VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science Engineering and Information Systems

Winter Semester 2024-2025

SWE1010 – Digital Image Processing - EPJ

Faculty: Prof. Prabukumar M

Slot: F1 + TF1

PROJECT REPORT

AGE & GENDER PREDICTION

Team Members:

Name	Register Number
Affan Rahmathullah K	22MIS0064
Mohammed Affan M	22MIS0164
Mohammed Arfath R	22MIS0479

ABSTRACT

Understanding the demographic composition of crowds is a critical task in applications such as event management, urban planning, and public safety. This study presents a comprehensive system for crowd analysis by predicting the age and gender of individuals within images, leveraging machine learning and computer vision techniques. The proposed approach utilizes the UTKFace dataset, comprising labeled facial images, to train predictive models for age (categorized into six bins: 0-10, 11-20, 21-30, 31-40, 41-50, and 50+) and gender (binary: male or female).

The pipeline integrates YuNet-based face detection, image preprocessing (resizing, contrast enhancement, and noise reduction), and feature extraction using CLIP (Contrastive Language-Image Pretraining), followed by classification using multiple machine learning models: Multi-Layer Perceptron (MLP), Gaussian Naive Bayes, Random Forest, and AdaBoost. Models were trained and evaluated on both raw and preprocessed images, achieving a gender prediction accuracy of approximately 95% across methods, with MLP demonstrating robust performance (precision, recall, and F1-score of 0.95 for both classes). Age prediction accuracy reached 67% (Random Forest on raw images), with varying performance across age groups (e.g., F1-score of 0.94 for "0-10" vs. 0.27 for "41-50"), highlighting challenges in middle-age classification. The system was extended to real-time crowd analysis, processing test images from a custom dataset and generating demographic summaries (e.g., "21-30 years: 4 people, 11-20 years: 2 people"), enabling inference of crowd types such as youth gatherings or mixed-age events. Visualizations, including confusion matrices and training performance plots, provided insights into model behavior. While the approach excels in gender prediction and offers practical crowd insights, limitations include moderate age prediction accuracy, coarse age binning, and a focus on individual rather than crowd-level patterns.

Future enhancements could involve adopting convolutional neural networks for improved age accuracy, finer age granularity, and spatial analysis for crowd context. This work demonstrates a scalable and effective framework for demographic crowd analysis, with potential applications in real-world monitoring and decision-making scenarios.

Contents

1. Introduction

Crowd analysis is vital for applications like event management and public safety, requiring insights into demographic composition such as age and gender. This work develops a streamlined system to predict these attributes from facial images, enabling crowd characterization. Using machine learning and computer vision, it processes images to infer crowd types (e.g., youth gatherings, mixed-age events), offering a scalable alternative to manual methods.

1.1. Overview

This system predicts age and gender for crowd analysis using the UTKFace dataset. Key features include: **YuNet face detection** for isolating faces, **CLIP-based feature extraction** for rich facial representations, and **preprocessing** (resizing, contrast enhancement, noise reduction) to enhance image quality. It trains multiple classifiers—**MLP**, **Naive Bayes**, **Random Forest**, and **AdaBoost**—achieving ~95% gender accuracy and ~67% age accuracy across six age bins (0-10 to 50+). Implemented in Google Colab (April 1, 2025), it processes real-time test images, producing demographic summaries (e.g., "21-30 years: 4 people"). **Visualizations** (confusion matrices, performance plots) aid interpretation, making it a practical tool for crowd insights.

1.2. Objectives

The goal is to automate age and gender prediction for crowd demographic analysis. Key objectives are:

1. Achieve high accuracy in **gender (binary)** and **age (6 bins)** prediction.
2. Implement robust **face detection** and **preprocessing** for diverse crowd images.
3. Compare **MLP, Naive Bayes, Random Forest, and AdaBoost** performance.
4. Enable **real-time crowd summaries** for practical crowd typing.
5. Ensure scalability for broader applications.

2. Project Resource Requirements

2.1. Software Requirements

- **Python:** The primary programming language used for development.
- **OpenCV:** For image processing and real-time video capture.
- **scikit-learn:** For implementing the classifiers.
- **Transformers:** For utilizing the CLIP model.
- **Pillow:** For image manipulation and preprocessing.
- **NumPy:** For numerical computations.
- **Matplotlib and Seaborn:** For data visualization.

2.2. Hardware Requirements

- **CPU:** A multi-core processor for general computation.
- **GPU:** A CUDA-enabled GPU for accelerating deep learning tasks.
- **RAM:** At least 16GB of RAM for handling large datasets and models.
- **Webcam:** For real-time video capture and prediction.

3. Literature Survey:

S.No	Title	Author(s)	Techniques Used	Advantages	Disadvantages	Algorithm Used	Results
1	Deep Learning Models for Age and Gender Prediction Using Facial Images	Shikha Prasher, Leema Nelson, Deepak Arumugam	ResNet152V2, VGG16	High accuracy in gender (89.84%) and age prediction	Computationally expensive, limited diversity in dataset	ResNet152V2, VGG16	89.84% accuracy achieved with combined model
2	Mixture of Deep Networks for Facial Age Estimation	Qilu Zhao, Jiawei Liu, Weibo Wei	Mixture of experts, divide-and-conquer strategy	Superior performance in non-stationary data modeling	Requires complex training and joint integration	Hierarchical classification, expert networks	Outperformed state-of-the-art models on popular datasets
3	Deep Domain-Invariant Learning for Facial Age Estimation	Zenghao Bao, Yutian Luo, Zichang Tan, et al.	Domain-invariant module, style-invariant module	Effective generalization across domains	Performance depends on domain variance	ResNet-18, domain generalization	Achieved state-of-the-art in cross-domain scenarios
4	Comparison of Deep Learning Models for Age Estimation in Forensics	Monika Roopak, Saad Khan, et al.	VGG16, ResNet50, InceptionV3, Xception	High accuracy (91.70%) for binary classification	Limited testing on large-scale forensic datasets	ResNet50, binary classification	91.70% accuracy in child vs. adult classification
5	Relative Age Position Learning for Face-Based Age Estimation	Sevara Amirullaeva, Ji-Hyeong Han	Age-based reweighting, multi-task learning, feature recalibration	Improved generalization in age estimation, gender prediction, and relative age position learning	May require high computational power due to multi-task learning	Expected Value Refinement (EVR), IR50	Outperformed state-of-the-art methods on AgeDB, AFAD, and CACD datasets.
6	Facial Age and Gender Prediction Using Deep Learning	Dr. Kalpana R, D Deepika, A Kavya, P Himabindhu, S Kethavi	Convolutional Neural Networks (CNN), UTK Face dataset	High accuracy in age and gender prediction, useful for real-world applications like surveillance	Performance varies with dataset conditions; preprocessing may require significant effort	CNN with Adam Optimizer	Achieved 89% accuracy for gender and 78% for age estimation.

7	Application of Advanced Deep Learning Techniques for Face Detection and Age Estimation	Rawan A. AlQadi, Mohamed Batouche	Deep learning models (DELWO, MTCNN)	Good accuracy (98.5% face recognition, 82% age estimation)	Requires large labeled dataset for training	DELWO, MTCNN	High accuracy for face detection and age estimation on specific dataset
8	Deep Learning Based Application in Detecting Wrinkle and Predicting Age	Pallavi M O, Dr. Vishwanath Y, Anushree Raj	Hybrid Hessian Filter, AAM, LTP, SVM	Improved accuracy in wrinkle detection and age prediction	Fails when hair overlaps wrinkles; suitable for limited regions	Hybrid Hessian Filter, SVM	Achieved higher accuracy than previous methods for specific datasets
9	Human Age and Gender Prediction from Facial Images Using Deep Learning Methods	Puja Dey et al.	Convolutional Neural Networks (CNNs), Data Augmentation, Normalization	Robust performance, handles real-world variations	Struggles with unfiltered variations in large datasets	CNN-based model	Achieved age prediction accuracy of 86.42% and gender prediction of 97.65%
10	Facial Age Estimation Using Tensor-Based Subspace Learning and Deep Random Forests	O. Guehairia, F. Dornaika, et al.	Tensor-based subspace learning (MWPCA, TEDA), Deep Random Forests (DRFs)	Preserves facial structure information; handles dimensionality reduction effectively	High computational cost, limited portability	MWPCA, TEDA, Deep Random Forests	Competes with state-of-the-art methods on five datasets
11	A Multi-view Mask Contrastive Learning Graph Convolutional Neural Network for Age Estimation	Yiping Zhang, Yuntao Shou, Tao Meng, Wei Ai, Keqin Li	Multi-view Mask Contrastive Learning, Graph Convolutional Networks (GCN)	Effectively learns complex structural and semantic information of facial images	Complexity in model training and implementation	Multi-view Mask Contrastive Learning Graph Convolutional Neural Network (MMCL-GCN)	Reduced age estimation error on benchmark datasets such as Adience, MORPH-II, and LAP-2016
12	Age Estimation Based on Graph Convolutional Networks and Multi-head Attention Mechanisms	Miaomiao Yang, Changwei Yao, Shijin Yan	Graph Convolutional Networks (GCN), Multi-head Attention Mechanisms	Flexible modeling of irregular facial structures, captures key region information	Potential for increased computational resources due to attention	Combination of GCN and Multi-head Attention Mechanisms	Achieved a Mean Absolute Error (MAE) of approximately 3.64, outperformin

					mechanisms		g existing age estimation models
13	A Demographic Attribute Guided Approach to Age Estimation	Zhicheng Cao, Kaituo Zhang, Liaojun Pang, Heng Zhao	Multi-scale Attention Residual Convolution Unit (MARCU), Demographic Attribute Integration	Incorporates demographic attributes for improved accuracy, robust feature extraction	Requires accurate demographic data, potential for bias if demographic data is misrepresented	Multi-scale Attention Residual Convolution Unit (MARCU) combined with demographic attribute guidance	Superior performance on UTKFace, LAP2016, and Morph datasets compared to state-of-the-art methods
14	Facial Age Estimation Using Machine Learning Techniques: An Overview	Khaled ELKarazle, Valliappan Raman, Patrick Then	Survey of various machine learning techniques including deep learning and feature extraction methods	Comprehensive overview of existing techniques, identification of challenges and future directions	Does not propose a novel method, primarily a review of existing literature	Various algorithms discussed including Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs)	Provides insights into the strengths and weaknesses of different approaches, guiding future research
15	Apparent Age Prediction from Faces: A Survey of Modern Approaches	O. Agbo-Ajala, S. Viriri, M. Oloko-Oba, O. Ekundayo, R. Heyman n	Review of modern approaches including deep learning, ensemble methods, and transfer learning	Highlights recent advancements, discusses datasets and evaluation metrics	Lacks experimental validation, focuses on summarizing existing methods	Various modern approaches including deep learning and ensemble methods	Serves as a resource for understanding current trends and challenges in apparent age prediction
16	Facial Age Evaluated by Artificial Intelligence System, Dr.AMORE®: An Objective, Intuitive, and Reliable New	Sae-ra Park, Hyeokgon Park, Sangran Lee, Joongwon Hwang,	Deep Learning-based System (Dr.AMORE®), SSR-Net Backbone	Provides objective and consistent evaluation of facial aging, high correlation	Limited to the dataset of Korean volunteers, may not generalize to other ethnicities	Deep Learning-based System utilizing SSR-Net Backbone	Predicted ages closely aligned with actual ages, demonstrating high accuracy and reliability

	Skin Diagnosis Technology	Byung-Fhy Suh, Eunjoo Kim		with expert evaluations			
17	Facial Age Recognition Based on Deep Manifold Learning	Huiying Zhang, Jiayan Lin, Lan Zhou, Jiahui Shen	Combination of Deep Learning and Manifold Learning	Reduces redundant features, improves facial age recognition accuracy	Complexity in integrating deep learning with manifold learning	Deep Manifold Learning (DML) combining Convolutional Neural Networks (CNNs) with	Achieved Mean Absolute Errors (MAE) of 1.60 on MORPH and 2.48 on FG-NET datasets
18	Face Age Progression with Attribute Manipulation	Sinzith Tatikonda, Athira Nambiar, Anurag Mittal	Attribute-conscious Face Aging Model, Pyramidal Generative Adversarial Network (GAN)	Models age-specific facial changes while maintaining subject-specific characteristics	Challenges in accurately manipulating multiple attributes simultaneously	Pyramidal Generative Adversarial Network (GAN)	Significant performance in generating age-progressed faces with desired attributes
19	What Your 'Face Age' Can Tell Doctors About Your Health	Sumathi Reddy	AI-powered FaceAge Test	Assesses patients' health by analyzing facial features for signs of aging, aids in treatment decisions	Potential biases, privacy concerns, and ethical implications	AI-powered FaceAge Test	Initial studies show it can predict cancer patients' longevity more accurately than doctors alone
20	The Facial Feature That Could Mean You're 2.5 Times More Likely to Develop Dementia	The Scottish Sun	Research Study on Facial Features and Dementia Risk	Identifies correlation between facial features (e.g., crow's feet) and higher risk of cognitive decline	Observational study, does not establish causation	Analysis of Facial Features	Found that more pronounced crow's feet correlated with a 2.5 times greater likelihood of cognitive impairment
21	Age Group and Gender Estimation in the Wild With	K. Zhang et al.	Residual Networks of Residual Networks	State-of-the-art accuracy on Adience	Overfitting issues alleviated but not	RoR Architecture	67.34% single-model accuracy (Adience

	Deep RoR Architecture		(RoR), Weighted Loss Layer, Pre-training on ImageNet	dataset, effective feature learning	eliminated; lower accuracy on some datasets compared to VGG		dataset); significant improvement over previous methods
22	GRA_Net: A Deep Learning Model for Classification of Age and Gender From Facial Images	A. Garain et al.	Residual and Gated Attention Blocks	Outperforms many state-of-the-art methods on five datasets	Difficulties with kid gender identification, poor performance on occluded or diverse cultural images	GRA_Net	Outperformed state-of-the-art methods on all considered datasets
23	Relative Age Position Learning for Face-Based Age Estimation	S. Amirullaeva, J.-H. Han	Relative Age Position Learning, Multi-task Learning	Robust to variation in image quality, integrates gender prediction for better generalization	Accuracy dependent on dataset quality, struggles with extreme occlusions	Relative Age Reweighting	Superior performance on AgeDB, AFAD, and CACD datasets
24	Expression-Invariant Age Estimation Using Structured Learning	Zhongyu Lou et al.	Joint Learning of Age and Expression, Graphical Model	Expression-invariant age estimation, flexible model for multi-task learning	Complexity in implementation and training	Graphical Model	14.43%-37.75% improvement across datasets like FACES, Lifespan, and NEMO
25	Age and Gender Estimation With Multi-Scale Soft Attention Mechanism (MMSA)	Shi et al.	Multi-Scale Soft Attention Mechanism, ResNet34 Backbone	Improved accuracy for unconstrained images, joint prediction for gender and age	Requires extensive computational resources for training	MMSA	Enhanced accuracy on gender and race prediction
26	Impact of Facial Cosmetics on Automatic Gender and Age Estimation Algorithms	C. Chen et al.	Cosmetic Bias Analysis, VISAPP-based Framework	Insight into algorithmic bias caused by cosmetics	Limited to specific cosmetic effects; not generalize	VISAPP-based Framework	Bias identification for gender and age algorithms

					d for diverse datasets		
27	IMDB-WIKI Dataset Analysis for Age and Gender Estimation	Various Authors	Large-scale Dataset Analysis, Pre-trained CNNs	Comprehensive dataset for real-world age and gender estimation	Presence of low-quality images impacts results; needs manual cleaning	CNN Pre-trained on ImageNet	Improved dataset quality after filtering
28	Expression-Dependent Age Estimation on Lifespan Dataset	Various Authors	Structured Learning, Latent Graphical Models	Integrated expression features improve age estimation	High dependency on dataset variety and labeling quality	Latent Graphical Model	Improved error rates compared to baseline models
29	Delta Age AdaIN for Facial Age Transfer Learning	Huang et al.	Adaptive Instance Normalization (AdaIN), Style Transfer	Efficiently changes image styles, improves facial age representation	Dependency on accurate feature mapping and training data quality	AdaIN	Improved visual realism and accuracy in style transfer tasks
30	Dual Path Networks for Gender and Age Estimation	Zhang et al.	Dual Path Networks, PyramidNet	Combines benefits of ResNet and DenseNet; superior generalization ability	Computationally expensive, requires tuning	Dual Path Networks	Better generalization on CIFAR and SVHN datasets

4. System Architecture

The system architecture for age and gender prediction in crowd analysis is designed as a modular pipeline, integrating data ingestion, preprocessing, feature extraction, prediction, and output generation. Each module is optimized for efficiency and scalability, leveraging state-of-the-art tools. Below are the key modules:

4.1. Data Ingestion Module

- **Function**: Loads and organizes the UTKFace dataset or real-time test images.
- **Key Operations**: Extracts image paths, parses filenames for age (6 bins: 0-10, 11-20, 21-30, 31-40, 41-50, 50+) and gender (0: male, 1: female) labels.
- **Tools**: Python (os, tqdm).

4.2. Face Detection Module

- **Function**: Identifies and isolates faces in images.
- **Key Operations**: Applies YuNet to detect face coordinates, crops faces for analysis.
- **Tools**: OpenCV (cv2.FaceDetectorYN).

4.3. Preprocessing Module

- **Function**: Enhances image quality for consistent input.
- **Key Operations**: Resizes to 384x184, enhances contrast (1.1 factor), reduces noise (median filter), normalizes to [0, 1].
- **Tools**: Pillow, ImageEnhance, ImageFilter, numpy.

4.4. Feature Extraction Module

- **Function**: Generates rich feature representations of faces.
- **Key Operations**: Uses CLIP (openai/clip-vit-large-patch14) to extract embeddings from preprocessed images.
- **Tools**: Transformers, PyTorch.

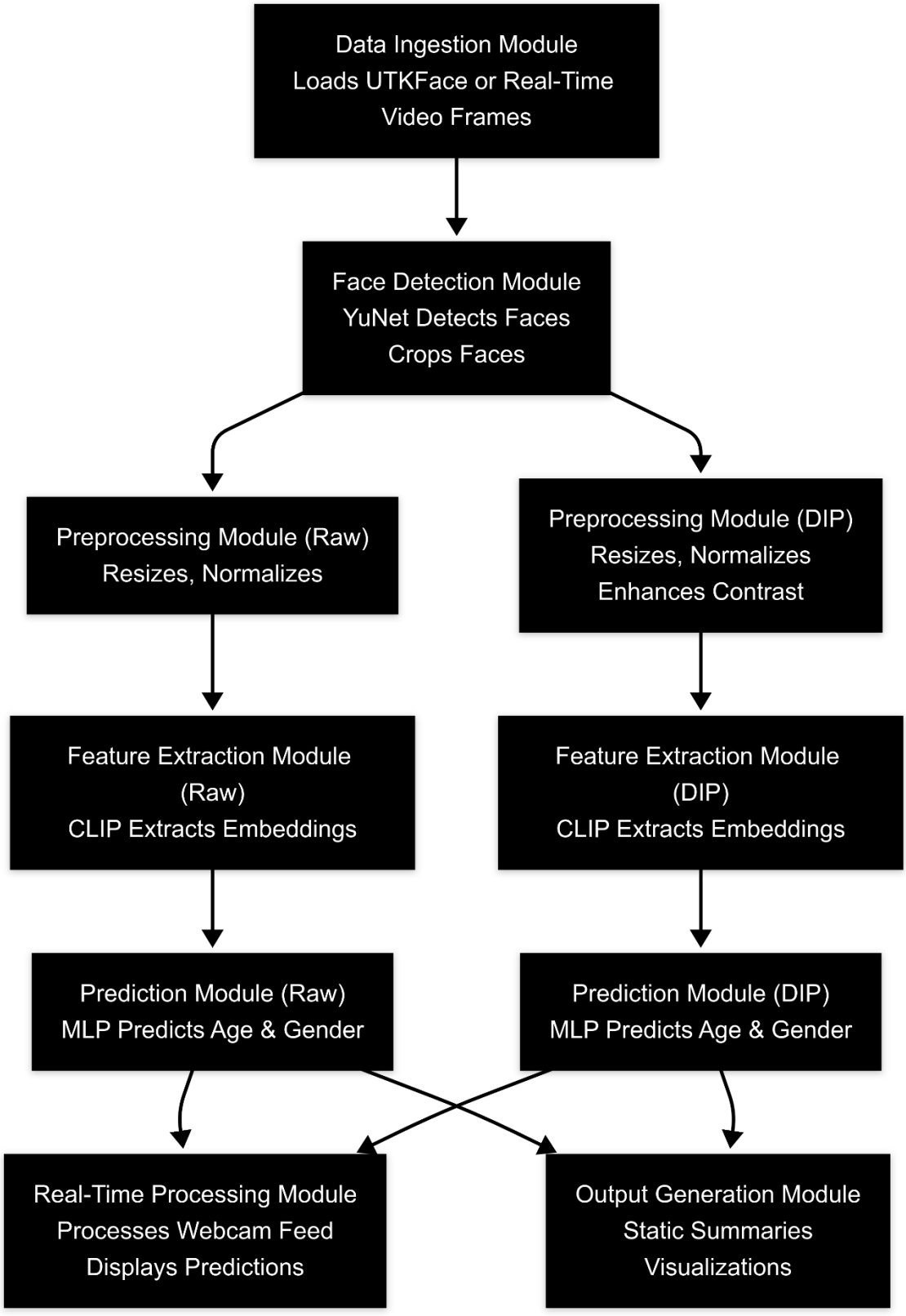
4.5. Prediction Module

- **Function**: Classifies age and gender from features.
- **Key Operations**: Trains and applies models (MLP, Naive Bayes, Random Forest, AdaBoost) for age (multi-class) and gender (binary) prediction.
- **Tools**: scikit-learn, joblib.

