Gender and Age Detection: Predict if a Person is a Male or Female and Also Their Age

Members:

Names	Roll no
Sanjita Jain	**53**
Rupprashik Khare	**54**
Samyak Lahire	**55**
Kalpesh Patil	**56**

Executive Summary

This report presents a detailed analysis of a computer vision application that detects human faces in images or video streams and predicts the gender and age range of detected individuals. The system utilizes pre-trained deep learning models with OpenCV's DNN (Deep Neural Network) module to perform three key tasks: face detection, gender classification, and age estimation. The implementation demonstrates how multiple neural networks can be effectively cascaded to create a comprehensive demographic analysis tool applicable in various domains including security, marketing analytics, and human-computer interaction.

1. Introduction

Demographic analysis through computer vision has become increasingly valuable across multiple industries. This application employs a pipeline approach that processes images or video streams to extract meaningful demographic information without requiring explicit user input beyond providing the visual data.

The system's architecture consists of three neural networks working in sequence:

1. A face detection network that locates and extracts faces from images
2. A gender classification network that categorizes detected faces as male or female
3. An age estimation network that assigns detected faces to predefined age range categories

2. Technical Components

2.1 Required Libraries and Dependencies

The application relies on two key Python libraries:

- **OpenCV (cv2)**: Provides the computer vision algorithms and deep learning interface
- **Math**: Supports mathematical operations required for image processing
- **Argparse**: Enables command-line argument parsing for flexible application execution

2.2 Pre-trained Models

The implementation requires six model files:

Purpose	Model Type	Files
Face Detection	Caffe-based SSD	opencv_face_detector.pbtxt, opencv_face_detector_uint8.pb
Gender Classification	Caffe model	gender_deploy.prototxt, gender_net.caffemodel
Age Estimation	Caffe model	age_deploy.prototxt, age_net.caffemodel

These models must be available in the execution directory for the application to function correctly.

2.3 Classification Categories

The system categorizes detected individuals into predefined groups:

Gender Classification:

- Male
- Female

Age Range Estimation:

- (0-2) years
- (4-6) years
- (8-12) years
- (15-20) years
- (25-32) years
- (38-43) years
- (48-53) years
- (60-100) years

3. Implementation Details

3.1 Face Detection Function (highlightFace)

1. **Input Preparation**:
   - Creates a copy of the input frame
   - Extracts frame dimensions (height and width)
   - Converts the frame to a blob, which normalizes, resizes, and prepares the image for the neural network
2. **Face Detection**:
   - Passes the blob through the face detection network
   - Processes detection results to extract bounding box coordinates
   - Filters detections based on confidence threshold (default 0.7)
3. **Visual Annotation**:
   - Draws green rectangles around detected faces
   - The rectangle thickness scales proportionally to the frame height
4. **Output**:
   - Returns the annotated frame and a list of face bounding boxes for further processing

3.2 Main Processing Pipeline

The main execution flow follows these steps:

1. **Argument Parsing**:
   - Accepts an optional --image parameter to specify an image file path
   - If no image is provided, defaults to using the webcam (device 0)
2. **Model Loading**:
   - Loads the face detection model
   - Loads the gender classification model
   - Loads the age estimation model
3. **Continuous Frame Processing**:
   - Captures frames from either webcam or specified image file
   - Applies face detection to each frame

    - For each detected face:
        - Extracts the face region with additional padding
        - Prepares the face image for gender and age networks
        - Performs gender classification
        - Performs age estimation
        - Annotates the original frame with predicted gender and age
        - Displays the result in a window

4. **User Interface**:
    - Displays processed frames in real-time
    - Continues until a key is pressed or the video/image stream ends

4. Technical Implementation Analysis

4.1 Face Detection

The application uses a Single Shot Multibox Detector (SSD) with a ResNet base network, which offers an effective balance between accuracy and speed. Key aspects include:

- **Input Transformation**: Resizes images to 300x300 pixels and normalizes pixel values using mean subtraction [104, 117, 123]
- **Confidence Filtering**: Only processes faces with detection confidence above 0.7 (configurable)
- **Bounding Box Scaling**: Converts normalized coordinates (0-1) to pixel coordinates based on frame dimensions

4.2 Face Preprocessing

Before gender and age prediction, faces undergo additional processing:

- **Region Extraction**: Crops the detected face with additional padding (20 pixels) to include contextual features
- **Boundary Handling**: Ensures extracted regions remain within frame boundaries
- **Size Normalization**: Resizes face regions to 227x227 pixels for compatibility with gender and age networks

