Real-Time Emotion Detection Using Python and Deep Learning

Abstract

This project aims to detect human emotions in real-time using computer vision and deep learning. Using a webcam, facial expressions are captured, processed, and classified into emotions such as Happy, Sad, Angry, Surprise, Neutral, etc. The project leverages OpenCV for video capture and face detection, and a trained Convolutional Neural Network (CNN) for emotion classification.

Objectives

- Develop a real-time system that identifies facial emotions from webcam input.

- Explore deep learning techniques for emotion classification.

- Implement a user-friendly and efficient application using Python.

Tools and Technologies

• Python: A high-level programming language known for its simplicity and readability, ideal for rapid application development and data analysis.

• OpenCV: An open-source computer vision and machine learning software library used for image and video processing, including real-time face detection.

• TensorFlow/Keras: Deep learning frameworks used for building and training neural networks, especially Convolutional Neural Networks (CNNs).

• NumPy: A powerful numerical computing library in Python, essential for handling arrays and matrix operations efficiently.

• FER-2013 Dataset: A publicly available dataset of labeled facial images for emotion recognition, commonly used to train models in emotion detection tasks.

• Webcam: A camera device used for real-time video feed, which serves as the input source for emotion detection.

System Architecture / Workflow

1. Capture: Webcam feed is accessed using OpenCV.
2. Detect: Faces are detected using Haar cascades or deep neural network-based face

detection models.
3. Preprocess: Extracted face regions are converted to grayscale, resized, and normalized.
4. Predict: Preprocessed image is passed to the trained CNN model which predicts the emotion.
5. Display: The predicted emotion is displayed on the screen, overlaid on the video feed.

Model Summary

The model is a Convolutional Neural Network (CNN), designed for image classification tasks. It takes 48x48 grayscale facial images as input and outputs a classification among 7 emotion categories: Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral. The model is trained using the FER-2013 dataset and achieves an accuracy of approximately 65-75% on validation data, depending on architecture and training conditions.

Project Modules

• data_preprocessing.py – For preparing and processing the dataset.

• train_model.py – For building, training, and saving the CNN model.

• emotion_detector.py – For capturing webcam feed and performing real-time emotion detection.

• model.h5 – The saved pre-trained CNN model file used for prediction.

Results

The system performs real-time emotion detection at an average frame rate of 15–25 FPS depending on hardware specifications. It can reliably detect emotions under good lighting and proper facial orientation. However, performance may drop under poor lighting, occlusions, or extreme facial angles.

Applications

• Virtual classroom and student engagement monitoring.

• Sentiment analysis for customer service improvement.

• Enhanced human-computer interaction systems.

• Therapeutic and mental health monitoring tools.

• Gaming and entertainment industry for real-time feedback.

Limitations and Challenges

• Sensitivity to lighting conditions and camera quality.

• Difficulty distinguishing between visually similar emotions like fear and surprise.

• FER dataset may lack real-world diversity in facial features and expressions.

Future Enhancements

• Improve model accuracy using more diverse datasets like AffectNet.

• Integrate voice-based emotion detection for multimodal analysis.

• Create mobile or embedded versions using hardware like Raspberry Pi.

• Enhance emotion prediction using facial landmark detection or attention mechanisms.

Conclusion

This project demonstrates the potential of deep learning and computer vision in understanding human emotions in real-time. While there are challenges, the system lays the foundation for emotion-aware applications that can be further developed and improved for greater reliability and scope.

PROJECT BY:

SUJEENDRA VELLURI – sujeendravelluri@gmail.com

JASSMITHA JAMMU - jjassmitha@gmail.com

JAABILY SRILEKHA - jaabilysrilekha05@gmail.com