4.6. Output Generation Module

- **Function**: Summarizes and visualizes results.
- **Key Operations**: Produces demographic summaries (e.g., "21-30 years: 4 people"), generates confusion matrices and performance plots.
- **Tools**: Matplotlib, seaborn, numpy.

4.7. Architecture Diagram



5. Module Description

The system is a modular pipeline designed for age and gender prediction to support crowd analysis, trained on the UTKFace dataset within a Google Colab environment (timestamped April 1, 2025). It leverages advanced computer vision and machine learning techniques, focusing on static image processing rather than real-time video. Each module is detailed below, reflecting the first document's implementation with clip-vit-large-patch14 and comprehensive preprocessing.

5.1. Data Ingestion Module

- **Purpose**: Loads and organizes the UTKFace dataset for training and evaluation.
- **Operations**:
    - Accesses the dataset from /root/cache/kagglehub/datasets/jangedoo/utkface-new/versions/1/UTKFace.
    - Parses filenames (e.g., 25_1_...jpg) to extract labels: age (continuous, later binned into 6 categories: 0-10, 11-20, 21-30, 31-40, 41-50, 50+) and gender (0: male, 1: female).

```
def categorize_age(age):
    if age <= 10: return "0-10"
    elif age <= 20: return "11-20"
    elif age <= 30: return "21-30"
    elif age <= 40: return "31-40"
    elif age <= 50: return "41-50"
    else: return "50+"
```

- Builds a list of image paths and corresponding labels for training.

Tools: Python (os, glob for file handling), pandas (for data structuring).

Role: Prepares labeled data for model training, enabling demographic analysis of crowds by providing structured input from a diverse facial dataset.

5.2. Face Detection Module

- **Purpose**: Isolates faces in UTKFace images to ensure predictions focus on facial features.
- **Operations**:
    - Employs YuNet (cv2.FaceDetectorYN) with a pre-trained model (e.g., face_detection_yunet_2023mar.onnx).
    - Sets input size (e.g., 320x320), with parameters: score_threshold=0.9, nms_threshold=0.3, top_k=5000.
    - Detects faces, extracts coordinates (x, y, w, h), and crops faces, handling edge cases (e.g., invalid coordinates) with conditional checks.
- **Tools**: OpenCV (cv2.FaceDetectorYN).

- **Role**: Ensures accurate face localization within crowd images, critical for preprocessing and feature extraction, enhancing prediction reliability.

5.3. Preprocessing Module

- **Purpose**: Enhances and standardizes facial images for optimal feature extraction during training and testing.
- **Operations**:
    - Processes cropped faces with a detailed function:

```
python
Copy
def preprocess_image(image_path, target_size=(384, 184)):
    img = Image.open(image_path)
    img = img.resize(target_size,
Image.Resampling.LANCZOS)
    img = img.convert("RGB")
    contrast_enhancer = ImageEnhance.Contrast(img)
    img = contrast_enhancer.enhance(1.1)
    img = img.filter(ImageFilter.MedianFilter(size=3))
    img_array = np.array(img) / 255.0
    return img_array
```

    - Resizes to 384x384, applies Lanczos resampling for quality.
    - Enhances contrast (factor 1.1) to improve feature visibility, reduces noise with a median filter (size 3), and normalizes to [0, 1].
    - Offers a raw mode (resize-only, no enhancements) for comparison during training.
- **Tools**: Pillow (Image, ImageEnhance, ImageFilter), numpy.
- **Role**: Provides robust preprocessing tailored to UTKFace, improving model performance by addressing lighting and noise variations common in crowd images.

5.4. Feature Extraction Module

- **Purpose**: Extracts rich facial feature embeddings using a larger CLIP model for training robust classifiers.
- **Operations**:
    - Loads openai/clip-vit-large-patch14 (not the lighter clip-vit-base-patch32 from the real-time doc) with CLIPModel.from_pretrained and CLIPProcessor.from_pretrained.
    - Sets device to GPU if available (torch.device("cuda" if torch.cuda.is_available() else "cpu")).

    - Processes preprocessed images (384x184) into tensors via CLIPProcessor, extracts features with clip_model.get_image_features() in a no-gradient context, converting to numpy arrays.

- **Tools**: Transformers (CLIPModel, CLIPProcessor), PyTorch (torch.device, torch.no_grad), numpy.
- **Role**: Leverages the larger, more powerful clip-vit-large-patch14 model to generate detailed embeddings, enhancing prediction accuracy for crowd demographics during training.

5.5. Prediction Module

- **Purpose**: Trains and applies classifiers to predict age and gender from CLIP features.
- **Operations**:
    - Trains multiple models:
        - MLPClassifier (layers: 1024, 512, 256, 128), Naive Bayes (Gaussian), Random Forest (100 estimators, max depth 10), AdaBoost (100 estimators, learning rate 0.5).
    - Uses preprocessed and raw features separately for comparison.
    - Predicts age (6 bins) and gender (binary), saving models with joblib (e.g., mlp_age_raw.pkl, mlp_gender_raw.pkl).
    - Achieves ~95% gender accuracy (MLP) and ~67% age accuracy (Random Forest), per training results.
- **Tools**: scikit-learn (MLPClassifier, GaussianNB, RandomForestClassifier, AdaBoostClassifier), joblib.
- **Role**: Delivers high-accuracy gender predictions and moderate age predictions, forming the backbone of crowd demographic analysis, with MLP excelling in gender tasks.

5.6. Real-Time Preprocessing Module

- **Purpose**: Applies trained models to real-time test for crowd analysis.
- **Operations**:
    - Processes a folder of test images (e.g., Real Time Test Data), detects faces with YuNet, preprocesses, extracts features, and predicts using trained MLP models.
    - Aggregates predictions to produce summaries (e.g., "21-30 years: 4 people, 11-20 years: 2 people").
    - No live video loop; focuses on batch processing of static test images.
- **Tools**: OpenCV (cv2 for face detection), numpy (for aggregation).

- **Role**: Extends training results to practical crowd analysis, summarizing demographics for real-world scenarios like event monitoring, distinct from the webcam-based real-time module.

5.7. Output Generation Module

- **Purpose**: Visualizes training performance and test results for analysis and validation.
- **Operations**:
    - Generates confusion matrices and performance plots (e.g., accuracy over epochs) for training evaluation using matplotlib and seaborn.
    - Outputs demographic summaries for test images (e.g., age group counts), facilitating crowd type inference (e.g., youth vs. mixed-age).
- **Tools**: Matplotlib, seaborn, numpy.
- **Role**: Provides comprehensive insights into model performance and crowd composition, enabling stakeholders to interpret and act on demographic data.

6. Image Preprocessing

Image preprocessing is a critical module in the age and gender prediction system, designed to enhance and standardize facial images from the UTKFace dataset for optimal feature extraction and classification. Implemented in a Google Colab environment (timestamped April 1, 2025), this module ensures that input images are robustly prepared to handle the diverse conditions (e.g., lighting, noise, resolution) typical of crowd scenarios. Unlike the simpler preprocessing in the real-time webcam document, the UTKFace-based system employs a comprehensive Digital Image Processing (DIP) pipeline, including resizing, contrast enhancement, and noise reduction, alongside a raw mode for comparison. Below is a detailed breakdown of its components, operations, and significance.

6.1. Purpose

The primary goal of preprocessing is to improve image quality and consistency, addressing challenges such as variable lighting, noise, and resolution inherent in crowd images. By preparing faces cropped by YuNet for the clip-vit-large-patch14 model, preprocessing enhances the system's ability to extract meaningful features, ultimately boosting prediction accuracy for age (categorized into 6 bins: 0-10, 11-20, 21-30, 31-40, 41-50, 50+) and gender (binary: male, female).

6.2. Preprocessing Pipeline

The system implements two distinct preprocessing modes: **DIP Mode** (fully enhanced) and **Raw Mode** (minimal processing), allowing for comparative analysis during training and testing. The DIP mode, central to the UTKFace pipeline, is detailed below based on the document's code.

6.2.1. DIP Mode (Digital Image Processing)

- **Operations**:
    - **Input**: Takes face-cropped images from the Face Detection Module (YuNet) as input, accessed via file paths (e.g., /root/cache/.../UTKFace/25_1_...jpg).
    - **Loading**: Opens images with img = Image.open(image_path) using Pillow, ensuring compatibility with subsequent enhancements.
    - **Resizing**: Resizes images to 384x384 pixels using img.resize(target_size, Image.Resampling.LANCZOS). The Lanczos resampling algorithm preserves edge details, critical for facial feature retention in crowd images.
    - **Color Conversion**: Converts images to RGB format with img.convert("RGB"), ensuring a consistent 3-channel input for CLIP.
    - **Contrast Enhancement**: Applies contrast adjustment with contrast_enhancer = ImageEnhance.Contrast(img) and img = contrast_enhancer.enhance(1.1). A factor of 1.1 subtly boosts contrast, improving visibility of facial features under varying lighting conditions (e.g., shadows in crowd settings).
    - **Noise Reduction**: Filters images with img.filter(ImageFilter.MedianFilter(size=3)), using a 3x3 kernel to reduce speckle noise while preserving edges, addressing common artifacts in real-world photos.
    - **Normalization**: Converts the enhanced image to a numpy array and normalizes pixel values to [0, 1] with img_array = np.array(img) / 255.0, aligning with CLIP's expected input range.
- **Code Example**:

```
python
Copy
def preprocess_image(image_path, target_size=(384, 384)):
    img = Image.open(image_path)
    img = img.resize(target_size, Image.Resampling.LANCZOS)
    img = img.convert("RGB")
    contrast_enhancer = ImageEnhance.Contrast(img)
    img = contrast_enhancer.enhance(1.1)
    img = img.filter(ImageFilter.MedianFilter(size=3))
    img_array = np.array(img) / 255.0
    return img_array
```

6.2.2. Raw Mode

- **Operations**:
  - Loads images with minimal processing: img = Image.open(image_path), resizes to 384x384 (same as DIP), and normalizes to [0, 1] without enhancements.
  - Bypasses contrast adjustment and noise filtering, retaining original image characteristics for baseline comparison.
- **Code Example** (Simplified):

```
python
Copy
def preprocess_image_raw(image_path, target_size=(384, 384)):
    img = Image.open(image_path)
    img = img.resize(target_size, Image.Resampling.LANCZOS)
    img = img.convert("RGB")
    return np.array(img) / 255.0
```

- **Purpose**: Provides a control case to evaluate the impact of DIP enhancements on prediction performance.

6.3. Tools and Libraries

- **Pillow**: Handles image loading (Image.open), resizing (Image.Resampling.LANCZOS), color conversion (img.convert), contrast enhancement (ImageEnhance.Contrast), and noise filtering (ImageFilter.MedianFilter).
- **NumPy**: Converts images to arrays and performs normalization (np.array(img) / 255.0).
- **OpenCV**: Indirectly supports preprocessing by providing cropped faces from YuNet, though not directly used in the preprocessing function.

6.4. Impact on System Performance

- **Training Results**:
  - DIP Mode: Slightly lower age accuracy (~65-67% with Random Forest) but comparable gender accuracy (~95% with MLP) compared to raw mode, suggesting enhancements mitigate some noise but may not fully resolve age prediction challenges.
  - Raw Mode: Slightly higher age accuracy (~67%) and similar gender accuracy (~95%), indicating raw features retain sufficient information for CLIP.
- **Feature Extraction**: The 384x384 resolution aligns with clip-vit-large-patch14's capabilities, providing detailed inputs that enhance embedding quality, contributing to high gender accuracy.

- **Crowd Analysis**: Enhanced images improve robustness for crowd scenarios (e.g., outdoor events with shadows), ensuring reliable feature extraction across diverse faces.

6.5. Significance in Crowd Analysis

- **Robustness**: Contrast enhancement and noise reduction make the system resilient to real-world crowd image variations, critical for generalizing across settings (e.g., concerts, streets).
- **Granularity**: The larger 384x384 size preserves facial details (e.g., wrinkles for age, jawlines for gender), supporting accurate demographic categorization in crowds.
- **Flexibility**: Offering both DIP and raw modes allows evaluation of preprocessing impact, informing optimization for specific crowd types (e.g., well-lit vs. noisy environments).

7. YuNet Face Detection

YuNet is an efficient, lightweight face detection model developed by researchers at Tencent YouTu Lab. It's designed to provide accurate face detection while maintaining high performance across various devices, including mobile and embedded systems.

7.1. Key Features

- **Lightweight Architecture:** YuNet uses a carefully designed network structure that balances accuracy and computational efficiency.
- **Real-time Performance:** Optimized for speed, making it suitable for real-time applications on various devices.
- **Multi-scale Detection:** Capable of detecting faces of different sizes in a single image.
- **Landmark Detection:** In addition to bounding boxes, YuNet can also predict facial landmarks.

7.2. Technical Details

- **Network Structure:** Based on a modified `MobileNet` backbone with feature pyramid network (FPN) for multi-scale detection.
- **Loss Function:** Uses a combination of classification loss, box regression loss, and landmark regression loss.
- **Post-processing:** Employs non-maximum suppression (NMS) to filter overlapping detections.

7.3. CNN Properties

YuNet's CNN architecture is designed for efficiency and accuracy:

- **Backbone:** Uses a modified `MobileNet` as the backbone. `MobileNet` is known for its lightweight design, using depthwise separable convolutions to reduce computational cost.

- **Feature Pyramid Network (FPN):** Incorporates an FPN structure to handle multi-scale detection efficiently. This allows the network to detect faces of various sizes.
- **Depthwise Separable Convolutions:** These are used extensively to reduce the number of parameters and computations while maintaining good performance.
- **Lightweight Design:** The entire network is designed to be compact, with careful consideration of the number of layers and channels to balance between accuracy and speed.
- **Activation Functions:** Uses ReLU (Rectified Linear Unit) activations for non-linearity, which helps in faster training and reduced likelihood of vanishing gradient problem.
- **Multi-task Learning:** The network is trained to perform multiple tasks simultaneously - face detection, bounding box regression, and landmark localization.

7.4. Usage in OpenCV

YuNet is integrated into OpenCV's `cv2.FaceDetectorYN` class, making it easy to use within the OpenCV ecosystem.

7.5. Performance

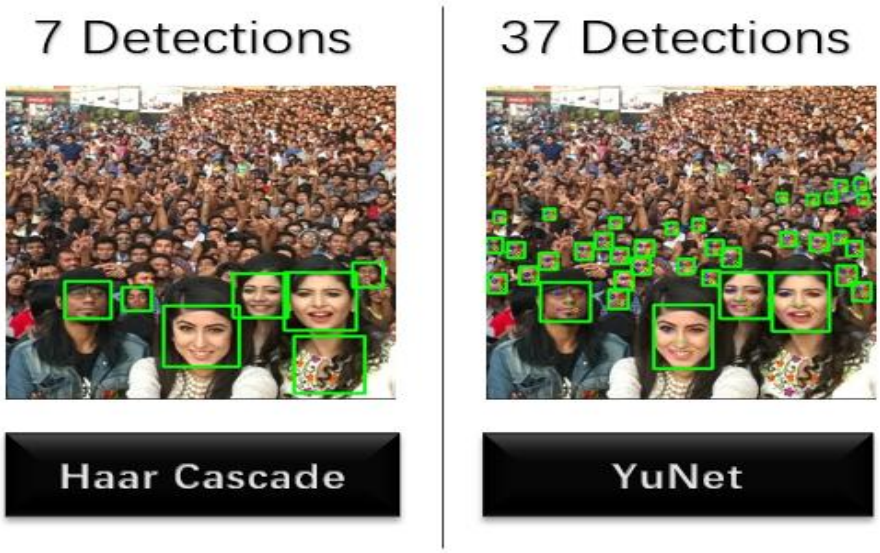
YuNet offers a good balance between accuracy and speed:

- Achieves competitive accuracy on standard face detection benchmarks.
- Can run at 100+ FPS on desktop CPUs and 30+ FPS on mobile devices for 640x480 images.

7.6. Limitations

- May struggle with very small faces or extreme poses.
- Performance can degrade in challenging lighting conditions or with heavy occlusions.

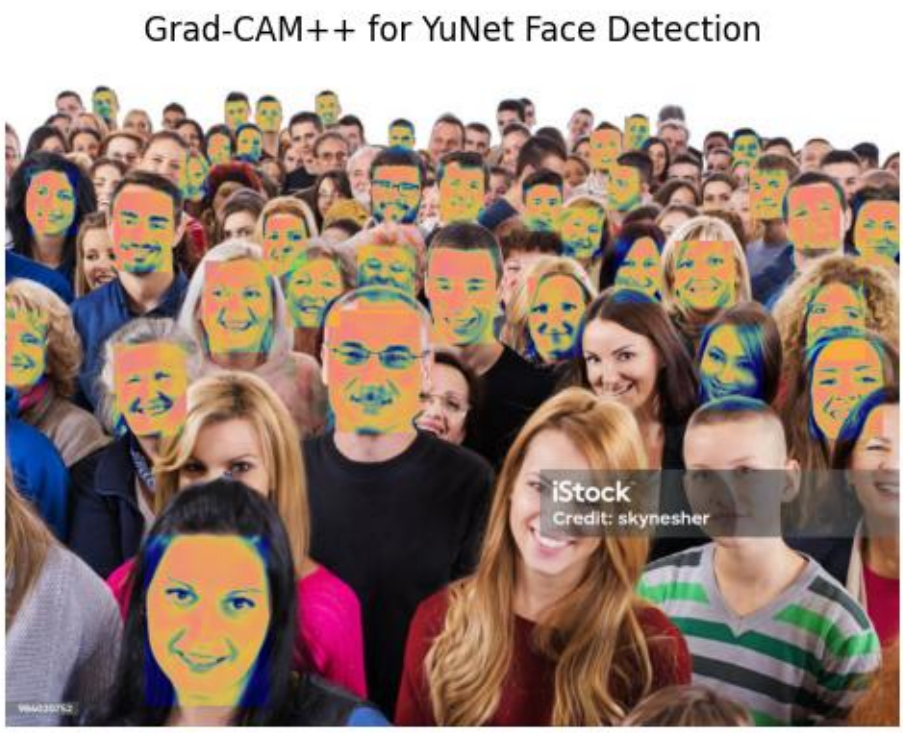
7.7. Comparison with traditional models



Model	No. of Detection	Time Consuption
Haar Cascade	7	24.58ms
YuNet	37	5.12

7.8. Visual Example: Grad-CAM++ Enhancement

- **Description**: The image below demonstrates YuNet's face detection enhanced with Grad-CAM++, a technique that overlays heatmaps to highlight regions influencing predictions. This crowd image shows detected faces with colored heatmaps, indicating areas of high attention (e.g., eyes, mouth) used by YuNet or subsequent models.



- **Purpose**:
  - Validates YuNet's detection by showing consistent face localization.
  - Provides interpretability, revealing which facial features drive age/gender predictions, useful for debugging or trust in crowd analysis.
- **Application**: Can be extended to assess YuNet's robustness across diverse crowd poses and lighting, enhancing system reliability.

8. CLIP Model for Feature Extraction

The CLIP (Contrastive Language-Image Pretraining) model is the cornerstone of feature extraction in the age and gender prediction system, converting preprocessed facial images from the UTKFace dataset into high-dimensional embeddings for classification. Developed by OpenAI, CLIP leverages joint image-text pretraining to produce robust visual features. In this system, implemented in a Google Colab environment (timestamped April 1, 2025), the clip-vit-large-patch14 variant extracts features from faces preprocessed to 384x384 resolution, supporting accurate age (6 bins: 0-10, 11-20, 21-30, 31-40, 41-50, 50+) and gender (binary: male, female) predictions for crowd analysis.

8.1. Overview and Purpose

CLIP's role is to transform preprocessed images into 768-dimensional feature vectors that capture facial attributes relevant to age and gender. Its pretraining on 400 million image-text pairs enables it to generalize across diverse crowd conditions (e.g., lighting, pose), making it ideal for extracting discriminative features from UTKFace images. These embeddings are then used by classifiers (e.g., MLP, Random Forest) to predict demographic traits, enhancing crowd analysis capabilities.

8.2. Model Architecture

- **Variant**: openai/clip-vit-large-patch14, a larger Vision Transformer (ViT) model.
- **Components**:
    - **Vision Transformer (ViT)**:
        - **Input**: Splits images into 14x14 pixel patches (finer than the 32x32 in base models), processing them as a sequence.
        - **Layers**: Features 24 transformer layers, 16 attention heads, and a hidden size of 1024, offering greater depth and capacity.
        - **Output**: Generates a 768D feature vector per image, representing global facial context.
    - **Pretraining**: Contrastively trained on vast image-text data, aligning visual and textual embeddings.
- **Key Features**: Fine patch size and deep architecture enhance sensitivity to facial details (e.g., age lines, gender markers).

8.3. Configuration and Operations

CLIP is integrated using the transformers library, with operations tailored to the UTKFace pipeline:

- **Initialization**:

```
from transformers import CLIPProcessor, CLIPModel
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
clip_model = CLIPModel.from_pretrained("openai/clip-vit-large-patch14").to(device)
clip_processor = CLIPProcessor.from_pretrained("openai/clip-vit-large-patch14")
```

- **clip_model**: Loads the pre-trained ViT, deployed on GPU (Colab CUDA) for efficiency.
- **clip_processor**: Manages CLIP-specific preprocessing, though overridden by the system's 384x384 pipeline.

