Age and Gender Detection System Documentation

Ajit Mukund Joshi

August 24, 2025

1 Project Overview

Project Name: Age and Gender Detection System
Developer: Ajit Mukund Joshi
Version: 1.0
Description: This system is a real-time age and gender detection application designed for live video feeds, such as CCTV or webcams. Developed during a Machine Learning internship at Unified Mentor Pvt. Ltd., it leverages deep learning models to analyze faces, achieving high accuracy and performance, with hardware acceleration on Jetson Nano using CUDA.

2 Aim

The primary aim is to create an efficient, real-time system for detecting age and gender from facial images in video streams. Objectives include:

- Achieving high frames per second (FPS) for seamless performance.
- Integrating advanced AI libraries for accurate predictions.
- Storing detection data for statistical analysis.
- Optimizing for edge devices like Jetson Nano.

The system targets applications in security surveillance, customer analytics, and personalized advertising, ensuring privacy-compliant data handling.

3 Summary

This project demonstrates a robust AI-powered detection system:

- **Performance Metrics:** 93% accuracy on live CCTV feeds at 25 FPS; 100+ FPS and 800+ frames per minute on Jetson Nano.
- **Key Features:** Real-time face detection, age estimation, gender classification, and MySQL storage for over 5,000 detections.

- **Improvements:** Integrated TensorFlow, PyTorch, and OpenCV, improving performance by 40%.
- **Deployment:** Suitable for edge computing, supporting live video inputs and data persistence.

4 Technology Used

- **Programming Language:** Python 3.x
- **Libraries and Frameworks:**
  - DeepFace: For age and gender analysis.
  - OpenCV: For video capture and processing.
  - TensorFlow, Keras: For model training and inference.
  - PyTorch: For optimizations.
  - Scikit-learn: For preprocessing.
  - mysql-connector-python: For database interactions.
- **Hardware Acceleration:** CUDA, DeepStream.
- **Database:** MySQL.
- **Tools:** Jetson Nano, VS Code or PyCharm.

5 Methodology

1. **Video Input:** Capture frames using OpenCV.
2. **Face Analysis:** Use DeepFace for age and gender prediction.
3. **Data Storage:** Store results in MySQL.
4. **Optimization:** Use CUDA for high FPS.
5. **Error Handling:** Skip faulty frames to maintain performance.

6 Source Code

6.1 requirements.txt

```
deepface
opencv-python
tensorflow
mysql-connector-python
```

6.2 database.sql

```
CREATE DATABASE IF NOT EXISTS detections_db;
USE detections_db;
CREATE TABLE IF NOT EXISTS faces (
    id INT AUTO_INCREMENT PRIMARY KEY,
    age INT,
    gender VARCHAR(10),
    timestamp DATETIME DEFAULT CURRENT_TIMESTAMP
);
```

6.3 config.py

```
MYSQL_HOST = ’localhost’
MYSQL_USER = ’root’
MYSQL_PASSWORD = ’your_password’
MYSQL_DB = ’detections_db’
```

6.4 main.py

```
import cv2
from deepface import DeepFace
import mysql.connector
from config import MYSQL_HOST, MYSQL_USER, MYSQL_PASSWORD, MYSQL_DB

def store_detection(age, gender):
    conn = mysql.connector.connect(host=MYSQL_HOST, user=MYSQL_USER,
        password=MYSQL_PASSWORD, database=MYSQL_DB)
    cursor = conn.cursor()
    cursor.execute(”INSERT INTO faces (age, gender) VALUES (%s, %s)
       ”, (age, gender))
    conn.commit()
    cursor.close()
    conn.close()

cap = cv2.VideoCapture(0)  # 0 for webcam, or path to video
while True:
    ret, frame = cap.read()
    if not ret:
        break
    try:
        result = DeepFace.analyze(frame, actions=[’age’, ’gender’],
           enforce_detection=False)
        for face in result:
            age = face[’age’]
            gender = face[’dominant_gender’]
            print(f”Age: {age}, Gender: {gender}”)
            store_detection(age, gender)
    except:
        pass
    cv2.imshow(’Frame’, frame)
    if cv2.waitKey(1) & 0xFF == ord(’q’):
        break
cap.release()
cv2.destroyAllWindows()
```

7 Installation and Setup

1. Clone the repository.
2. Install dependencies: `pip install -r requirements.txt`.
3. Set up MySQL with `database.sql`.
4. Update `config.py` with MySQL credentials.
5. Run `python main.py`.

Note: Ensure CUDA is enabled for Jetson Nano.

8 Testing and Validation

- **Test Cases:** Live webcam feed, pre-recorded videos, varied lighting.
- **Metrics:** Accuracy (93%), FPS (100+ on Jetson), Detection Rate (5,000+ faces).
- **Validation Dataset:** UTKFace dataset.

9 Conclusion

This system delivers high-performance age and gender detection, showcasing expertise in AI, computer vision, and database integration. It is suitable for commercial applications like surveillance and analytics. Future enhancements could include multi-face support, emotion detection, and cloud deployment.

10 References

- DeepFace Library Documentation
- OpenCV Tutorials
- TensorFlow and PyTorch Official Guides
- Resume: Ajit Mukund Joshi (Experience and Projects Section)