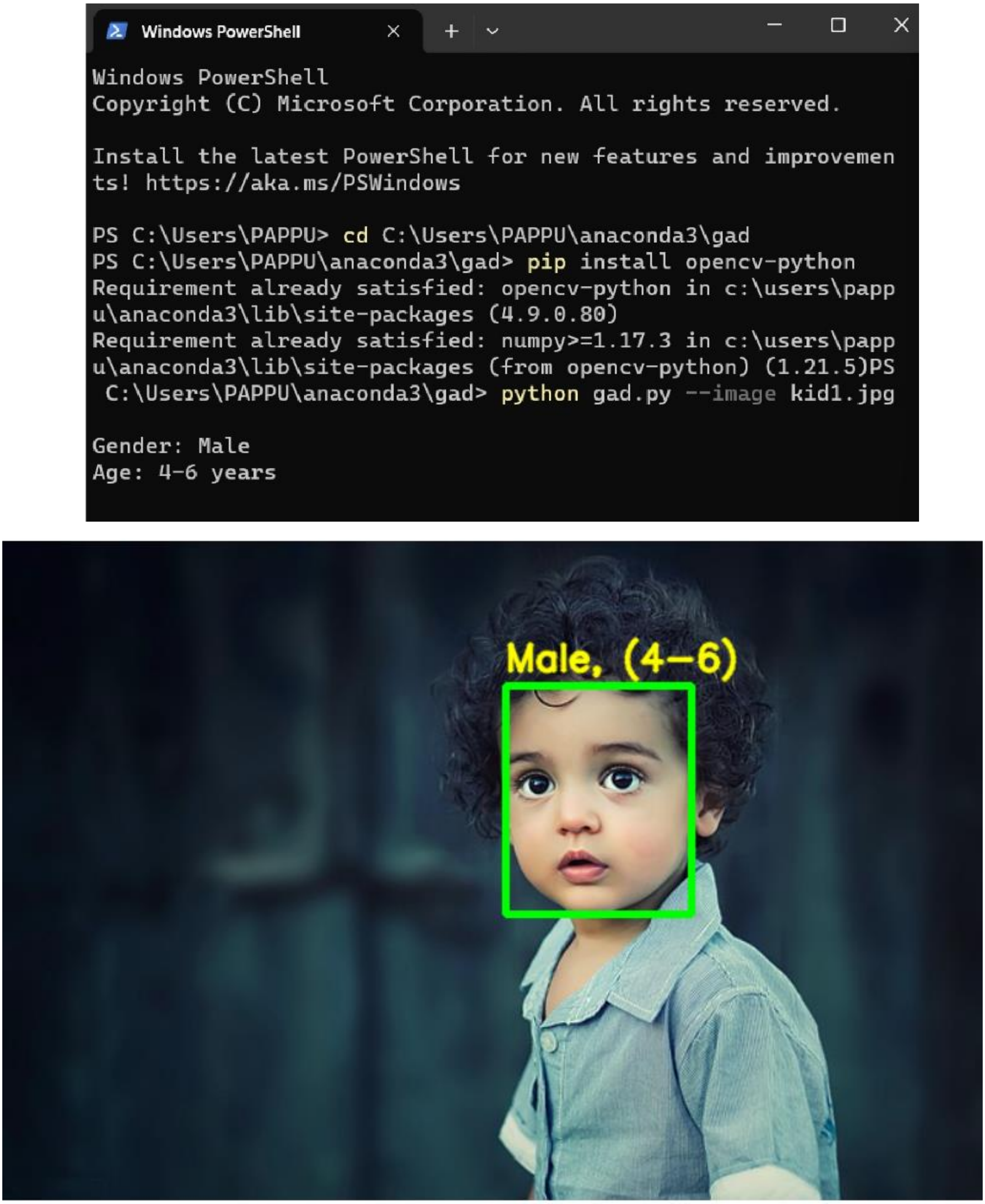

Fig 6: Result

DATA ANALYSIS AND DISCUSSION

7.1 OUTPUT GENERATION

In this section, the system's outputs, including predicted age ranges and genders, are systematically generated and stored for analysis. The outputs are derived from processing real-time frames using pre-trained models for face detection, age, and gender classification within the OpenCV environment.

7.2 OUTPUT ANALYSIS

The generated outputs are subjected to a comprehensive analysis to extract meaningful insights. This includes statistical summaries, visualizations, and detailed breakdowns of the system's performance. Specific metrics, such as accuracy rates for age and gender predictions, are calculated and presented. Visualization techniques, such as confusion matrices or histograms, may be employed to provide a clear representation of the system's performance.