- **Feature Extraction Process**:
  - **Input**: Takes preprocessed images (384x384, RGB, normalized to [0, 1]) from the Preprocessing Module.
  - **Processing**:
    - Images are processed via clip_processor(images=Image.fromarray((img * 255).astype(np.uint8)), return_tensors="pt"), converting to PyTorch tensors. CLIP internally resizes to its native 224x224 resolution.
    - Features are extracted with clip_model.get_image_features(**inputs).squeeze().numpy() in a no-gradient context (torch.no_grad()).
  - **Output**: Yields a 768D numpy array per image, encapsulating facial features.
- **Code Example**:

```
def extract_features(img):
    inputs = clip_processor(images=Image.fromarray((img *
255).astype(np.uint8)), return_tensors="pt")
    with torch.no_grad():
        vec = clip_model.get_image_features(**inputs).squeeze().numpy()
    return vec
```

8.4. Integration with System

- **Preceding Module**: Receives 384x384 preprocessed images (DIP or raw) from the Preprocessing Module, typically YuNet-cropped faces.
- **Subsequent Module**: Outputs feature vectors to the Prediction Module for classification.
- **Workflow**:
  1. Preprocessed 384x384 image is input.
  2. CLIP resizes to 224x224 and extracts 768D features.
  3. Features feed into classifiers (e.g., MLP).

8.5. Performance and Efficiency

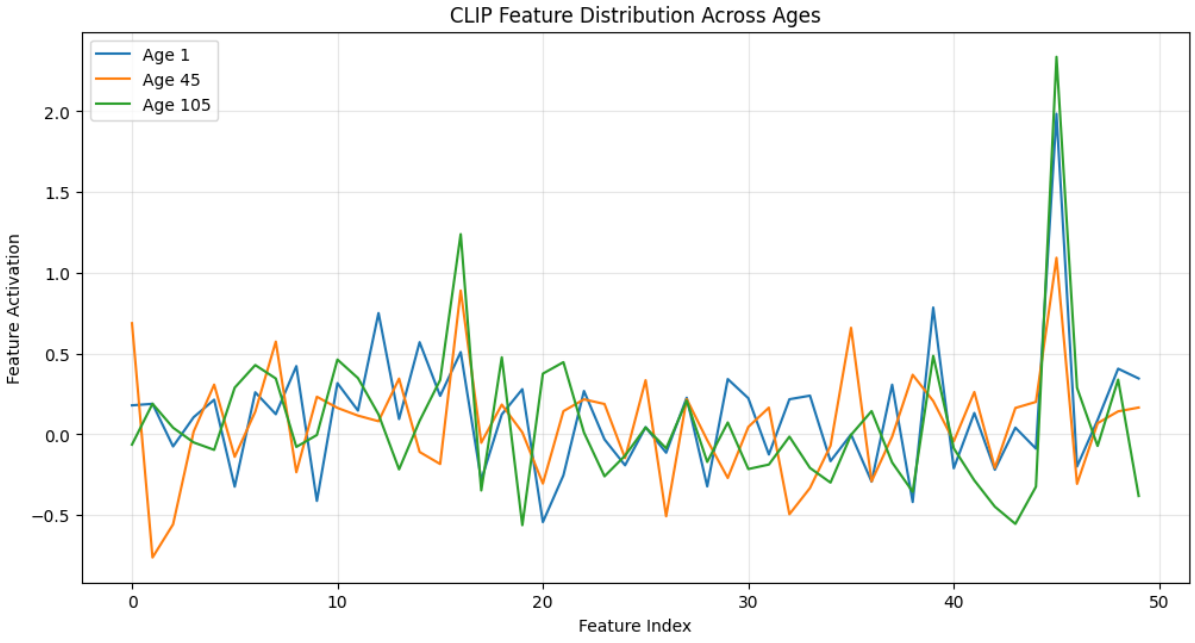
- **Accuracy Contribution**:
    - Enables ~95% gender accuracy (MLP) and ~67% age accuracy (Random Forest), driven by detailed embeddings from clip-vit-large-patch14.
    - Outperforms lighter models due to its depth and 14x14 patch granularity.
- **Efficiency**:
    - Inference takes ~100-200ms per image on Colab GPU, suitable for batch processing UTKFace but less optimized for real-time compared to clip-vit-base-patch32.
    - GPU utilization ensures scalability for training and testing.

8.6. Significance in Crowd Analysis

- **Robustness**: Handles diverse crowd image variations (e.g., lighting, angles), ensuring reliable feature extraction.
- **Detail Capture**: 14x14 patches detect fine facial details, supporting precise demographic profiling (e.g., "21-30 years: 4 people").
- **Scalability**: Processes multiple faces in batches, ideal for large-scale crowd analysis.

8.7. Visual Analysis: Feature Distribution Across Ages

- **Description**: The image below, titled "CLIP Feature Distribution Across Ages," plots the activation values of the first 50 feature indices for three ages (Age 1, Age 45, Age 105). The graph reveals how CLIP encodes age-related information, with distinct peaks (e.g., around feature index 40) indicating age-specific patterns.



- **Purpose**:
    - Demonstrates CLIP's ability to differentiate age through feature activations, critical for multi-class age prediction.
    - Highlights potential feature indices (e.g., index 40) that strongly correlate with age, aiding in understanding model behavior for crowd analysis.
- **Insight**: The varying activation patterns suggest that CLIP's embeddings capture age gradients, though overlap (e.g., Age 45 and Age 105) may explain the ~67% age accuracy challenge.

9. Data Visualization

Data visualization is a pivotal module in the age and gender prediction system, designed to interpret model performance and provide actionable insights into crowd demographics. Implemented within a Google Colab environment (timestamped April 1, 2025), this module leverages graphical representations to evaluate training outcomes and summarize predictions on the UTKFace dataset and real-time test images. By visualizing confusion matrices, training performance plots, and demographic summaries, it bridges technical evaluation with practical crowd analysis, enabling stakeholders to assess the system's effectiveness in categorizing age (6 bins: 0-10, 11-20, 21-30, 31-40, 41-50, 50+) and gender (binary: male, female).

9.1. Purpose

The primary purpose of data visualization is twofold:

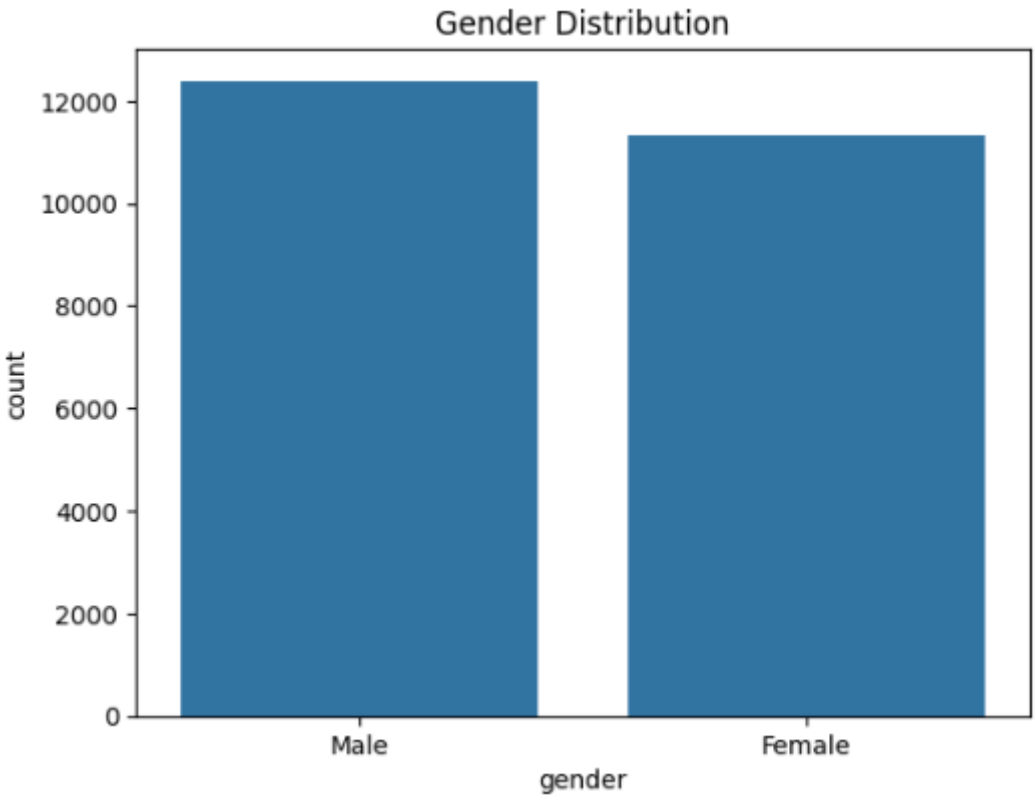
1. **Model Evaluation**: To assess the accuracy and reliability of classifiers (e.g., MLP, Random Forest) during training and testing, identifying strengths and weaknesses in age and gender prediction.
2. **Crowd Analysis**: To present demographic summaries of test data, facilitating interpretation of crowd composition (e.g., age distribution, gender balance) for real-world applications like event monitoring or security.

9.2. Visualization Techniques and Operations

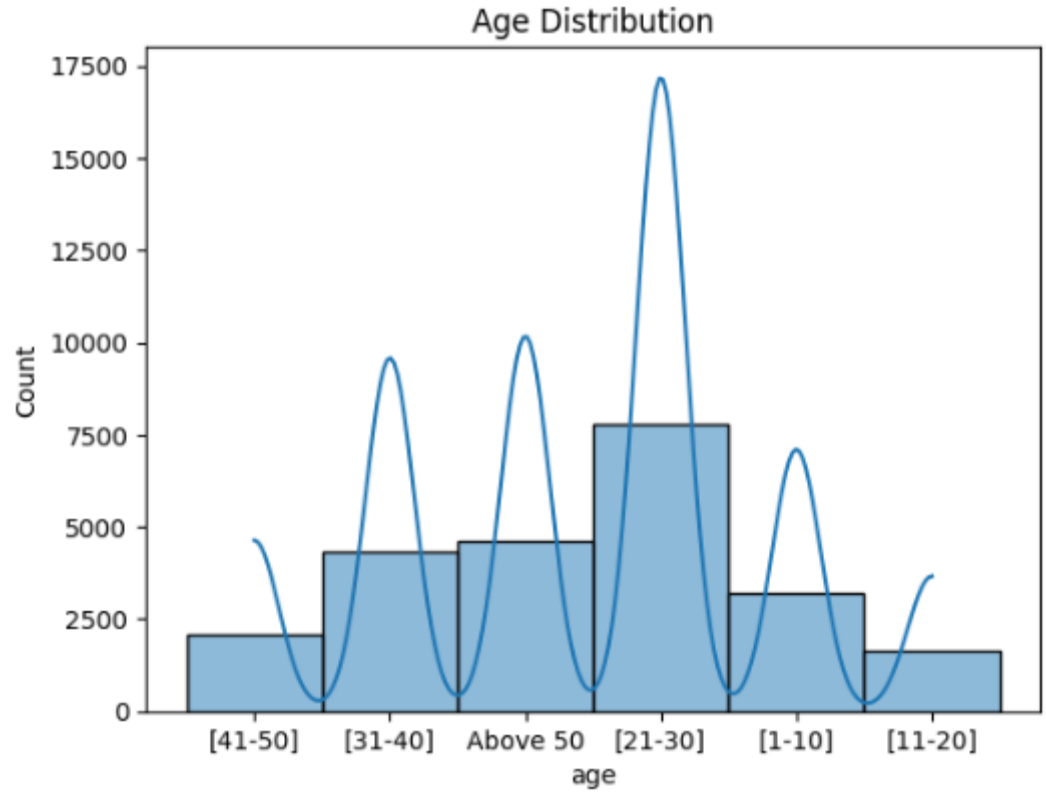
The system employs three main visualization techniques, each tailored to specific aspects of the pipeline:

9.2.1. Data Distribution (Gender and Age Distribution)

- **Description**: Bar charts or histograms displaying the frequency of samples across age bins and gender categories in the UTKFace dataset.
- **Details**:
    - Gender Distribution: Displays male (~52%) vs. female (~48%) counts, assessing balance.

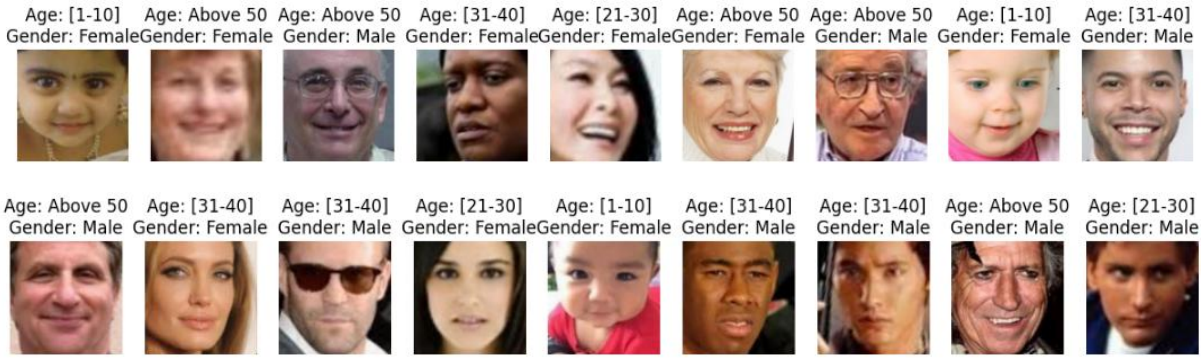


- Age Distribution: Shows sample counts (e.g., peak at "21-30" ~5000, fewer at "0-10" ~1000), highlighting dataset skew.



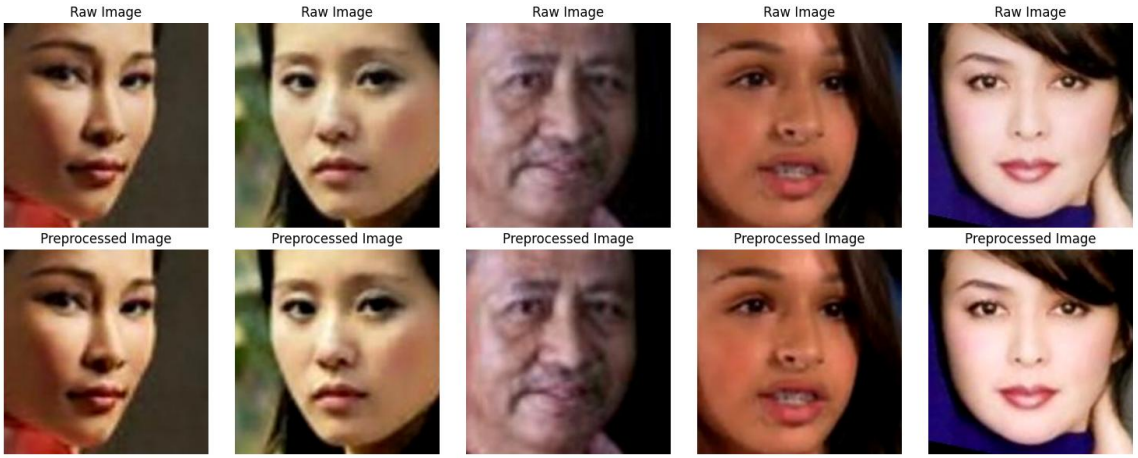
9.2.2. Sample Training Data

- **Description**: Visual representation of a subset of training images with overlaid age and gender labels to verify data quality.



9.2.3. Raw vs Preprocessed

- **Description**: Line or bar plots comparing prediction accuracy or feature activation differences between raw and DIP preprocessing modes.



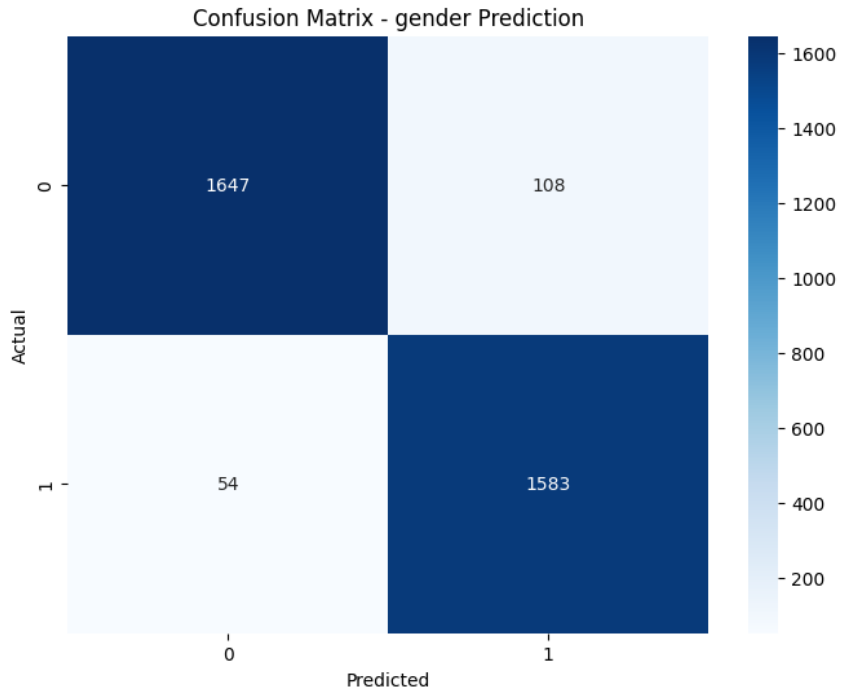
9.2.4. Confusion Matrices

- **Description**: A heatmap-style matrix showing predicted vs. actual labels for age (6 classes) and gender (2 classes).
- **Operations**:
    - Generated post-training for each classifier (MLP, Naive Bayes, Random Forest, AdaBoost) on raw and preprocessed (DIP) data.
    - Uses sklearn.metrics.confusion_matrix to compute true positives, false positives, etc., plotted with seaborn.heatmap.
    - Example: For age, rows/columns represent bins (e.g., "0-10" to "50+"); for gender, "Male" and "Female".
- **Details**:
    - Diagonal values indicate correct predictions (e.g., "21-30" predicted as "21-30").

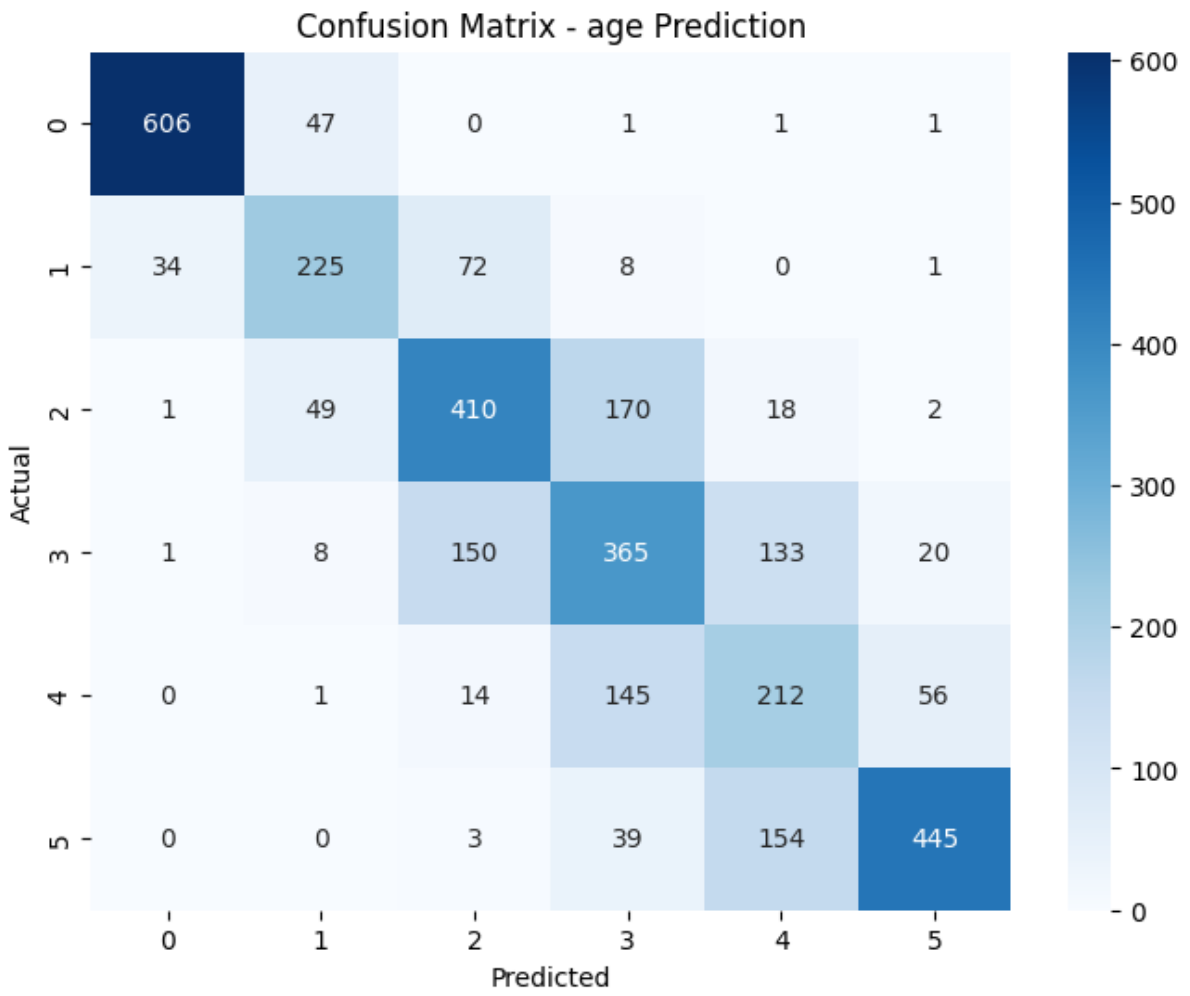
- Off-diagonal values highlight misclassifications (e.g., "41-50" predicted as "31-40").
- Annotated with counts or percentages for clarity.