4.3 Gender and Age Prediction

Both gender and age predictions follow similar workflows:

1. **Blob Creation**: Converts face images to blobs using specified mean values (78.4, 87.8, 114.9)
2. **Network Forward Pass**: Processes the blob through respective networks
3. **Prediction Extraction**: Determines the prediction by finding the index with maximum probability
4. **Result Mapping**: Maps the prediction index to human-readable categories

6. Implementation Requirements

To run this application, the following components are required:

6.1 Software Dependencies

- Python 3.x
- OpenCV (cv2) library with DNN module support
- Required model files in the execution directory

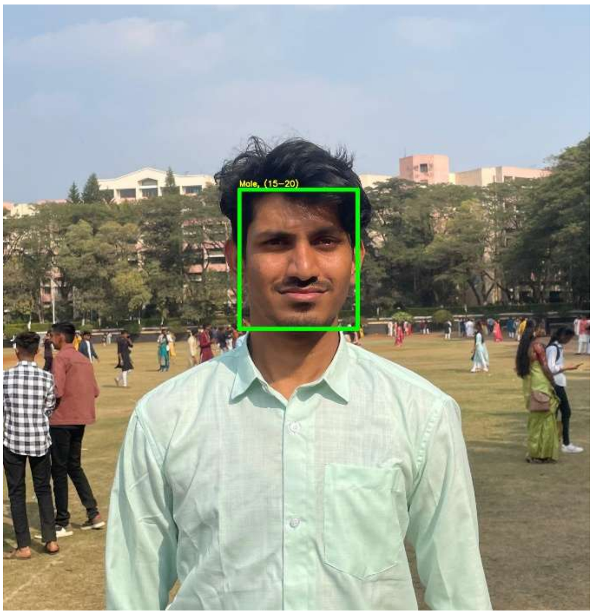
6.2 Hardware Recommendations

- Computer with webcam (for live video processing)
- Sufficient RAM for model loading and inference
- GPU support recommended for real-time processing
- Adequate storage for model files (approximately 50MB total)

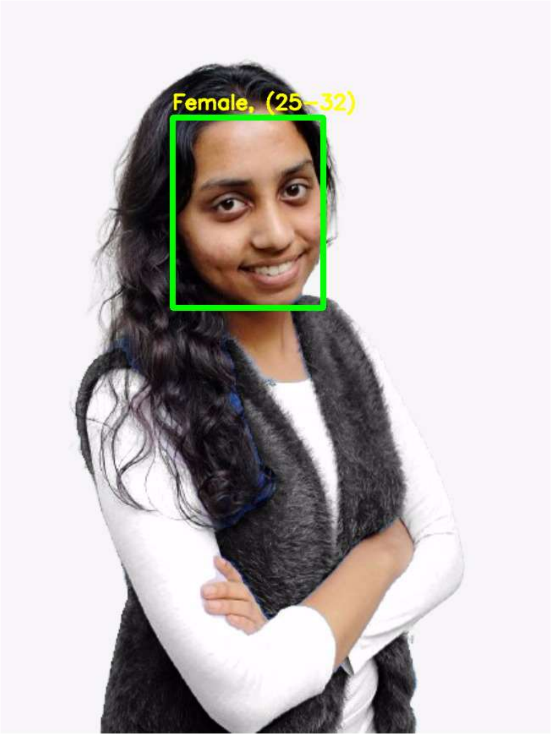
6.3 Execution Instructions

1. Install required dependencies: pip install opencv-python
2. Download required model files to the application directory
3. Run with webcam: python gender_age_detection.py
4. Run with image file: python gender_age_detection.py --image path_to_image.jpg

7. Results



```
(venv) ➜  Gender-and-Age-Detection git:(master) ✗ python3 detect.py --image kalpesh.png
Gender: Male
Age: 15-20 years
```



```
(venv) ➜  Gender-and-Age-Detection python3 detect.py --image sanjita.jpg
Gender: Female
```

Age: 25-32 years

8. Conclusion

This gender and age detection system demonstrates the effective application of deep learning models for demographic analysis through computer vision. By combining face detection, gender classification, and age estimation in a sequential pipeline, the system provides valuable demographic insights from visual data.

While the implementation has limitations in terms of prediction granularity and demographic categories, it provides a solid foundation for applications across multiple domains. With appropriate ethical considerations and potential enhancements, this technology can be adapted for various practical applications where demographic information provides value.

The modular design of the system also allows for future improvements through model updates or architectural changes without requiring a complete redesign, making it an adaptable solution for evolving requirements in computer vision applications.