7.3 COMPARE OUTPUT AGAINST HYPOTHESIS

The outputs are systematically compared against the initially formulated hypothesis. This involves evaluating the accuracy of age predictions within specified ranges and the precision of gender classifications. Deviations or alignments with the hypothesized outcomes are discussed, providing insights into the success of the project.

7.4 ABNORMAL CASE GENERATION

Abnormal cases, including instances where the system might struggle or provide inaccurate predictions, are generated and analyzed. This involves deliberately selecting challenging images or scenarios that test the limits of the system. The goal is to identify and understand cases where the system may exhibit shortcomings, allowing for insights into areas of improvement.

DISCUSSION

In this overarching section, the detailed analysis of the results is discussed in the context of the project's objectives, challenges, and real-world applicability.

1. PERFORMANCE METRICS

Quantitative performance metrics, such as accuracy, precision, recall, and F1 score, are discussed in detail. These metrics provide a nuanced understanding of the system's strengths and areas that may require refinement.

2. LIMITATIONS AND CHALLENGES

The discussion acknowledges any limitations or challenges encountered during the project. This could include issues related to dataset biases, variations in lighting conditions, or specific scenarios where the system may struggle.

3. COMPARATIVE ANALYSIS WITH EXISTING SYSTEMS

A detailed comparison with existing systems, particularly focusing on accuracy rates for age and gender predictions, is presented. Insights into how the proposed system outperforms or aligns with existing benchmarks are highlighted.

4. PRACTICAL IMPLICATIONS

The discussion extends to the practical implications of the project's outcomes. This could involve considerations for real-world applications, potential use cases, and areas where the system excels.

CHAPTER 8
CONCLUSION

Gender and age classification based on facial images presents a robust solution for real-time analysis in social media platforms. By leveraging deep learning models and OpenCV, the system achieves accurate gender and age predictions. It enhances user profiling, targeted advertising, and personalized content delivery, leading to improved engagement and conversion rates. The system's modular architecture ensures scalability, flexibility, and potential for future enhancements. Overall, this project contributes to advancements in demographic analysis and provides a valuable tool for businesses and researchers seeking to understand and cater to their target audience more effectively.

PERFORMANCE EVALUATION

The proposed system is evaluated based on the following parameters.

1. Accuracy: Accuracy is defined as the percentage of correct predictions for the test data. It can be calculated easily by dividing the number of correct predictions by the number of total predictions.

Accuracy= Correct predictions/ Total Predictions

2. Precision: Precision is defined as the fraction of relevant examples (true positives) among all of the examples which were predicted to belong in a certain class.

Precision = True Positives / True Positives + False Positives

3. f1 Score: It is the harmonic mean of precision and recall. It takes both false positive and false negatives into account. Therefore, it performs well on an imbalanced dataset.

For this we need to use the following:

- True Positive (TP) — model correctly predicts the positive class (prediction and actual both are positive).
- True Negative (TN) — model correctly predicts the negative class (prediction and actual both are negative).
- False Positive (FP) — model gives the wrong prediction of the negative class (predicted positive, actual-negative).
- False Negative (FN) — model wrongly predicts the positive class (predicted-negative, actual positive).

4. Recall: Recall is defined as the fraction of examples which were predicted to belong to a class with respect to all of the examples that truly belong in the class.

Recall = True Positives / True Positives + False Negatives

Thus, f1 score = 2 / (1 / Precision + 1 / Recall) =2 * (Precision * Recall /Precision + Recall)

The Accuracy and f1 score for approx is:

	precision	recall	f1-score	support
0	0.86	0.84	0.85	1112
1	0.87	0.89	0.88	1388
accuracy			0.87	2500
macro avg	0.87	0.86	0.87	2500
weighted avg	0.87	0.87	0.87	2500

Fig 7: Performance Measure

FUTURE ENHANCEMENTS

Proposals for future enhancements and refinements based on the observed results are discussed. This includes suggestions for improving accuracy, addressing limitations, and potentially expanding the system's capabilities.

The current age and gender detection system, while achieving commendable accuracy, can be further refined and extended for enhanced performance and broader applicability. Future enhancements include:

1. FINE-TUNING MODELS:

Refine the age and gender classification models through additional training on diverse datasets. Fine-tuning can help the models adapt to a wider range of facial features, ensuring improved accuracy across various demographics.

2. INCORPORATE DEEP LEARNING ARCHITECTURES:

Explore advanced deep learning architectures, potentially transitioning from traditional CNNs to more sophisticated models like ResNet or EfficientNet. These architectures may capture more intricate features, potentially boosting the system's overall performance.