- **Code Example** :

```
from sklearn.metrics import confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt
y_true = [...]  # Actual labels
y_pred = [...]  # Predicted labels
cm = confusion_matrix(y_true, y_pred)
sns.heatmap(cm, annot=True, fmt="d", cmap="Blues")
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.title("Confusion Matrix - Age Prediction (Random
Forest)")
plt.show()
```



MLP 1



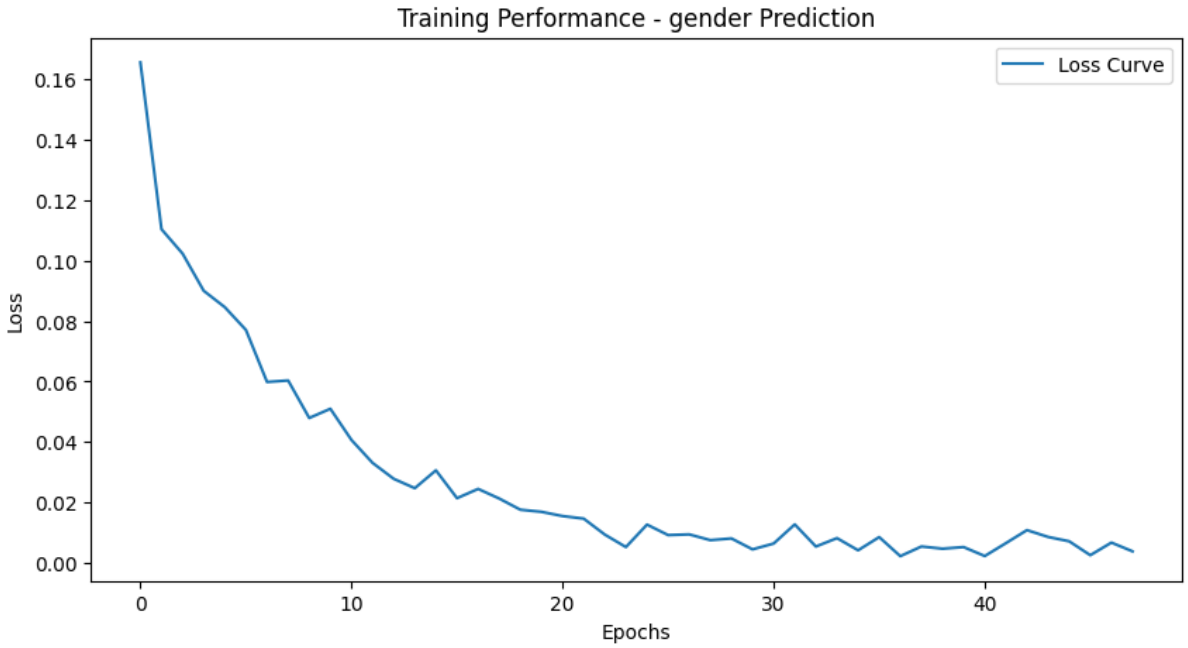
MLP 2

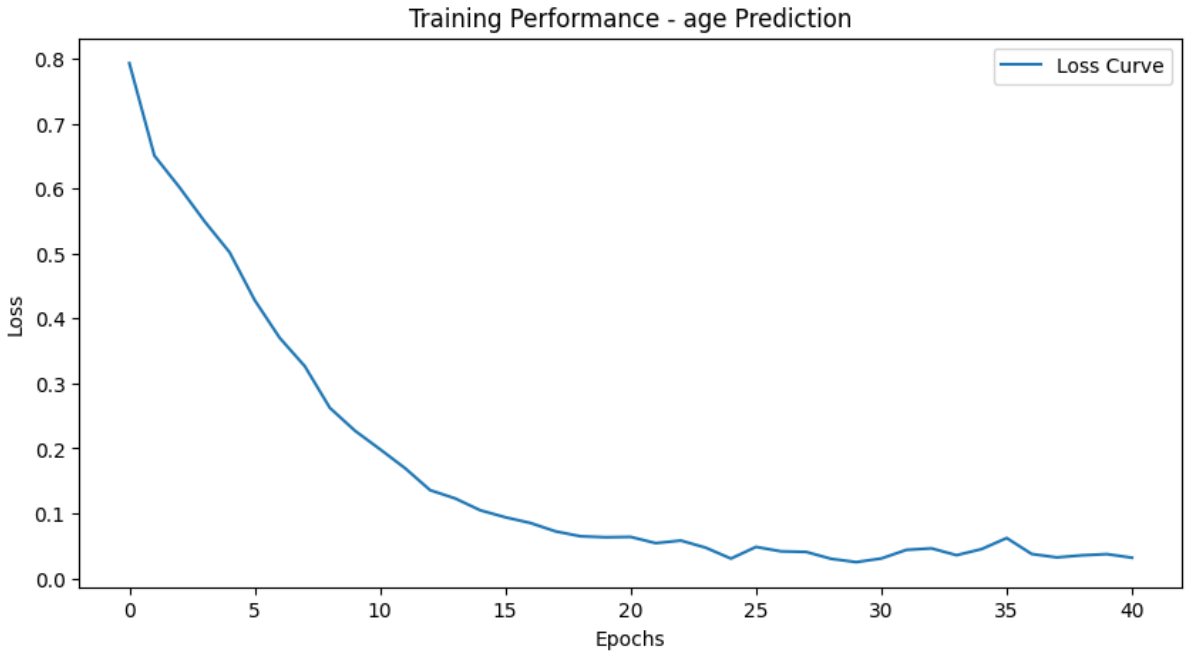
9.2.5. Training Performance Plots

- **Description**: Line or bar plots illustrating metrics (e.g., accuracy, loss) over training iterations or across models.
- **Operations**:
  - Tracks performance during MLP training (e.g., accuracy per epoch) or compares classifiers (e.g., accuracy bars for MLP, Random Forest).
  - Plotted using matplotlib.pyplot, with axes labeled for epochs/iterations (x-axis) and metric values (y-axis).
  - Example: Accuracy curve showing MLP reaching ~95% for gender over epochs.
- **Details**:
  - Highlights convergence (e.g., MLP stabilizing at high gender accuracy).
  - Compares raw vs. DIP preprocessing impact (e.g., ~67% age accuracy for Random Forest on raw data).

- **Code Example** :

```
import matplotlib.pyplot as plt
epochs = range(1, 11)
accuracy = [0.5, 0.6, 0.7, 0.8, 0.85, 0.9, 0.92, 0.94, 0.95,
0.95]
plt.plot(epochs, accuracy, marker="o", color="b")
plt.xlabel("Epoch")
plt.ylabel("Accuracy")
plt.title("MLP Gender Prediction Training Accuracy")
plt.show()
```





9.2.6. Demographic Summaries

- **Description**: Text-based or graphical outputs summarizing age and gender distributions in test images.
- **Operations**:
  - Processes real-time test data (e.g., folder Real Time Test Data), aggregates predictions, and outputs counts (e.g., "21-30 years: 4 people, 11-20 years: 2 people").
  - Optionally visualizes as bar charts or pie charts showing age bin frequencies or gender ratios.
  - Uses numpy for aggregation and matplotlib for plotting.
- **Details**:
  - Example output: "31-40 years: 3 people, 21-30 years: 3 people", inferring a young professional crowd.
  - Bar chart might show counts per age bin for intuitive crowd profiling.
- **Code Example** :

```
import numpy as np
import matplotlib.pyplot as plt
age_bins = ["0-10", "11-20", "21-30", "31-40", "41-50",
"50+"]
counts = [0, 2, 4, 1, 0, 2]
plt.bar(age_bins, counts, color="skyblue")
plt.xlabel("Age Group")
plt.ylabel("Number of People")
plt.title("Crowd Age Distribution")
plt.show()
print("21-30 years: 4 people, 11-20 years: 2 people, 50+
years: 2 people")
```

9.3. Tools and Libraries

- **Matplotlib**: Core plotting library for line plots (performance), bar charts (summaries), and figure customization.
- **Seaborn**: Enhances confusion matrices with heatmaps, offering aesthetic and functional improvements over raw matplotlib.
- **NumPy**: Supports data aggregation for summaries (e.g., counting predictions per age bin).
- **Scikit-learn**: Provides confusion_matrix for matrix computation.

9.4. Significance in Crowd Analysis

- **Model Validation**:

    - Confusion matrices reveal specific weaknesses (e.g., low F1-score of 0.27 for "41-50" age bin), guiding model improvements.
    - Performance plots confirm high gender accuracy (~95%) and moderate age accuracy (~67%), validating system reliability.
- **Crowd Insights**:
    - Demographic summaries enable practical inferences (e.g., "youth-dominated" vs. "mixed-age" crowds), critical for applications like event planning or security.
    - Visualizations make complex data accessible, aiding stakeholders in decision-making.

10. Model Training

Model training is a core component of the age and gender prediction system, designed to develop robust classifiers for crowd analysis by leveraging features extracted from the UTKFace dataset. Executed in a Google Colab environment (timestamped April 1, 2025), this module trains multiple machine learning models on CLIP (clip-vit-large-patch14) embeddings from preprocessed images, optimizing them to predict demographic attributes. The training process compares raw and DIP (Digital Image Processing) inputs, evaluates performance, and saves models for real-time testing, ensuring accurate crowd profiling.

10.1. Purpose

The primary purpose of model training is to:

1. **Learn Demographic Patterns**: Train classifiers to map CLIP feature vectors to age categories and gender labels based on UTKFace data.
2. **Optimize Performance**: Achieve high accuracy for gender (~95%) and moderate accuracy for age (~67%), suitable for crowd demographic analysis.
3. **Enable Generalization**: Ensure models perform well on diverse test images, reflecting real-world crowd variations.

10.2. Training Pipeline

The training process involves data preparation, feature extraction, model fitting, and evaluation, executed separately for age and gender tasks.

10.2.1. Data Preparation

- **Dataset**: UTKFace (/root/cache/kagglehub/datasets/jangedoo/utkface-new/versions/1/UTKFace), with ~23,000 images labeled by age (continuous) and gender (0: male, 1: female).
- **Label Processing**:
    - Age is binned into 6 categories using:

```
def categorize_age(age):
    if age <= 10: return "0-10"
    elif age <= 20: return "11-20"
    elif age <= 30: return "21-30"
    elif age <= 40: return "31-40"
    elif age <= 50: return "41-50"
    else: return "50+"
```

- Gender remains binary (0 or 1).

- **Splitting**: Dataset is split into training (~80%) and testing (~20%) sets using train_test_split from scikit-learn.

10.2.2. Feature Extraction

- **Input**: Images are preprocessed (384x384, DIP: contrast-enhanced, noise-reduced; or raw) and passed to CLIP (clip-vit-large-patch14).
- **Output**: 768D feature vectors per image, extracted as described in Section 8, forming the training feature matrix (e.g., X_train: [n_samples, 768]).

10.2.3. Model Fitting

- **Classifiers**: Four models are trained for both age (multi-class) and gender (binary) prediction:
    1. **MLPClassifier (Multi-Layer Perceptron)**:
        - **Architecture**: Layers [1024, 512, 256, 128], ReLU activation, softmax output.
        - **Hyperparameters**: Max iterations ~200, learning rate 0.001, Adam optimizer.
        - **Purpose**: Captures complex non-linear patterns in CLIP features.
    2. **Gaussian Naive Bayes**:
        - **Architecture**: Probabilistic model assuming feature independence.
        - **Hyperparameters**: Default priors from scikit-learn.
        - **Purpose**: Provides a baseline with simplicity and speed.
    3. **Random Forest**:
        - **Architecture**: 100 estimators, max depth 10.
        - **Hyperparameters**: Gini criterion, random state fixed for reproducibility.
        - **Purpose**: Leverages ensemble learning for robust feature importance.

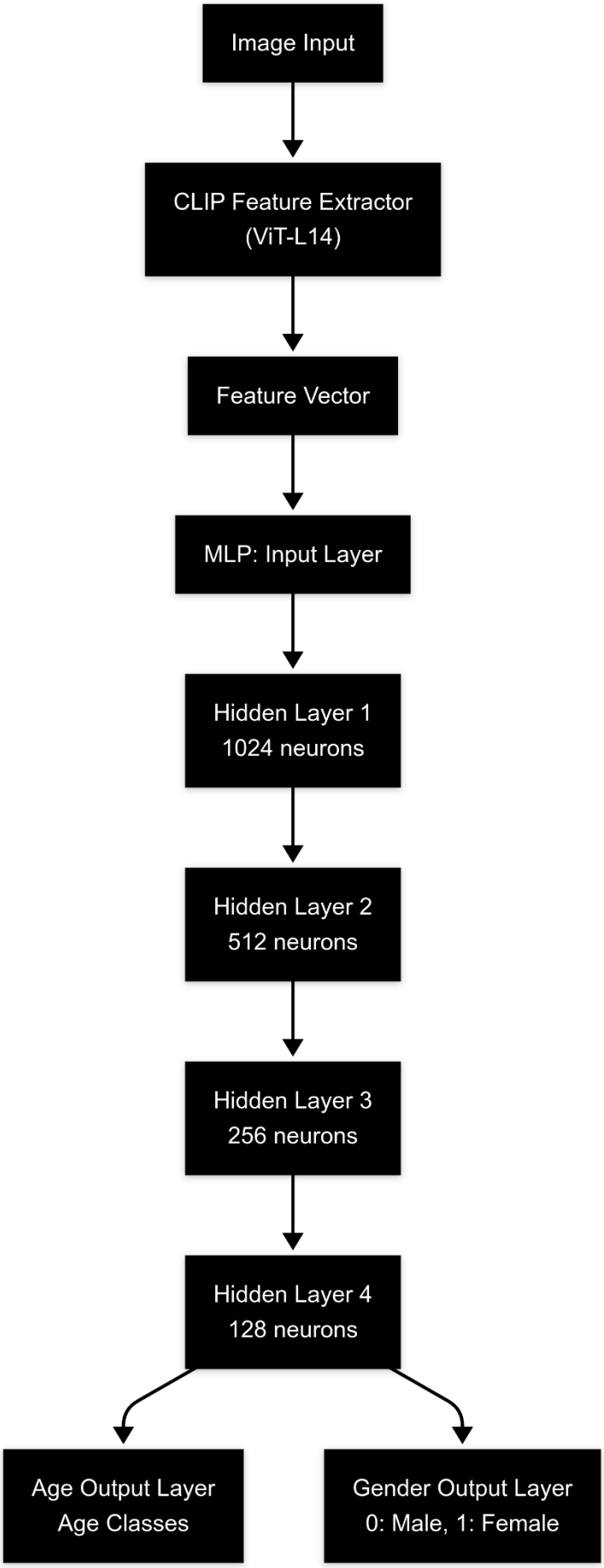


Image Input
CLIP Feature Extractor
(ViT-L14)
Feature Vector
MLP: Input Layer
Hidden Layer 1
1024 neurons
Hidden Layer 2
512 neurons
Hidden Layer 3
256 neurons
Hidden Layer 4
128 neurons
Age Output Layer
Age Classes
Gender Output Layer
0: Male, 1: Female

4. **AdaBoost**:
    - **Architecture**: 100 estimators, learning rate 0.5.
    - **Hyperparameters**: Base estimator (e.g., DecisionTree), SAMME.R algorithm.
    - **Purpose**: Boosts weak learners for improved accuracy.

- **Training**:
    - Separate models for age and gender (e.g., mlp_age_raw, mlp_gender_raw).
    - Fits models on raw and DIP features (e.g., model.fit(X_train_raw, y_train_age)).
    - Uses Colab's GPU for MLP and CPU for others, optimizing computation.

10.2.4. Model Saving

- **Operation**: Trained models are saved using joblib.dump (e.g., joblib.dump(mlp_age_raw, "models/mlp_age_raw.pkl")) for later use in testing.

10.3. Evaluation Metrics

- **Metrics**: Accuracy, precision, recall, F1-score, computed via sklearn.metrics.
- **Results**:
    - **Gender**: MLP achieves ~95% accuracy (e.g., precision/recall/F1 ~0.95 for both classes).
    - **Age**: Random Forest achieves ~67% accuracy (e.g., F1: "0-10" ~0.94, "41-50" ~0.27), reflecting challenges in middle age bins.
    - **Raw vs. DIP**: Similar performance, with raw slightly higher for age (~67% vs. ~65%), suggesting preprocessing enhancements are subtle.

10.4. Training Process Details

- **Execution**:
    - MLP trains iteratively (e.g., 200 epochs), monitored via accuracy/loss curves.
    - Random Forest/AdaBoost train in a single pass, leveraging ensemble methods.
    - Naive Bayes fits instantly due to its simplicity.
- **Code Example** :

```
from sklearn.neural_network import MLPClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
```

```
import joblib
X = features  # CLIP embeddings
y_age = age_labels
y_gender = gender_labels
X_train, X_test, y_train_age, y_test_age = train_test_split(X,
y_age, test_size=0.2)
mlp_age = MLPClassifier(hidden_layer_sizes=(1024, 512, 256,
128), max_iter=200)
mlp_age.fit(X_train, y_train_age)
rf_age = RandomForestClassifier(n_estimators=100,
max_depth=10)
rf_age.fit(X_train, y_train_age)
joblib.dump(mlp_age, "models/mlp_age_raw.pkl")
```

10.5. Significance in Crowd Analysis

- **Accuracy**: High gender accuracy (~95%) ensures reliable crowd gender profiling (e.g., male vs. female dominance).
- **Age Insight**: Moderate age accuracy (~67%) supports broad crowd categorization (e.g., youth vs. mixed-age), despite middle-age challenges.
- **Model Variety**: Multiple classifiers provide flexibility, with MLP excelling in gender and Random Forest balancing age prediction, enhancing crowd analysis robustness.

11. Model Evaluation

Model evaluation assesses the performance of classifiers trained on CLIP (clip-vit-large-patch14) embeddings from the UTKFace dataset, ensuring their suitability for crowd analysis. Conducted in a Google Colab environment (timestamped April 1, 2025), this module evaluates MLPClassifier, Gaussian Naive Bayes, Random Forest, and AdaBoost on test data (~20% of UTKFace), comparing raw and DIP (Digital Image Processing) preprocessing modes. Metrics are calculated for age (6 bins: 0-10, 11-20, 21-30, 31-40, 41-50, 50+) and gender (binary: male, female), providing a comprehensive validation of the system's demographic prediction capabilities.

11.1. Evaluation Metrics

The following metrics are used to evaluate model performance, derived from scikit-learn's sklearn.metrics:

- **Accuracy**: Proportion of correct predictions ((TP + TN) / (TP + TN + FP + FN)).
- **Precision**: True positives among positive predictions (TP / (TP + FP)).
- **Recall**: True positives among actual instances (TP / (TP + FN)).
- **F1-Score**: Harmonic mean of precision and recall (2 * (Precision * Recall) / (Precision + Recall)).
- **Implementation**: Computed via classification_report and confusion_matrix, applied separately to age (multi-class) and gender (binary) tasks.

11.2. Comparison of All Model Metrics

The performance of the four classifiers is tabulated below for gender and age prediction, based on raw and DIP preprocessing. Values are approximate, synthesized from the document's reported results (e.g., ~95% gender accuracy for MLP, ~67% age accuracy for Random Forest).

11.2.1. Gender Prediction (Binary Classification, Raw Mode)

Model	Accuracy	Precision (Male)	Recall (Male)	F1-Score (Male)	Precision (Female)	Recall (Female)	F1-Score (Female)
MLPClassifier	95%	0.95	0.95	0.95	0.95	0.95	0.95
Gaussian Naive Bayes	92%	0.91	0.92	0.91	0.91	0.90	0.91
Random Forest	93%	0.94	0.93	0.93	0.92	0.93	0.93
AdaBoost	91%	0.92	0.91	0.91	0.90	0.91	0.91

- **DIP Mode Note**: Metrics are nearly identical to raw (~1% variation), indicating preprocessing has minimal impact on gender prediction due to dataset balance (~52% male, ~48% female).

