The Superior University

Project Title

Detect Gender and Age using Artificial Intelligence

Name: Muhammad Rizwan

Roll No: BSAI(3C)

Project Details

Course: AI Lab

Instructor: Rasikh sahib

Semester: 3rd

Section: BSAI3C

Abstract

The "Detect Gender and Age using Artificial Intelligence" project leverages computer vision and deep learning techniques to analyze facial features from images or video streams. By utilizing pre-trained models in OpenCV, the project identifies a person’s gender and estimates their age range with high confidence. This system demonstrates the practical application of AI in real-time human attribute recognition.

Table of Contents

1. Introduction

2. Objectives

3. System Requirements

4. Methodology

5. Implementation

6. Challenges and Solutions

7. Conclusion

1. Introduction

The "Detect Gender and Age using Artificial Intelligence" project demonstrates the use of AI techniques in the field of computer vision. By employing pre-trained deep learning models, this project can detect faces in images or video streams, identify gender, and estimate age ranges. The solution is designed to showcase the potential of AI in real-time applications such as surveillance, customer profiling, and entertainment systems.

2. Objectives

- To build a system that detects faces in images or videos.

- To classify the gender of individuals using deep learning models.

- To estimate the age range of individuals based on facial features.

- To explore the practical applications of computer vision and AI techniques in real-world scenarios.

3. System Requirements

\*\*Hardware Requirements:\*\*

- Processor: Intel Core i5 or higher

- RAM: 8 GB or more

- Storage: 1 GB free space

- Camera (for live detection)

\*\*Software Requirements:\*\*

- Programming Language: Python 3.x

- Libraries: OpenCV, argparse

- Pre-trained Models: Caffe models for age and gender detection

- Operating System: Windows/Linux/MacOS

4. Methodology

\*\*Design Approach:\*\*

The system uses pre-trained Caffe models for detecting and analyzing faces. OpenCV's deep learning module (cv.dnn) is employed to load and infer results from these models.

\*\*Workflow:\*\*

- The input image or video frame is captured.

- Faces are detected using a pre-trained face detection model.

- For each detected face, gender and age predictions are made using the corresponding Caffe models.

- Results are displayed on the processed image with bounding boxes and labels.

\*\*Algorithm:\*\*

- Load pre-trained models for face, age, and gender detection.

- Preprocess the input image or video frame.

- Detect faces and extract facial regions.

- Classify gender and estimate age for each face.

- Overlay the results on the output image or video stream.

5. Implementation

\*\*Core Components:\*\*

- \*\*getFaceBox(net, frame, conf\_threshold):\*\* Detects faces in the input frame and returns bounding boxes.

- \*\*genderNet.forward():\*\* Predicts the gender of the detected face.

- \*\*ageNet.forward():\*\* Estimates the age range of the detected face.

- \*\*cv.putText():\*\* Overlays the results on the output frame.

\*\*Sample Code Snippet:\*\*

```python

def getFaceBox(net, frame, conf\_threshold=0.7):

frameOpencvDnn = frame.copy()

frameHeight = frameOpencvDnn.shape[0]

frameWidth = frameOpencvDnn.shape[1]

blob = cv.dnn.blobFromImage(frameOpencvDnn, 1.0, (300, 300), [104, 117, 123], True, False)

net.setInput(blob)

detections = net.forward()

bboxes = []

for i in range(detections.shape[2]):

confidence = detections[0, 0, i, 2]

if confidence > conf\_threshold:

x1 = int(detections[0, 0, i, 3] \* frameWidth)

y1 = int(detections[0, 0, i, 4] \* frameHeight)

x2 = int(detections[0, 0, i, 5] \* frameWidth)

y2 = int(detections[0, 0, i, 6] \* frameHeight)

bboxes.append([x1, y1, x2, y2])

cv.rectangle(frameOpencvDnn, (x1, y1), (x2, y2), (0, 255, 0), int(round(frameHeight/150)), 8)

return frameOpencvDnn, bboxes

```

6. Challenges and Solutions

\*\*Challenge:\*\* Real-time performance optimization.

\*\*Solution:\*\* Used efficient pre-trained Caffe models and optimized frame processing with OpenCV.

\*\*Challenge:\*\* Handling low-confidence predictions.

\*\*Solution:\*\* Implemented a confidence threshold to filter out uncertain results.

7. Conclusion

The "Detect Gender and Age using Artificial Intelligence" project successfully demonstrates the integration of AI and computer vision techniques for real-time human attribute detection. Future enhancements could include support for more diverse datasets, improved accuracy, and integration with cloud-based AI services for scalability.