3. MULTI-MODAL FUSION:

Integrate additional modalities, such as voice or gesture analysis, for a more comprehensive understanding of individuals. This multi-modal approach can provide a more holistic view, enhancing the accuracy and robustness of demographic predictions.

4. TRANSFER LEARNING:

Implement transfer learning techniques to leverage knowledge gained from related tasks. Transfer learning can accelerate model training and improve performance by utilizing pre-trained models on tasks with similar characteristics.

5. REAL-TIME OPTIMIZATION:

Optimize real-time processing capabilities to achieve higher frame rates, especially in resource-constrained environments. Techniques like model quantization or deploying specialized hardware can enhance the system's efficiency.

6. BIAS MITIGATION:

Develop strategies to address potential biases in the dataset and models. This includes actively seeking diverse datasets and implementing techniques to mitigate biases related to ethnicity, age groups, or gender, ensuring fair and unbiased predictions.

7. CONTINUOUS DATASET EXPANSION:

Regularly update and expand the dataset to incorporate new trends, demographics, and variations in facial expressions. This ongoing dataset expansion ensures the system remains relevant and adaptable to evolving user demographics.

8. USER FEEDBACK INTEGRATION:

Implement a feedback loop where users can provide feedback on the system's predictions. This user-centric approach can help identify and rectify potential shortcomings, contributing to continuous improvement.

9. PRIVACY-PRESERVING MEASURES:

Introduce privacy-preserving measures, such as face anonymization or encryption, to address concerns related to data privacy. This ensures that the system can be deployed ethically and in compliance with privacy regulations.

10. CLOUD INTEGRATION:

Explore the integration of cloud-based solutions to leverage the computing power of cloud platforms. This allows for scalability and facilitates real-time updates without the need for significant local hardware resources.

REFERENCES

[1]. Mohd Rafey and Gurubasava, “Age and Gender Detection using Python”, International Journal of Interdisciplinary Innovative Research &Development (IJIIRD) ISSN: 2456-236X Vol. 05 Issue 01 | 2020.

[2]. Akash. B. N, Akshay. K Kulkarni, Deekshith. A and Gowtham Gowda4, “Age and Gender Recognition using Convolution Neural Network”, IJESC, ISSN 2321 3361 Volume 10 Issue No.6.

[3]. Shubham Kumar Tiwari (1613112045), “AGE AND GENDER DETECTION”, GALGOTIAS UNIVERSITY, Project Report Of Capstone Project- 2.

[4]. Anto A Micheal and R Shankar, “Automatic Age and Gender Estimation using Deep Learning and Extreme Learning Machine”, Turkish Journal of Computer and Mathematics Education Vol.12 No.14 (2021), 63- 73.

[5]. Balavan Chauhan, Vivek Ravi Rangarej, Sandeep Hunnu Rathod and Mr. H R Ravikumar, “Age and Gender Detection using Deep Learning”, International Research Journal of Engineering and Technology (IRJET), Volume: 08 Issue: 02 Feb 2021.

[6]. Amit Dhomne, Ranjit Kumar and Vijay Bhan, “Gender Recognition Through Face Using Deep Learning”, International Conference on Computational Intelligence and Data Science (ICCIDS 2018).

[7]. Sahra Tilki, Hasibe Busra Dogru, Alaa Ali Hameed, Akhtar Jamil, Jawad Rasheed and Erdal Alimovski, “Gender Classification using Deep Learning Techniques”, Manchester Journal of Artificial Intelligence & Applied Sciences VOLUME 02, NO: 01. 2021.

[8]. Enes Bajrami and Festim Halili, “PYTHON SOFTWARE FOR GENDER DETECTION USING ARTIFICIAL INTELLIGENCE LIBRARIES TENSORFLOW, CNN, KERAS, OPENCV ON LIVE WEB CAMERA”, International Research Journal of Modernization in Engineering Technology and Science Volume:04/Issue:03/March-2022.

[9]. Gil Levi and Tal Hassner, “Age and Gender Classification using Convolutional Neural Networks”, IEEE Workshop on Analysis and Modeling of Faces and Gestures (AMFG), at the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Boston, 2015.

[10]. Sai Teja Challa, Sowjanya Jindam, Ruchitha Reddy Reddy and Kalathila Uthej, “Age and Gender Prediction using Face Recognition”, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249-8958 (Online), Volume-11 Issue-2, December 2021.

[11]. AVISHEK GARAIN, BISWARUP RAY, PAWAN KUMAR SINGH, ALI AHMADIAN, NORAZAK SENU AND RAM SARKAR, “GRA_Net: A Deep Learning Model for Classification of Age and Gender From Facial Images”, IEEE Access, Digital Object Identifier 10.1109/ACCESS .2021.3085971.