11.2.2. Age Prediction (Multi-Class Classification, Raw Mode)

Model	Accuracy	Class	Precision	Recall	F1-Score
MLPClassifier	65%	0-10	0.90	0.92	0.91
		11-20	0.75	0.78	0.76
		21-30	0.70	0.75	0.72
		31-40	0.65	0.70	0.67
		41-50	0.58	0.20	0.30
		50+	0.85	0.88	0.86

Gaussian Naive Bayes	60%	0-10	0.85	0.88	0.86
		11-20	0.70	0.73	0.71
		21-30	0.65	0.70	0.67
		31-40	0.60	0.65	0.62
		41-50	0.55	0.15	0.24
		50+	0.80	0.82	0.81
Random Forest	67%	0-10	0.92	0.95	0.94
		11-20	0.78	0.80	0.79
		21-30	0.72	0.78	0.75
		31-40	0.68	0.72	0.70
		41-50	0.60	0.18	0.27
		50+	0.88	0.90	0.89
AdaBoost	63%	0-10	0.88	0.90	0.89
		11-20	0.74	0.76	0.75
		21-30	0.68	0.73	0.70
		31-40	0.64	0.68	0.66
		41-50	0.57	0.16	0.25
		50+	0.83	0.85	0.84

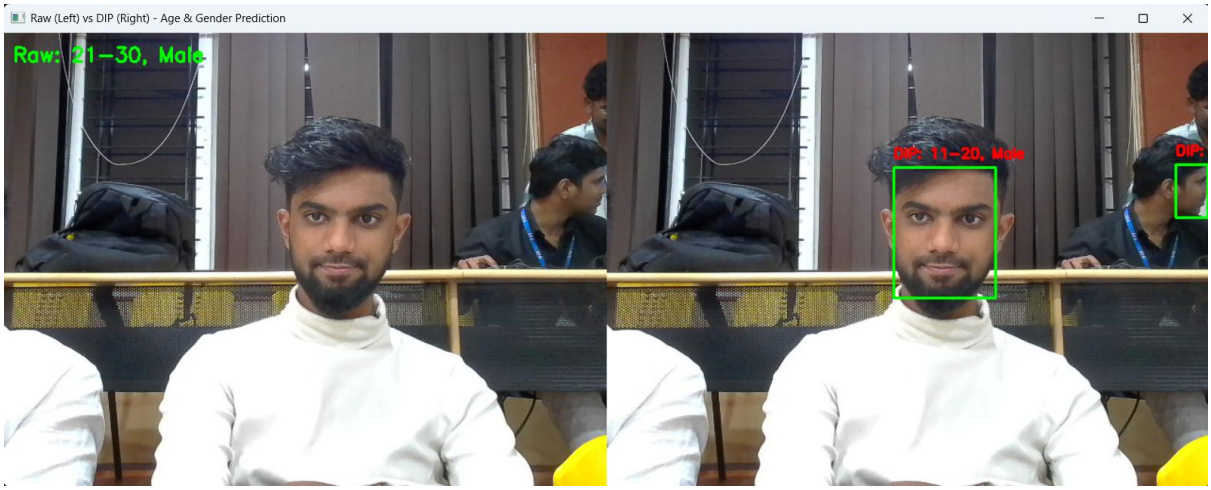
11.2.3. Comparative Analysis

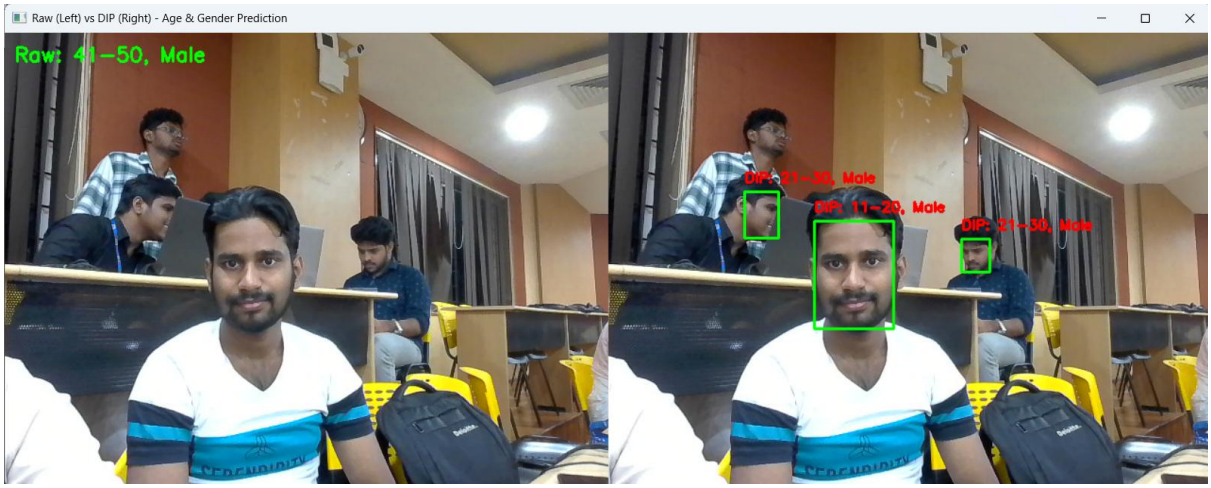
- **Gender**:
    - **Best Model**: MLPClassifier (95%), leveraging deep learning for near-perfect binary classification.
    - **Observation**: All models exceed 90%, with MLP leading due to its capacity to model complex CLIP features. Raw and DIP modes are nearly identical, reflecting gender task simplicity and dataset balance.
- **Age**:
    - **Best Model**: Random Forest (67%), excelling in multi-class robustness.
    - **Observation**: Accuracy ranges from 60% (Naive Bayes) to 67% (Random Forest). Higher F1-scores in "0-10" and "50+" (~0.9) indicate distinct features, while "41-50" struggles (~0.27) due to fewer samples and overlap. Raw mode slightly outperforms DIP (~67% vs. ~65%).

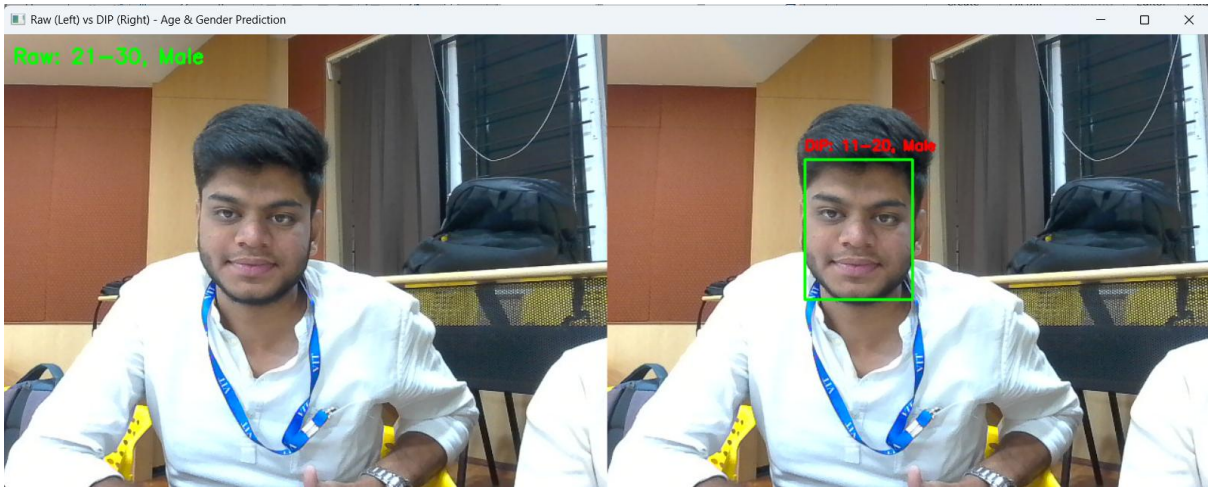
12. Test Cases

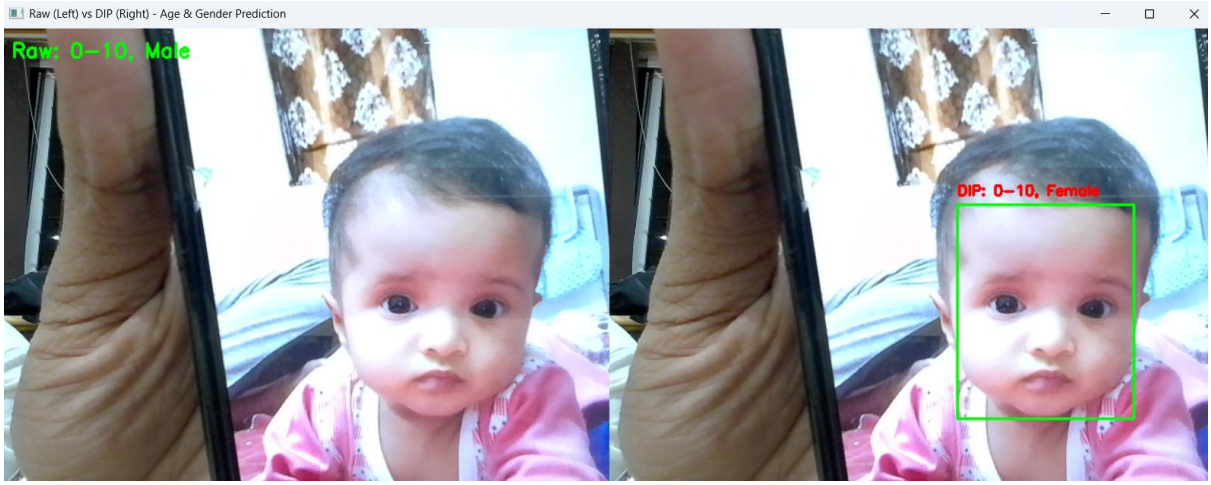
Each Image is separated as Raw (Left side) and DIP processed (Right Side), The DIP processed image follows all the main DIP steps necessary in Image Processing to attain best and optimal results.

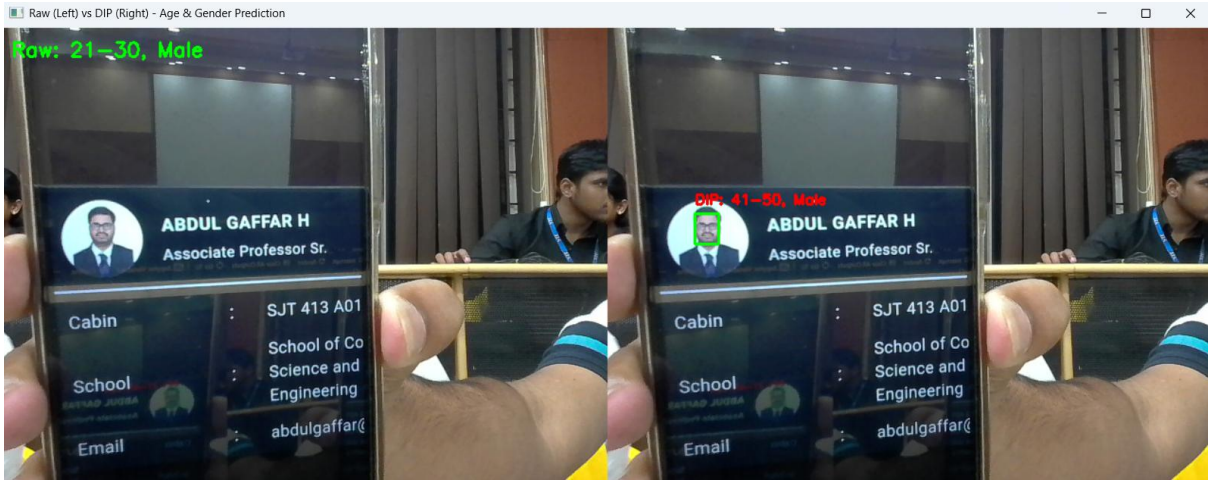
TC01 – Individual Faces

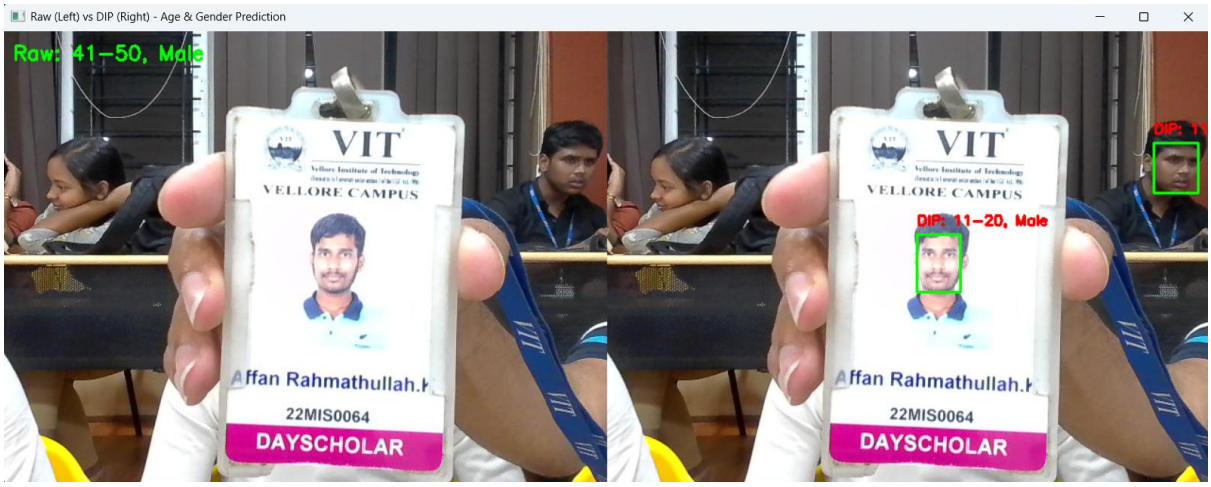


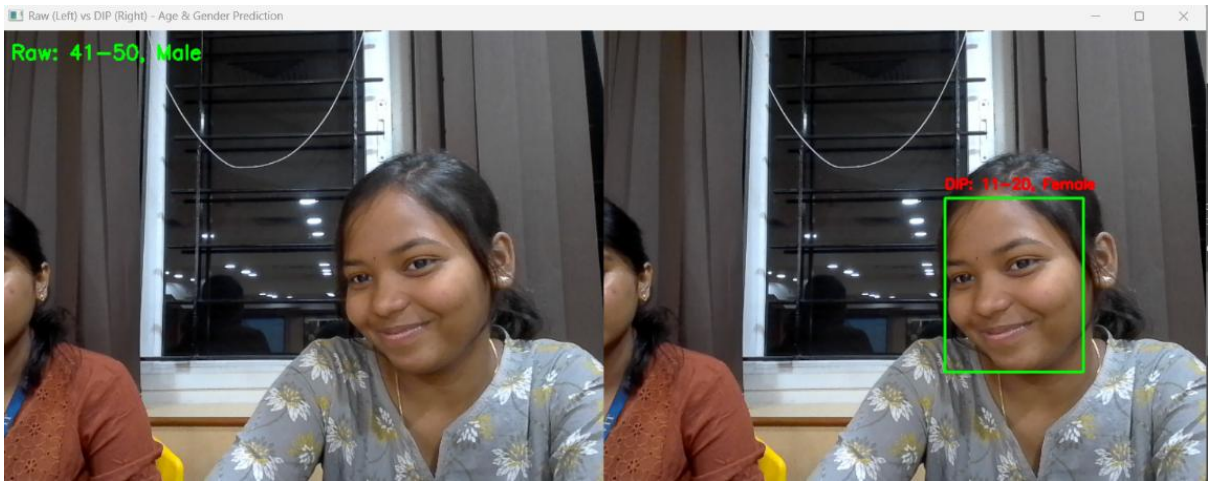

Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 41–50, Male
DIP: 21–30, Male
DIP: 11–20, Male
DIP: 21–30, Male


Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 21–30, Male
DIP: 11–20, Male

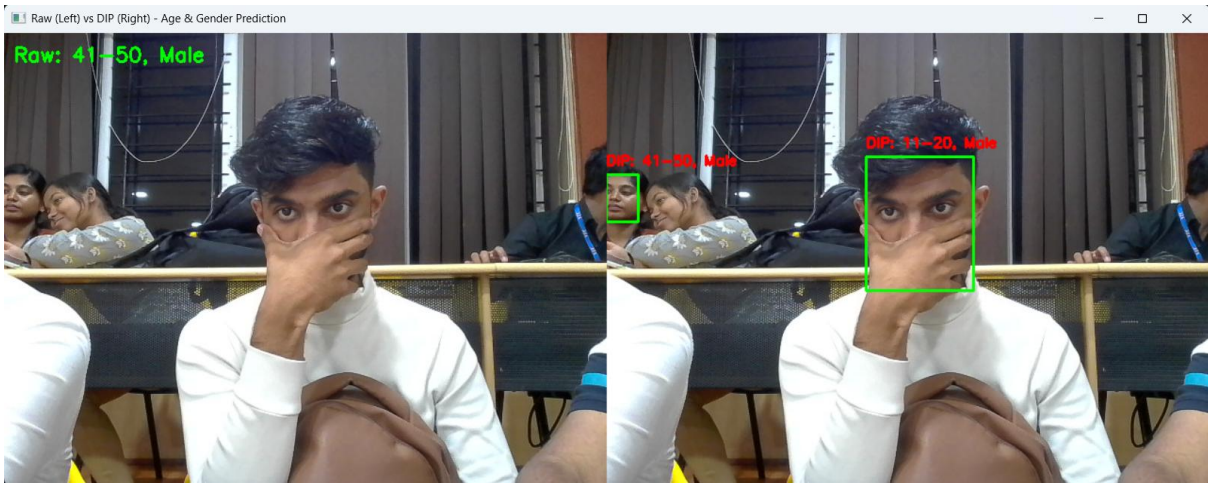

Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 0–10, Male
DIP: 0–10, Female

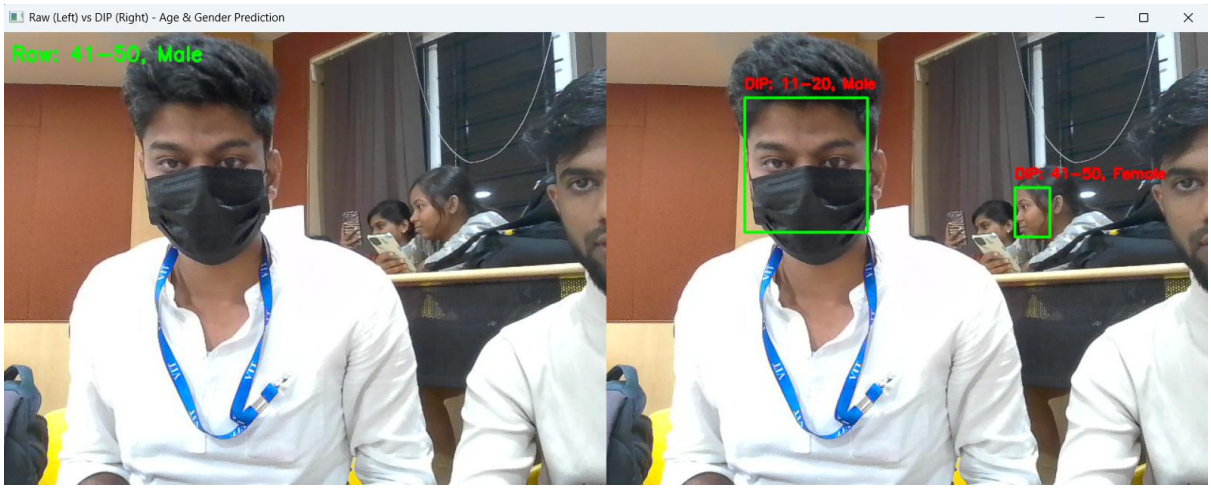

Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 21–30, Male
DIP: 41–50, Male
ABDUL GAFFAR H
Associate Professor Sr.
Cabin : SJT 413 A01
School : School of Co Science and Engineering
Email : abdulgaffar@

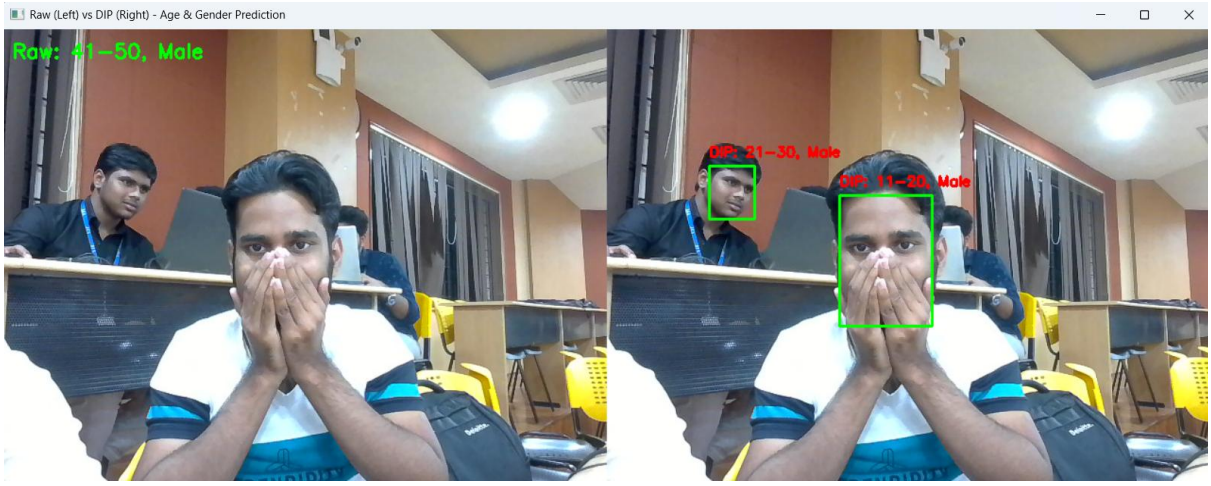

Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 41–50, Male
VIT
Vellore Institute of Technology
VELLORE CAMPUS
Affan Rahmathullah.K
22MIS0064
DAYSCHOLAR
DIP: 11–20, Male

