Facial Emotion Detection using Python (OpenCV & DeepFace)

➢ Introduction

This guide will walk you through setting up a real-time facial emotion detection system using OpenCV for face detection and DeepFace for emotion recognition.

➢ Prerequisites

Before running the code, ensure you have the required software and packages installed.

1. Install Python

```
Python 3.10.11
```

Ensure you have Python 3.7 or later installed. You can check your version by running:

python --version

If Python is not installed, download and install it from Python's official website.

2. Install Required Packages

```
python -m pip install --upgrade pip
```

```
pip install opencv-python deepface
```

```
pip install opencv-python deepface numpy
```

```
pip install tf-keras
```

Use the following command to install the required Python libraries:

pip install opencv-python numpy deepface tensorflow

- **Code Implementation**

Save the following script as facial_emotion_detection.py and execute it.

```
import cv2
import numpy as np
from deepface import DeepFace
import os

# Suppress TensorFlow warnings
os.environ['TF_CPP_MIN_LOG_LEVEL'] = '3'

# Load OpenCV's pre-trained Haar Cascade face detector
face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_frontalface_default.xml')

# Start webcam capture
cap = cv2.VideoCapture(0)

if not cap.isOpened():
    print("Error: Could not access the webcam.")
    exit()

while True:
    ret, frame = cap.read()
    if not ret:
        print("Error: Failed to capture frame.")
        break

    # Convert frame to grayscale for better face detection
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    # Detect faces in the frame
    faces = face_cascade.detectMultiScale(gray, scaleFactor=1.3, minNeighbors=5, minSize=(50, 50))

    for (x, y, w, h) in faces:
```

```
        # Draw a rectangle around the detected face
        cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 2)

        # Extract and preprocess the face for emotion detection
        face_img = frame[y:y + h, x:x + w]
        face_resized = cv2.resize(face_img, (48, 48))  # Resize for better detection

        try:
            # Perform emotion analysis using DeepFace
            result = DeepFace.analyze(face_resized, actions=['emotion'], enforce_detection=False)

            # Get the dominant emotion
            emotion = result[0]['dominant_emotion']

            # Display the detected emotion
            cv2.putText(frame, emotion, (x, y - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.8, (0, 255, 0), 2)

        except Exception as e:
            print(f"Emotion detection error: {e}")

    # Show the video feed with detected faces and emotions
    cv2.imshow("Facial Emotion Detection", frame)

    # Exit the loop when 'q' is pressed
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

# Release resources
cap.release()
cv2.destroyAllWindows()
```

➢ How to Run the Script

1. **Open Terminal or Command Prompt**
   Navigate to the directory where you saved the script.
2. **Run the script**
   Use the command:
3. python facial_emotion_detection.py
4. **Usage**
   - The script will access your webcam.
   - It will detect faces and analyze emotions in real time.
   - Press **'q'** to exit.

➢ Common Errors & Fixes

1. ModuleNotFoundError

If you get an error like:

ModuleNotFoundError: No module named 'deepface'

Run:

pip install deepface

2. Webcam Not Working

If the webcam does not start, check:

- Another application might be using the camera.
- Ensure you have given permission for the webcam.

3. TensorFlow Warnings

If you see deprecation warnings, they are safe to ignore as long as the program runs.