[12]. Sasikumar Gurumurthy, C. Ammu and B. Sreedevi, “Age Estimation and Gender Classification Based On Face Detection And Feature Extraction”, International Journal of Management & Information Technology, ISSN 2278-5612 Vol.4, No.1.
[13]. Shubham Patil, Bhagyashree Patil and Ganesh Tatkare, “Gender Recognition and Age Approximation using Deep Learning Techniques”, International Journal of Engineering Research & Technology (IJERT), Vol. 9 Issue 04, April-2020.
[14]. Mahija Kante, Dr. Esther Sunandha Bandaru, Gadili Manasa, Meghana Emandi and Vanarasi Leela Lavanya, “Age and Gender Detection using OpenCV”, INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY, (Volume 7, Issue 3 - V7I3-2163).
[15]. GOLOMB, Beatrice A.; LAWRENCE, David T; SEJNOWSKI, Terrence J. SEXNET: A Neural Network Identifies Sex From Human Faces. In: MPS. 1990.
[16]. MOGHADDAM, B.; MING-HSUA N YANG. Gender classification with support vector machines. In: Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580). 2000, pp. 306-311. Available from DOI: 10 .1109/AFGR. 2000.840651.
[17]. AGUSTSSON, E; TIMOFTE, R; ESCALERA, S; BARO, X; GUYON, I; ROTHE., R. Apparent and real age estimation in still images with deep residual regressors on APPA-REA L database. In: 12th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), 2017. 2017.
[18]. CLAPES, Albert; BILICI, Ozan; TEMIROVA, Dariia; AVOTS, Egils; ANBARJAFARI, Gholamreza; ESCALERA, Sergio. From apparent to real age: gender, age, ethnic, makeup, and expression bias analysis in real age estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2018, pp. 2373-2382.
[19]. ZHANG, Yunxuan; LIU, Li; LI, Cheng; LOY, Chen Change. Quantifying Facial Age by Posterior of Age Comparisons. In: British Machine Vision Conference (BMVC). 2017.
[20]. SIMONYAN, Karen; ZISSERMAN, Andrew. Very Deep Convolutional Networks for Large-Scale Image Recognition. 2014. Available from eprint: arXiv: 1409.1556.
[21]. HEURITECH. Atomic force microscopy. 2013. Available also from: https: //blo g. heuritech. com/2016/02/29/a-brief- report - of-the-heuritech-deep-learning-meetup-5/. [Online; accessed May 15, 2019].
[22]. SZEGEDY, Christian; IOFFE, Sergey; VANHOUCKE, Vincent. Inceptionv4, Inception-ResNet and the Impact of Residual Connections on Learning. CoRR. 2016, vol. abs/1602.07261. Available from arXiv: 1602.07261.

[23]. SZEGEDY, Christian; LIU, Wei; JIA, Yangqing; SERMANET, Pierre; REED, Scott; ANGUELOV, Dragomir; ERHAN, Dumitru; VAN - HOUCKE, Vincent; RABINOVICH, Andrew. Going Deeper with Convolutions. 2014. Available from eprint: arXiv: 1409.4842.
[24]. HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing; SUN, Jian. Deep Residual Learning for Image Recognition. 2015. Available from eprint: arXiv:1512.03385.
[25]. WOLF, Lior; HASSNER, Tal; MAOZ, Itay. Face recognition in unconstrained videos with matched background similarity. In: in Proc. IEEE Conf. Comput. Vision Pattern Recognition. 2011.
[26]. SZEGEDY, Christian; VANHOUCKE, Vincent; IOFFE, Sergey; SHLENS, Jonathon; WOJNA, Zbigniew. Rethinking the Inception Architecture for Computer Vision. CoRR. 2015, vol. abs/1512.00567. Available from arXiv: 1512.00567.
[27]. KINGMA, Diederik P.; BA, Jimmy. Adam: A Method for Stochastic Optimization. 2014. Available from eprint: arXiv: 1412.6980.
[28]. NEWELL, Alejandro; YANG , Kaiyu; DENG , Jia. Stacked Hourglass Networks for Human Pose Estimation. Computing Research Repository (CoRR). 2016, vol. abs/1603.06937. Available from arXiv: 1603. 06937.
[29]. YOUN G H O KWON; D A VITORIA LOBO. Age classification from facial images. In: 2994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 1994, pp. 762-767. ISSN 1063-6919. Available from DOI: 10.1109/CVPR. 1994.323894.
[30]. ROTHE, Rasmus; TIMOFTE, Radu; GOOL, Luc Van. DEX: Deep Expectation of apparent age from a single image. In: IEEE International Conference on Computer Vision Workshops (ICCVW). 2015.