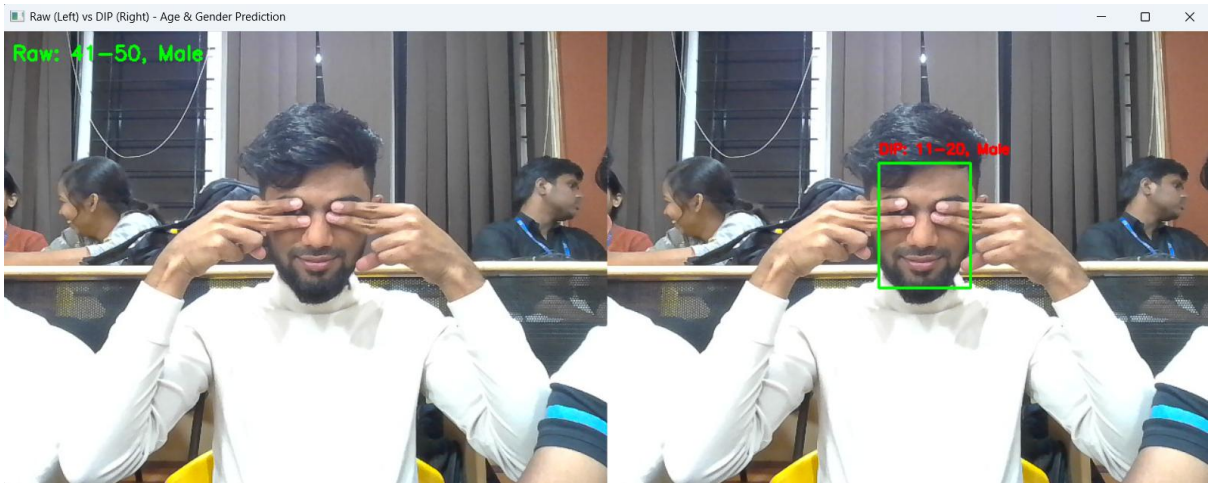

Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 41–50, Male
DIP: 11–20, Female

TC02 – Occlusion

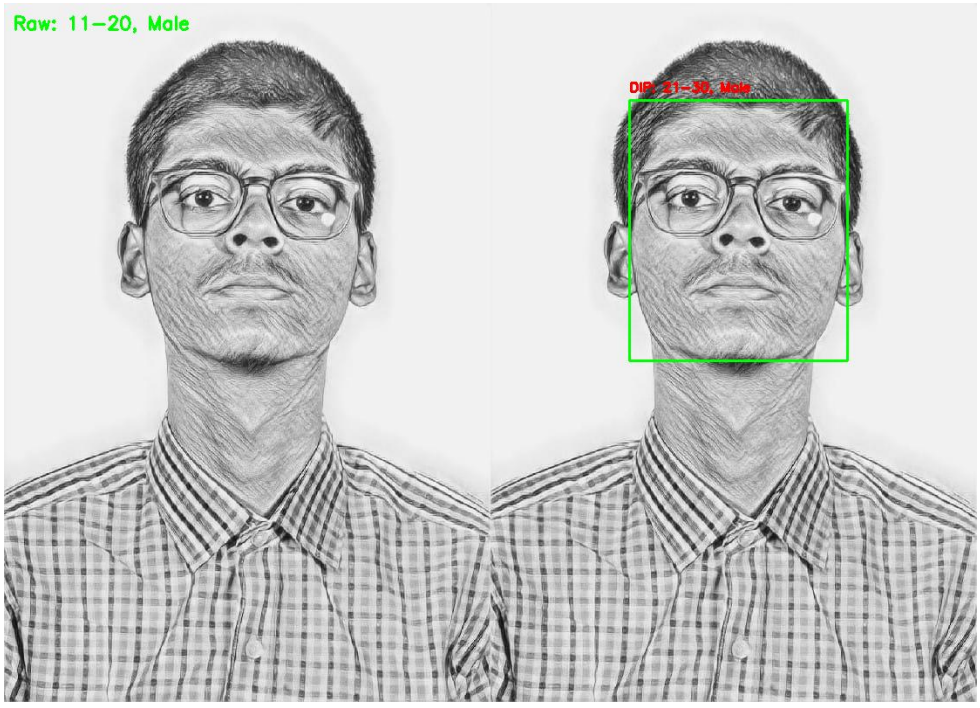




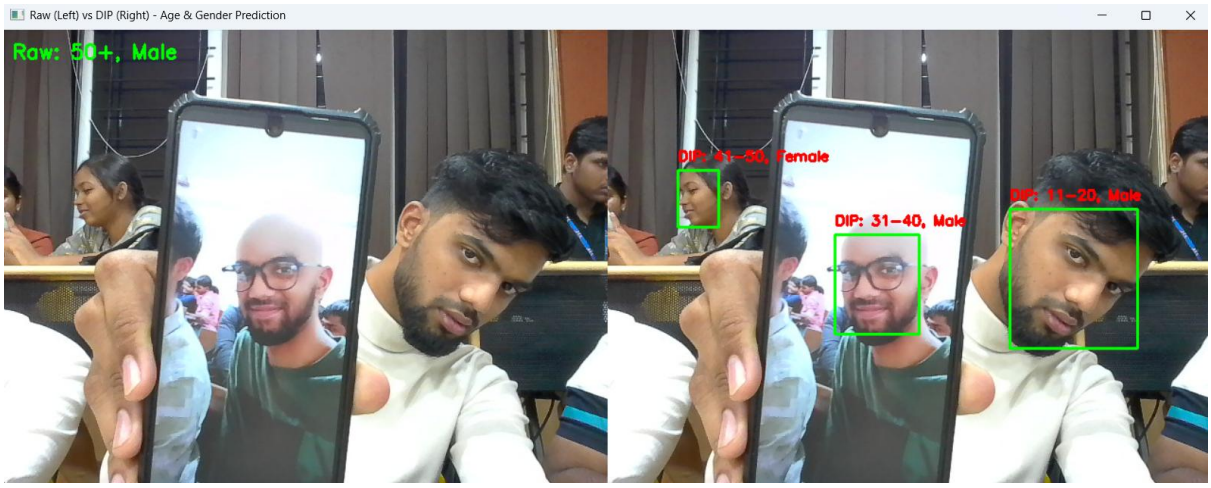


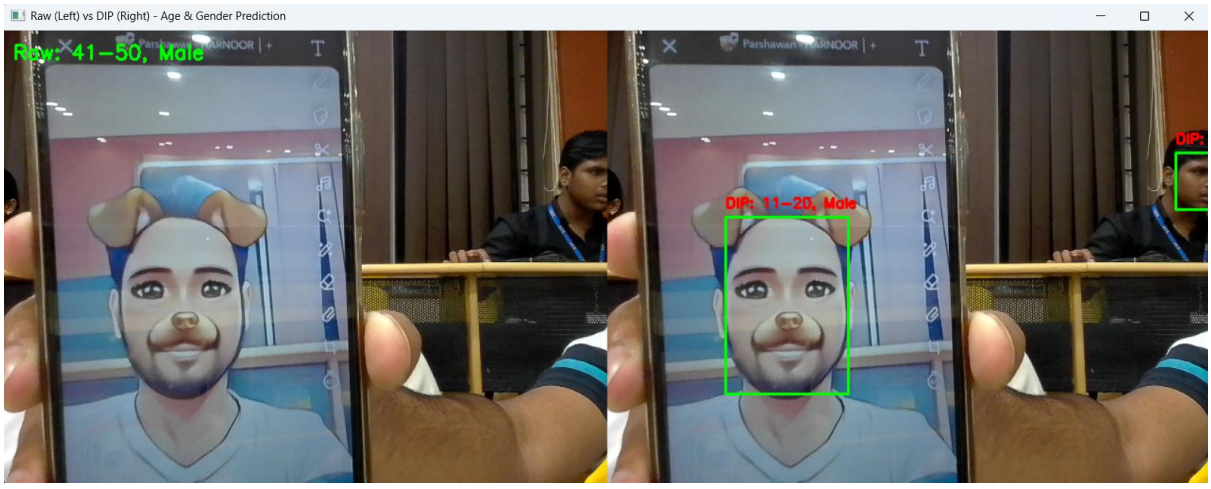

Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 41–50, Male
DIP: 11–20, Male

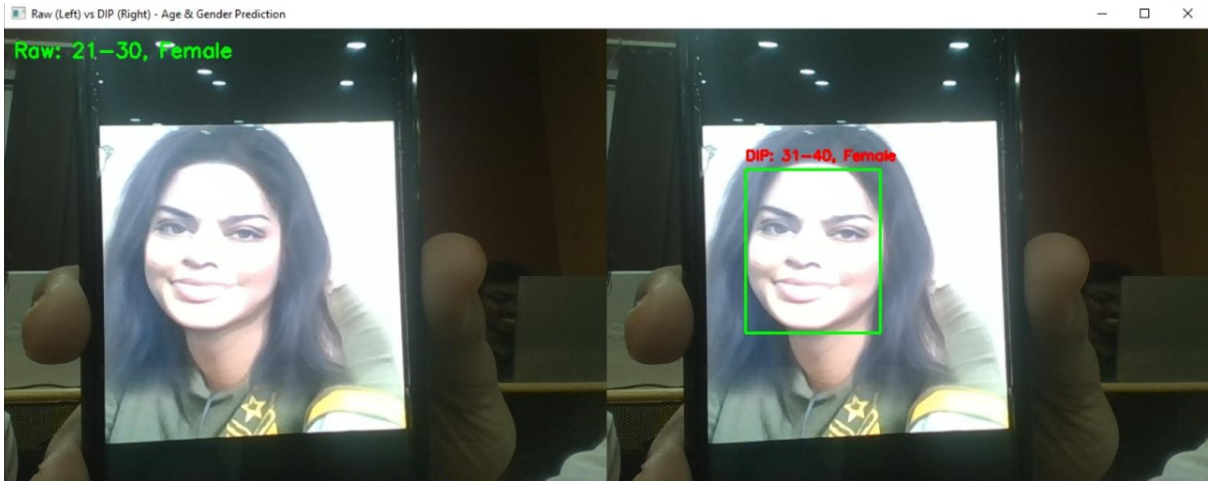
TC02 – Sketch

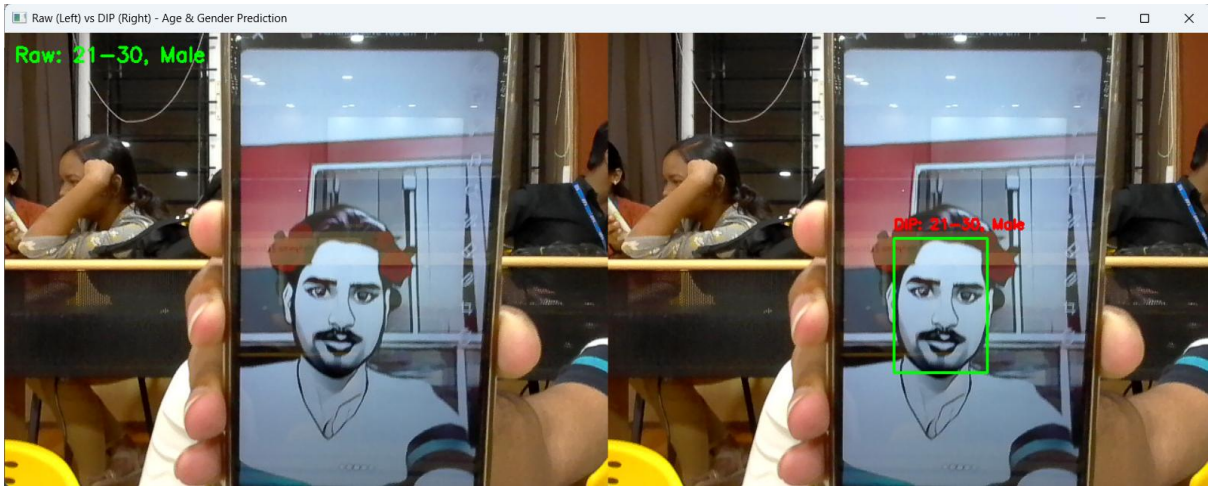

Raw: 11–20, Male
DIP: 31–36, Male

TC03 – Filters

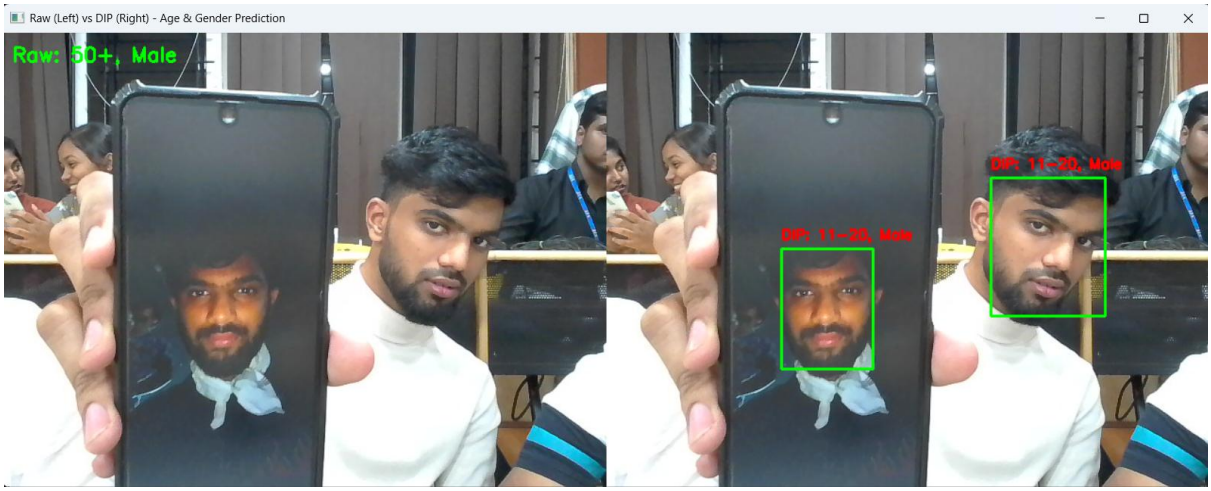


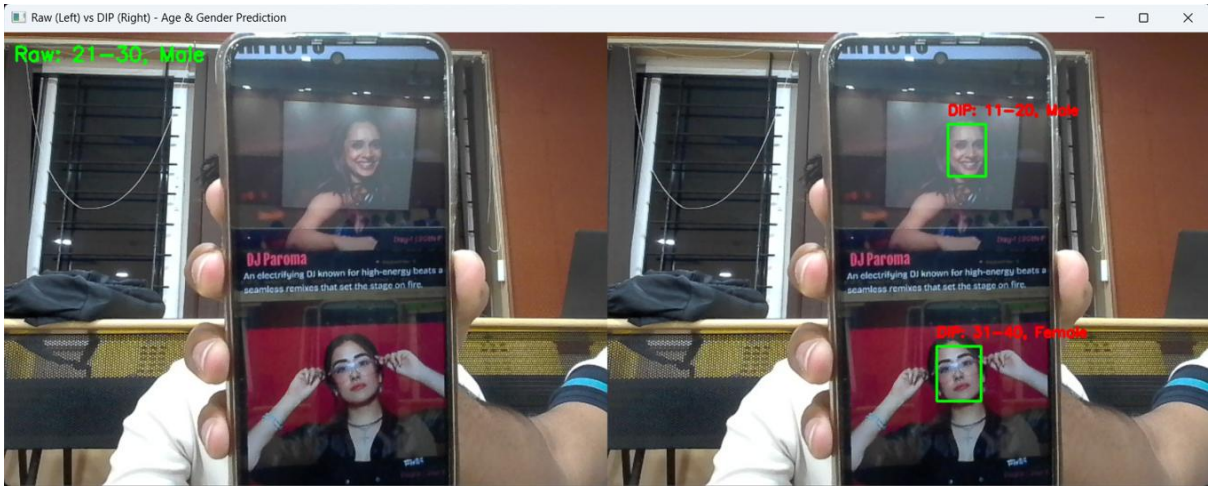




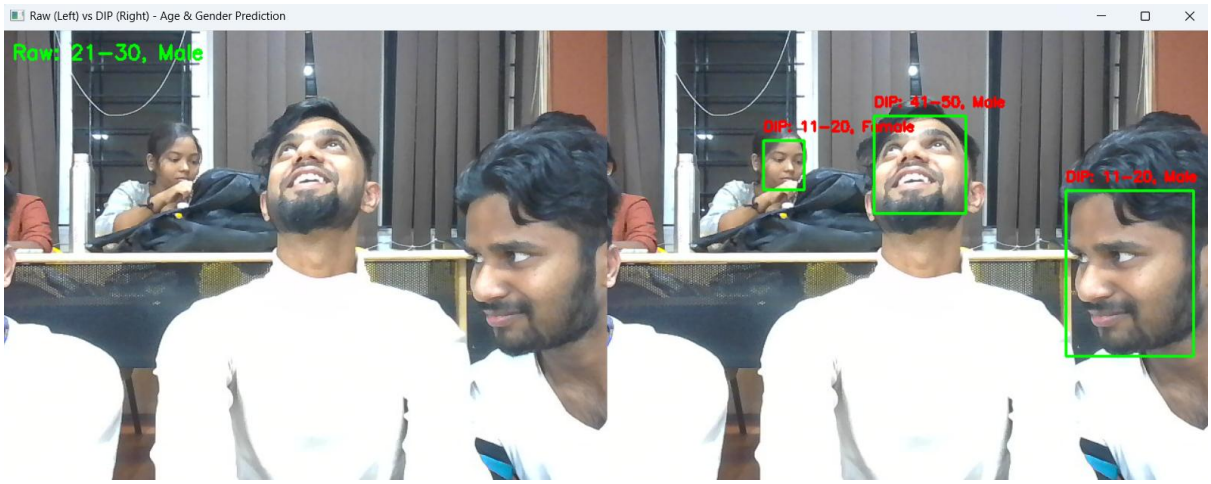


TC04 – Illumination Variation

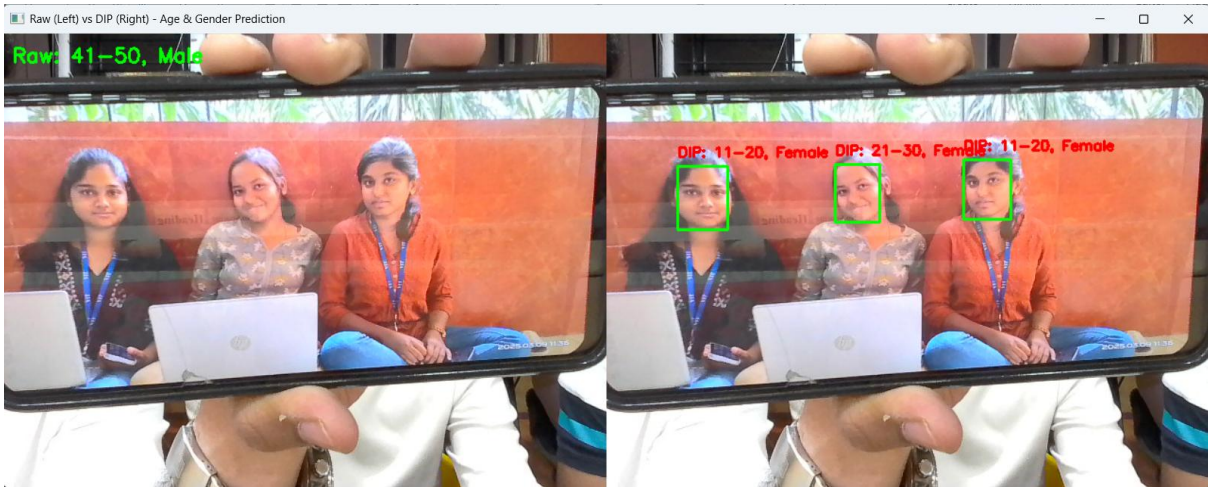


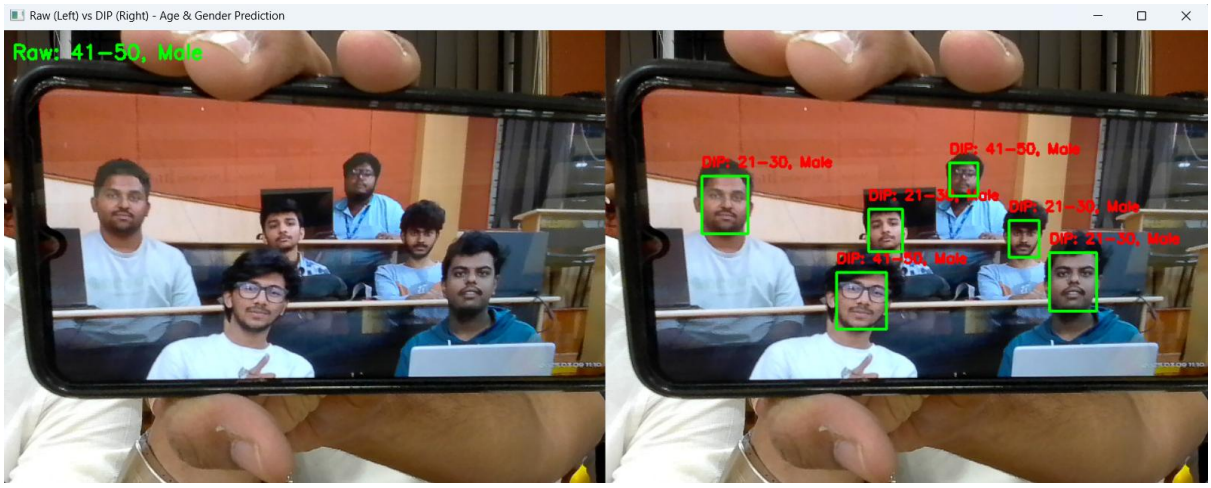


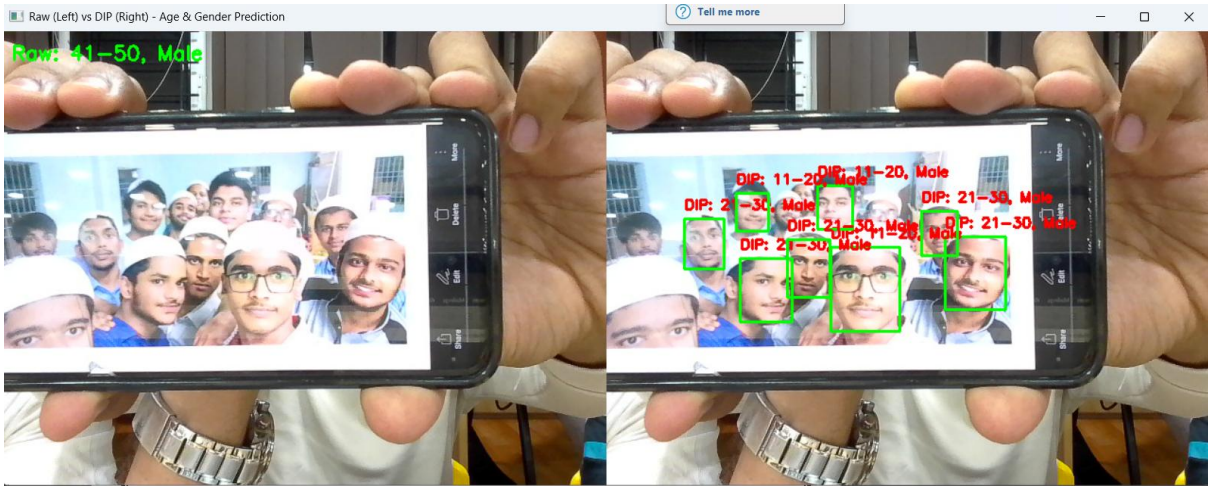
TC05 – Pose Variation:



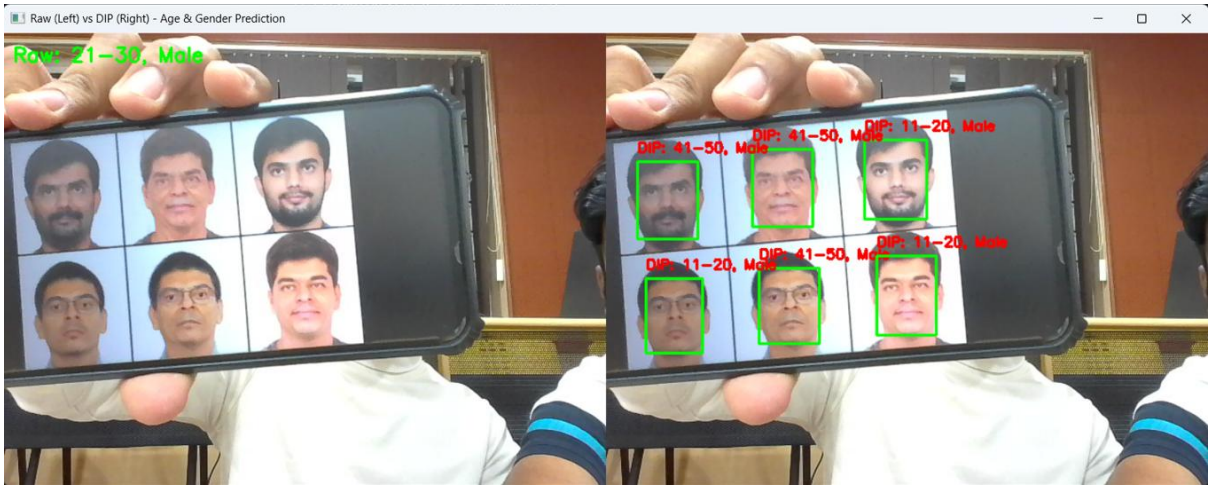
TC06 – Group Photo

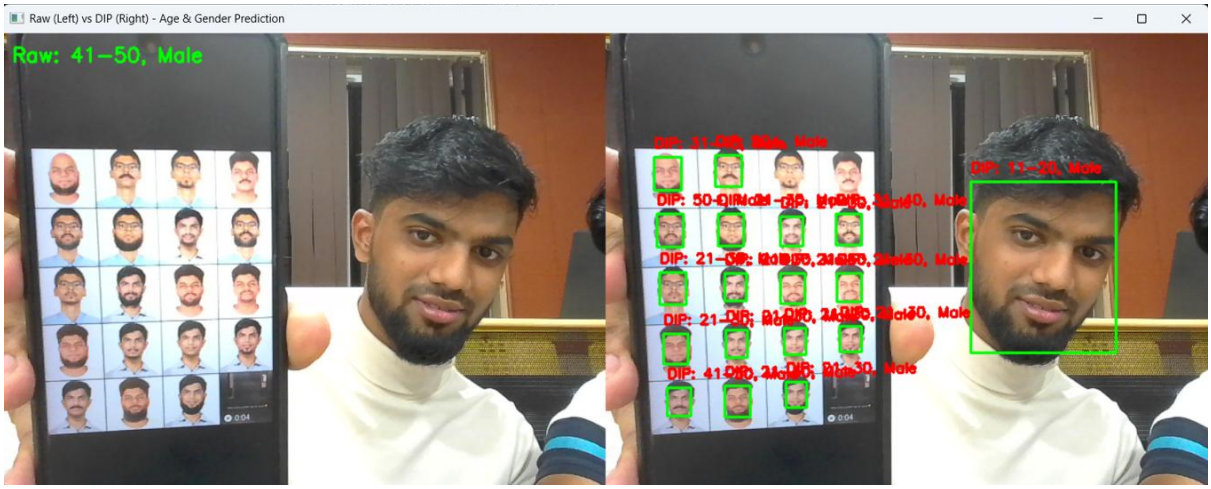


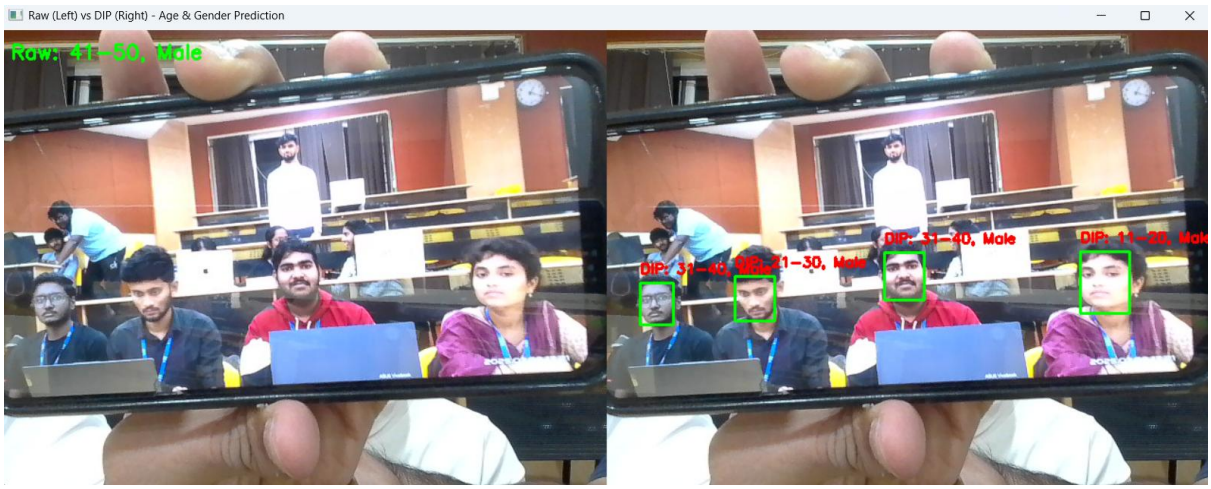


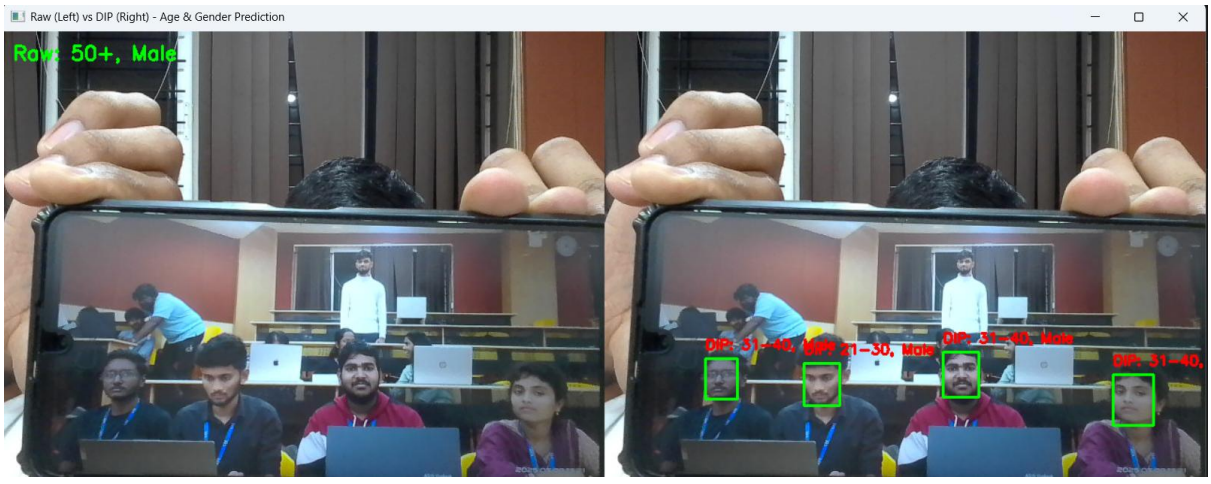

Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 41–50, Male
DIP: 11–20, Male
DIP: 21–30, Male
DIP: 41–50, Male


Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 11–20, Male
DIP: 11–20, Male

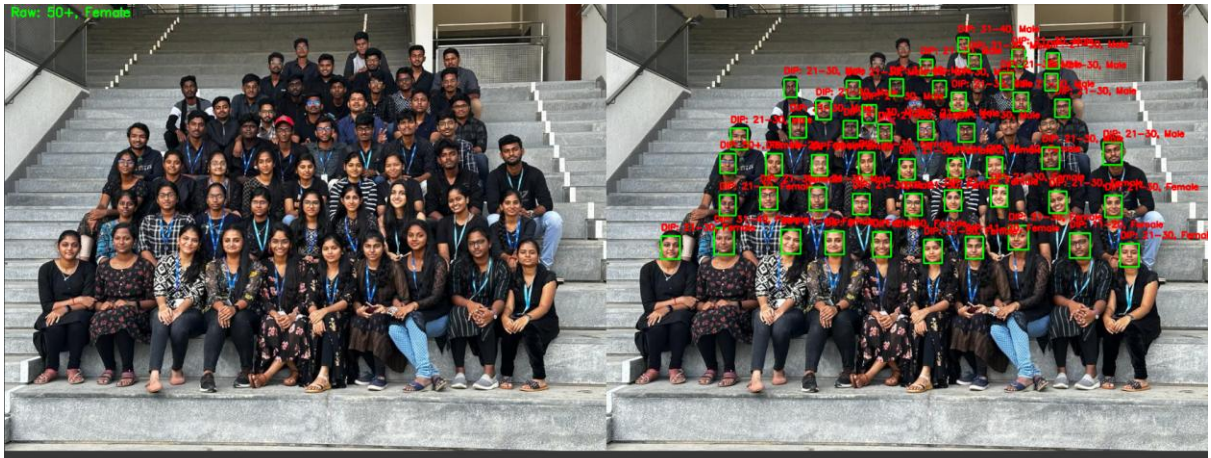

Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 21–30, Male
DIP: 41–50, Male
DIP: 11–20, Male


Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 41–50, Male

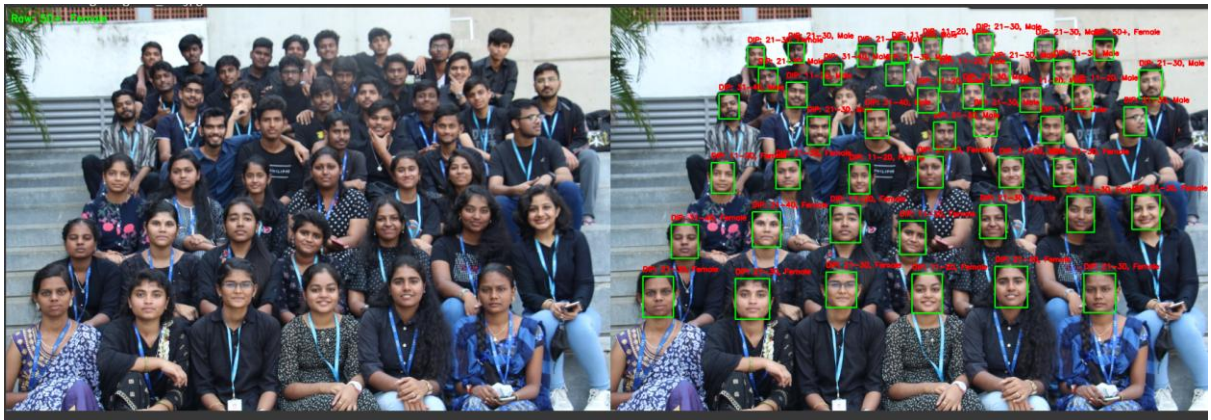

Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 41–50, Male


Raw (Left) vs DIP (Right) - Age & Gender Prediction
Raw: 50+, Male

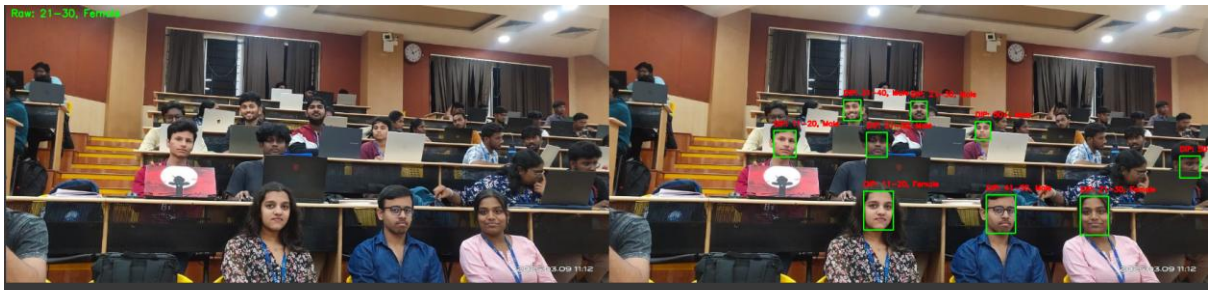
Positive Test cases



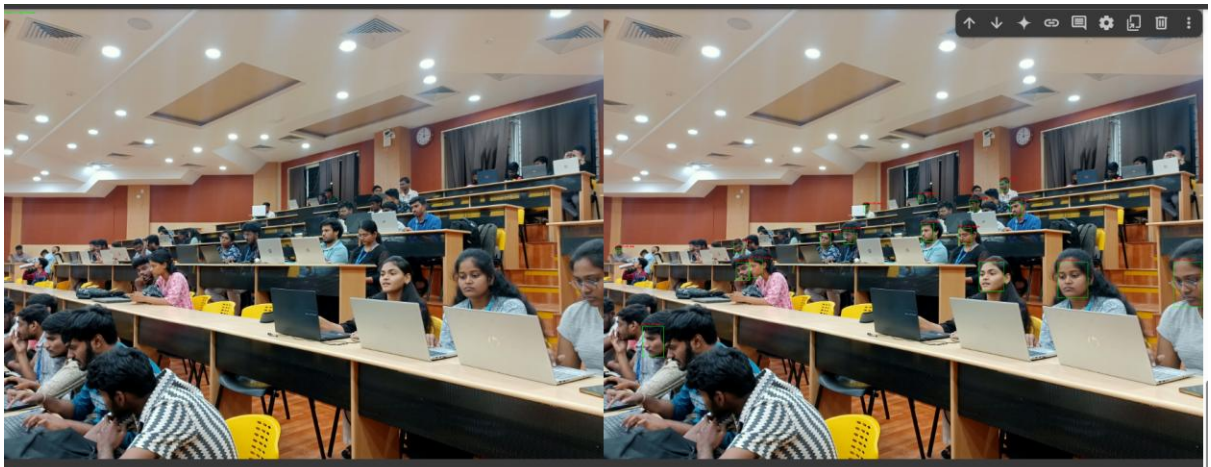

```
📊 **Summary for Image:**
50+ years: 3 people
31-40 years: 6 people
21-30 years: 39 people
11-20 years: 3 people
```



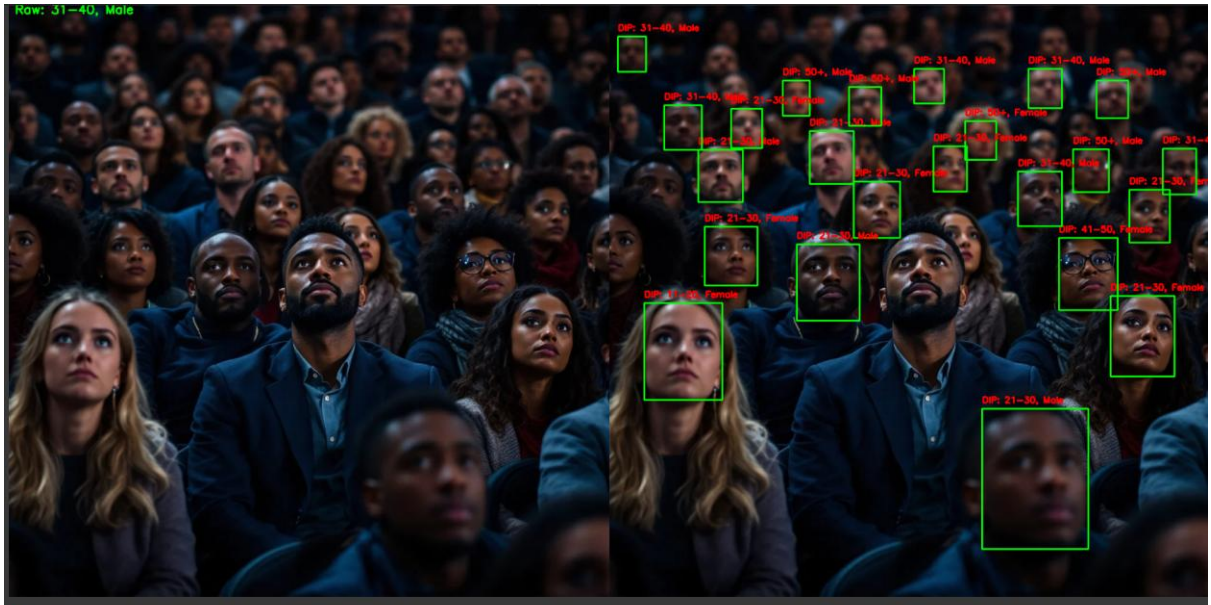
```
📊 **Summary for Image:**
50+ years: 2 people
21-30 years: 26 people
11-20 years: 14 people
31-40 years: 5 people
```



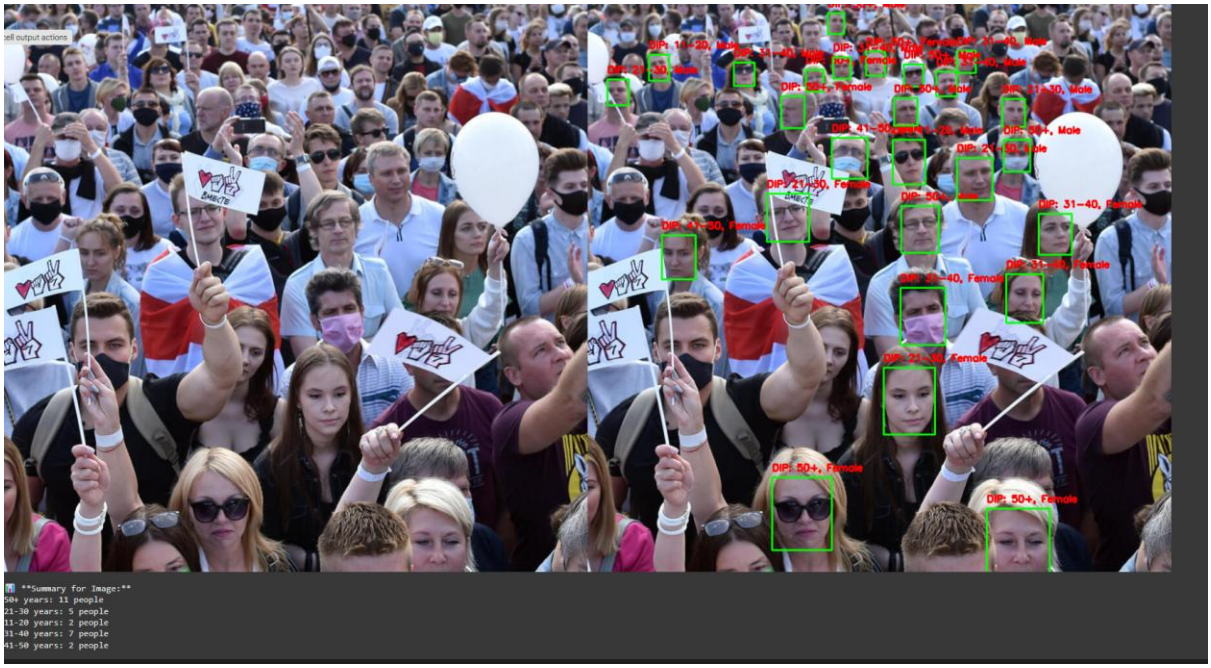

```
📊 **Summary for Image:**
21-30 years: 4 people
11-20 years: 2 people
41-50 years: 1 people
50+ years: 2 people
31-40 years: 1 people
```



