Mini Project: Gender and Age Detection using Computer Vision and Deep Learning

Title:

Gender and Age Detection using Computer Vision and Deep Learning

Problem Statement:

In real-time applications like surveillance, marketing, and social media analytics, it's important to predict demographic details such as gender and age. Manually identifying these attributes is time-consuming and impractical. This project aims to use computer vision and deep learning techniques to automatically predict a person’s gender and age based on facial features from images or video.

Objective:

- To detect a person's face from an image or video.
- To classify the gender as male or female.
- To estimate the age group of the person.
- To implement the solution using a pre-trained deep learning model.

Expected Outcome:

- A system that takes an image as input and predicts gender and age.
- Accuracy assessment of the model for both gender and age predictions.
- A GUI or command-line tool for testing and demonstration.

Theory:

Gender and age detection is a computer vision task that involves identifying the gender and estimating the age of a person by analyzing their facial features. These features are extracted using deep learning models that have been trained on large datasets such as IMDB-WIKI, Adience, and UTKFace.

The project typically uses a Convolutional Neural Network (CNN), which is highly effective in image classification tasks. For this task, pre-trained models are used to reduce training time and improve accuracy.

Face Detection:
OpenCV provides a DNN module or Haar cascades to detect faces in an image or video frame.

Preprocessing:

The detected face region is resized and normalized to match the input requirements of the deep learning models.

Gender and Age Classification:
Two separate models are used:
- Gender Model: Predicts probabilities for 'Male' or 'Female'.
- Age Model: Predicts age group like (0–2), (4–6), (8–12), ..., (60–100).

Display Results:
Predicted age and gender are displayed as labels over the detected face in the image or video.

Steps to Perform the Project:

1. **Setup Environment:**
   - Install Python, OpenCV, and NumPy.
   - Download pre-trained models for gender and age detection (Caffe or ONNX models).

2. **Load Models:**
   - Use OpenCV DNN to load the gender and age models.

3. **Face Detection:**
   - Use Haar cascades or DNN face detector to find faces in the input.
   - Extract the face region from the image.

4. **Preprocess Face Image:**
   - Resize to 227x227 or as required by the model.
   - Normalize pixel values.

5. **Gender Prediction:**
   - Feed the face image to the gender model.
   - Interpret output as 'Male' or 'Female'.

6. **Age Prediction:**
   - Feed the face image to the age model.
   - Output is the predicted age group.

7. **Display Output:**
   - Draw bounding boxes on the face.
   - Display predicted gender and age above the box.

8. **Run on Multiple Images or Live Camera (Optional):**
   - Extend the script to work on webcam or video input for real-time prediction.

Pseudocode:

```
Input: Image or video frame
1. Detect face in the input
2. For each detected face:
   a. Extract and resize the face
   b. Predict gender using gender model
   c. Predict age using age model
   d. Display predicted gender and age on the image
Output: Image with gender and age labels
```

Test Case:

Input:
Image containing a person's face

Expected Output:
Gender: Male
Age: 25-32

Test multiple images with different gender and age to evaluate model accuracy.

Conclusion:

The gender and age detection system provides a fast and efficient way to estimate demographic information from facial images. It is useful for real-time applications and demonstrates the power of deep learning and computer vision when combined with pre-trained models. With further training on diverse datasets, accuracy can be improved even more.