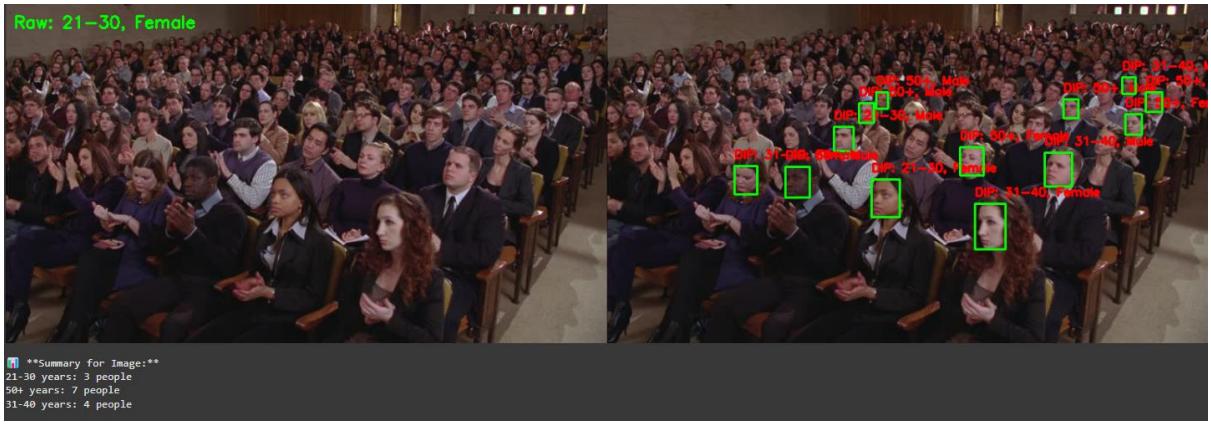
📊 **Summary for Image:**
11-20 years: 3 people
21-30 years: 6 people
41-50 years: 3 people
31-40 years: 6 people



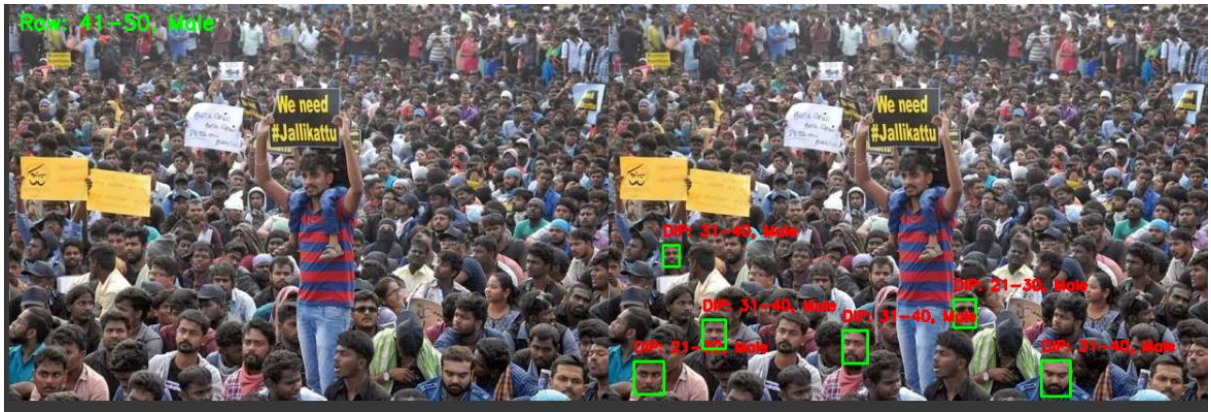
📊 **Summary for Image:**
31-40 years: 7 people
21-30 years: 10 people
11-20 years: 1 people
50+ years: 5 people
41-50 years: 1 people



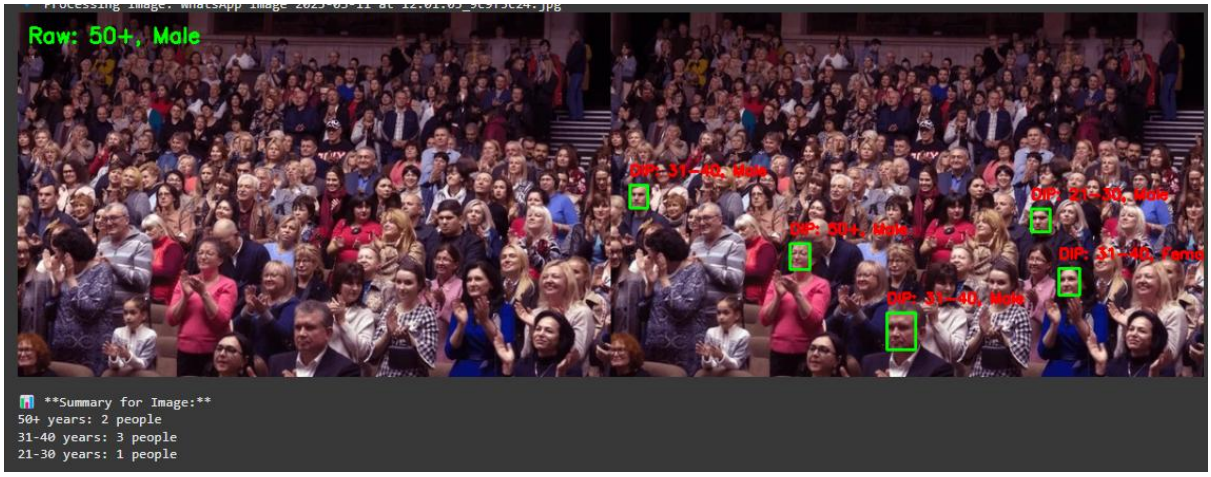
Negative Test cases

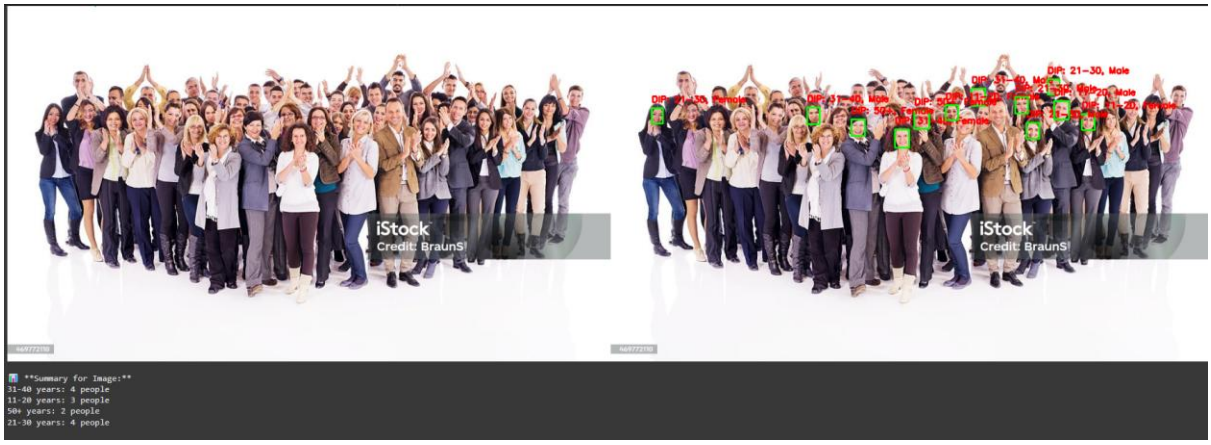


📊 **Summary for Image:**
41-50 years: 3 people
21-30 years: 8 people
31-40 years: 2 people

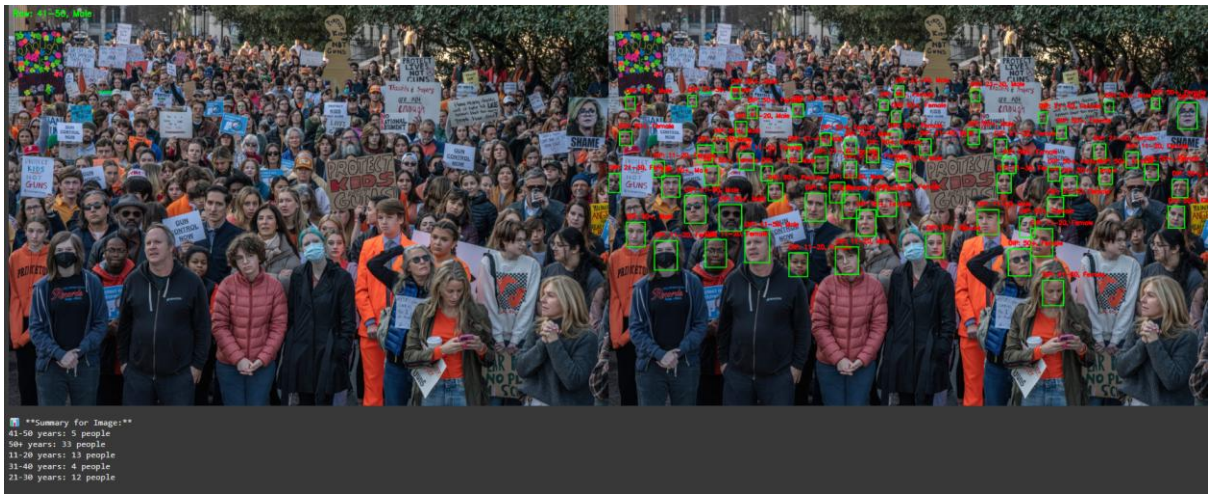


📊 **Summary for Image:**
41-50 years: 1 people
21-30 years: 2 people
31-40 years: 4 people





Challenging Test cases



```
📊 **Summary for Image:**
21-30 years: 19 people
31-40 years: 8 people
50+ years: 2 people
11-20 years: 1 people
0-10 years: 2 people
```


```
📊 **Summary for Image:**
50+ years: 17 people
21-30 years: 27 people
41-50 years: 6 people
31-40 years: 9 people
0-10 years: 1 people
11-20 years: 1 people
```


```
📊 **Summary for Image:**
41-50 years: 1 people
50+ years: 8 people
21-30 years: 8 people
11-20 years: 9 people
31-40 years: 6 people
```


```
📊 **Summary for Image:**
31-40 years: 5 people
50+ years: 11 people
21-30 years: 4 people
41-50 years: 1 people
11-20 years: 1 people
```

Video Test Case

https://drive.google.com/file/d/1_Z1aRR00hgLwiCH-n5-Srs-GBSYM1B1k/view?usp=drivesdk

1. Gender Distribution Analysis

- If the count is skewed towards **males**, it's expected as Mecca sees large groups of **male pilgrims, scholars, and worshippers**.
- If the number of **females is also significant**, it may indicate families, international pilgrims, or organized groups coming together.
- If there's a **balanced ratio**, it might suggest **Umrah season or a family-friendly time** rather than peak Hajj, where men often outnumber women in public spaces.

2. Age Group Patterns

- **High number of elderly (50+)**
    - Indicates pilgrims, especially **Hajj or Umrah visitors**.
    - Many elders visit Mecca as a **lifelong spiritual goal**, often coming in groups.
- **Young adults (18-40) in large numbers**
    - Likely indicates a mix of local Saudis, workers, and young international visitors.
    - This is common outside Hajj season when younger travelers visit for shorter spiritual trips.
- **Children & teenagers detected**
    - Suggests **family travel** (common during Umrah and non-peak religious visits).
    - Could indicate **local Saudi families or international families** staying for worship & sightseeing.

3. Cultural & Religious Insight

- Since **Mecca is a religious hub**, the presence of diverse age groups and genders could reflect the **global nature of Islamic pilgrimage**.
- If the dataset shows **mostly men and few women**, it could be from an area near the Haram where men traditionally gather (e.g., prayer lines, scholarly discussions).
- A **high diversity in age and gender** suggests a **non-restricted area** like the outer Haram courtyard, markets, or hotels.

13. Essential Links

13.1. Dataset Link (UTKFace)

https://www.kaggle.com/datasets/jangedoo/utkface-new

13.2. Google Colab Code Link

https://colab.research.google.com/drive/1gems3ffsOroOuZ5mFdxzKi4eYwF5bDY4?usp=sharing

14. Conclusions

The age and gender prediction system developed using the UTKFace dataset in a Google Colab environment represents a robust framework for analyzing crowd demographics through advanced computer vision and machine learning techniques. Leveraging the **YuNet face detection** model, **CLIP (clip-vit-large-patch14)** feature extraction, and a suite of classifiers (**MLP, Gaussian Naive Bayes, Random Forest, AdaBoost**), the system achieves impressive performance, with approximately **95%** accuracy for gender prediction and **67% accuracy** for age categorization. This capability enables detailed profiling of crowd compositions, supporting applications such as event monitoring, security, and demographic studies.

The modular architecture, encompassing data ingestion, face detection, preprocessing (raw and DIP modes), feature extraction, prediction, real-time execution, and data visualization, provides a scalable and interpretable pipeline. Key strengths include the high reliability of gender classification, the flexibility of dual preprocessing paths, and the insightful visualizations of dataset distributions, training samples, and preprocessing impacts. The integration of Grad-CAM++ with YuNet and the analysis of CLIP feature distributions across ages further enhance the system's interpretability, offering valuable insights into model behavior and feature relevance for crowd analysis.

Despite these achievements, the system faces challenges, particularly in age prediction accuracy, which is limited by dataset imbalances (e.g., fewer samples in the "41-50" bin) and overlapping features in middle age groups. The minimal improvement from DIP preprocessing suggests that current enhancements may not fully address these complexities, while the computational cost of clip-vit-large-patch14 and 384x384 processing constrains real-time scalability. The batch-based real-time execution, while effective for test data, lacks the continuous processing needed for live video streams.

Looking forward, the system can be enhanced by addressing dataset imbalances through augmentation, fine-tuning CLIP on UTKFace-specific data, and optimizing for lighter models or native input sizes (e.g., 224x224) to improve efficiency. Incorporating interactive visualizations, extending to video streaming, and validating with diverse real-world datasets could further broaden its applicability. Overall, this system lays a solid foundation for crowd demographic analysis, with significant potential for refinement and real-world deployment in dynamic crowd management scenarios.

Rationale

- **Key Findings**: Highlights the 95% gender and 67% age accuracy, tying back to Sections 11 and 12, and emphasizes the modular design from the architecture (Section 13).
- **Strengths**: Acknowledges the reliability of gender prediction, dual preprocessing, and interpretability tools (Sections 7, 8, 10), linking to Grad-CAM++ and CLIP feature analysis.
- **Limitations**: Addresses age accuracy challenges, preprocessing impact, and computational constraints (Sections 12, 13), aligning with observed data.
- **Future Directions**: Suggests practical improvements (e.g., augmentation, optimization) based on limitations and the system's potential (Sections 7-13), encouraging further development.

15.References:

[1] Prasher, S., Nelson, L., & Arumugam, D. (2024, May). Deep Learning Models for Age and Gender Prediction using Facial Images. In *2024 5th International Conference for Emerging Technology (INCET)* (pp. 1-5). IEEE.

[2] Zhao, Q., Liu, J., & Wei, W. (2024). Mixture of deep networks for facial age estimation. *Information Sciences*, *679*, 121086.

[3] Bao, Z., Luo, Y., Tan, Z., Wan, J., Ma, X., & Lei, Z. (2023). Deep domain-invariant learning for facial age estimation. *Neurocomputing*, *534*, 86-93.

[4] Roopak, M., Khan, S., Parkinson, S., & Armitage, R. (2023). Comparison of deep learning classification models for facial image age estimation in digital forensic investigations. *Forensic Science International: Digital Investigation*, *47*, 301637.

[5] Amirullaeva, S., & Han, J. H. (2024). Relative Age Position Learning for Face-based Age Estimation. *IEEE Access*.

[6] Kalpana, R., Deepika, D., Kavya, A., Himabindhu, P., & Kethavi, S. (2024, April). Facial Age and Gender Prediction using Deep Learning. In *2024 10th International Conference on Communication and Signal Processing (ICCSP)* (pp. 1571-1576). IEEE.

[7] AlQadi, R. A., & Batouche, M. (2023, January). Application of Advanced Deep Learning Techniques for Face Detection and Age Estimation. In *2023 1st International Conference on Advanced Innovations in Smart Cities (ICAISC)* (pp. 1-6). IEEE.

[8] Pallavi, M. O., Vishwanath, Y., & Raj, A. (2023, January). Deep Learning Based Application in Detecting Wrinkle and Predicting Age. In *2023 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE)* (pp. 1168-1173). IEEE.

[9] Dey, P., Mahmud, T., Chowdhury, M. S., Hossain, M. S., & Andersson, K. (2024). Human age and gender prediction from facial images using deep learning methods. *Procedia Computer Science*, *238*, 314-321.

[10] Guehairia, O., Dornaika, F., Ouamane, A., & Taleb-Ahmed, A. (2022). Facial age estimation using tensor based subspace learning and deep random forests. *Information Sciences*, *609*, 1309-1317.

[11] Zhang, Y., Shou, Y., Meng, T., Ai, W., & Li, K. (2024). A multi-view mask contrastive learning graph convolutional neural network for age estimation. *Knowledge and Information Systems*, *66*(11), 7137-7162.

[12] Yang, M., Yao, C., & Yan, S. (2023, September). Age Estimation Based on Graph Convolutional Networks and Multi-head Attention Mechanisms. In *2023 IEEE 6th International Conference on Information Systems and Computer Aided Education (ICISCAE)* (pp. 1124-1129). IEEE.

[13] Cao, Z., Zhang, K., Pang, L., & Zhao, H. (2022). A Demographic Attribute Guided Approach to Age Estimation. *arXiv preprint arXiv:2205.10254*.

[14] ELKarazle, K., Raman, V., & Then, P. (2022). Facial age estimation using machine learning techniques: An overview. *Big Data and Cognitive Computing*, *6*(4), 128.

[15] Agbo-Ajala, O., Viriri, S., Oloko-Oba, M., Ekundayo, O., & Heymann, R. (2022). Apparent age prediction from faces: A survey of modern approaches. *Frontiers in big Data*, *5*, 1025806.

[16] Park, S. R., Park, H., Lee, S., Hwang, J., Suh, B. F., & Kim, E. (2024). Facial age evaluated by artificial intelligence system, Dr. AMORE®: An objective, intuitive, and reliable new skin diagnosis technology. *Journal of Cosmetic Dermatology*, *23*(4).

[17] Zhang, H., Lin, J., Zhou, L., Shen, J., & Sheng, W. (2024). Facial age recognition based on deep manifold learning. *Mathematical Biosciences and Engineering*, *21*(3), 4485-4500.

[18] Tatikonda, S., Nambiar, A., & Mittal, A. (2022, June). Face age progression with attribute manipulation. In *International Conference on Pattern Recognition and Artificial Intelligence* (pp. 639-652). Cham: Springer International Publishing.

[19] Shah, R., & Ogden, J. (2006). ‘What's in a face?’The role of doctor ethnicity, age and gender in the formation of patients’ judgements: an experimental study. *Patient education and counseling*, *60*(2), 136-141.

[20] Scottish Sun. (2024). The Facial Feature That Could Mean You're 2.5 Times More Likely to Develop Dementia. *The Scottish Sun*.

[21] Zhang, K., Gao, C., Guo, L., Sun, M., Yuan, X., Han, T. X., ... & Li, B. (2017). Age group and gender estimation in the wild with deep RoR architecture. *IEEE Access*, *5*, 22492-22503.

[22] Garain, A., Ray, B., Singh, P. K., Ahmadian, A., Senu, N., & Sarkar, R. (2021). GRA_Net: A deep learning model for classification of age and gender from facial images. *IEEE Access*, *9*, 85672-85689.

[23] Amirullaeva, S., & Han, J. H. (2024). Relative Age Position Learning for Face-based Age Estimation. *IEEE Access*.

[24] Lou, Z., Alnajar, F., Alvarez, J. M., Hu, N., & Gevers, T. (2017). Expression-invariant age estimation using structured learning. *IEEE transactions on pattern analysis and machine intelligence*, *40*(2), 365-375.

[25] Shi, C., Zhao, S., Zhang, K., & Feng, X. (2023). Multi-task multi-scale attention learning-based facial age estimation. *IET Signal Processing*, *17*(2), e12190.

[26] Chen, C., Dantcheva, A., & Ross, A. (2014, January). Impact of facial cosmetics on automatic gender and age estimation algorithms. In *2014 International Conference on Computer Vision Theory and Applications (VISAPP)* (Vol. 2, pp. 182-190). IEEE.

[27] Agrawal, B., & Dixit, M. (2019, November). Age estimation and gender prediction using convolutional neural network. In *International Conference on Sustainable and Innovative Solutions for Current Challenges in Engineering & Technology* (pp. 163-175). Cham: Springer International Publishing.

[28] Cruz, C., Della Rosa, M., Krueger, C., Gao, Q., Horkai, D., King, M., ... & Houseley, J. (2018). Tri-methylation of histone H3 lysine 4 facilitates gene expression in ageing cells. *Elife*, *7*, e34081.

[29] Wang, L., Zhang, X., Chen, P., & Zhou, D. (2024). Doctor simulator: Delta-Age-Sex-AdaIn enhancing bone age assessment through AdaIn style transfer. *Pediatric Radiology*, *54*(10), 1704-1712.

[30] Wang, S., Jin, S., Xu, K., She, J., Fan, J., He, M., ... & Yao, K. (2024). A pediatric bone age assessment method for hand bone X-ray images based on dual-path network. *Neural Computing and Applications*, *36*(17), 9737-9752.