Sania Arshad 70138511

Maira Ashraf 70137927

Emotion Detection AI System – Project Documentation

Abstract

The Emotion Detection AI System is an Artificial Intelligence–based application designed to identify human emotions from facial images. Emotion categories like happy, sad, angry, surprised, or neutral are classified by the system based on its analysis of facial expressions. The project is created with Python in the Anaconda environment, making use of well-known computer vision and machine learning libraries. This documentation provides a clear and organized explanation of the system's entire operation, design, tools, and future scope.

1. Introduction

An image of a person is provided as input for this project. The system recognizes the face, analyzes the picture, and forecasts the person's emotion. Anaconda, which offers a full Python development environment with Jupyter Notebook and necessary libraries, is used to implement the project.

2. Problem Statement

Understanding human emotions manually from facial expressions is subjective and time-consuming. There is a need for an automated system that can accurately detect and classify emotions from facial images using Artificial Intelligence techniques.

3. Objectives

The main objectives of this project are:

- To develop an AI-based system that detects human emotions from facial expressions
- To train a machine learning model capable of classifying different emotions
- To use the Anaconda environment for easy development and execution
- To implement the project using Python and Jupyter Notebook
- To generate accurate emotion predictions from input images

4. Scope of the Project

- The system works on static facial images
- It detects basic human emotions
- It runs completely offline after setup
- The project can be extended to real-time emotion detection

---

5. Tools and Technologies Used

The following tools and technologies are used in this project:

Software Tools

- **Anaconda Navigator** – For managing Python environment and libraries
- **Jupyter Notebook** – For writing, testing, and executing code

Programming Language

- **Python 3.x**

Libraries

- **OpenCV** – Image processing and face detection
- **TensorFlow / Keras** – Building and training the neural network model
- **NumPy** – Numerical and mathematical operations
- **Matplotlib** – Displaying images and prediction results

---

6. Dataset Description

The project uses the **FER-2013 (Facial Expression Recognition)** dataset. This dataset contains thousands of grayscale facial images labeled with different emotions.

Emotion Classes

- Happy
- Sad
- Angry
- Surprised

- Neutral
- Fear
- Disgust

The dataset is divided into training and testing sets for model learning and evaluation.

---

7. System Architecture

The working of the system follows these steps:

1. Image Input
2. Face Detection using OpenCV
3. Image Preprocessing (resizing and grayscale conversion)
4. Feature Extraction using CNN
5. Emotion Classification
6. Display of Detected Emotion

---

8. Methodology

8.1 Image Acquisition

The user uploads or selects an image within the Jupyter Notebook environment.

8.2 Face Detection

OpenCV is used to detect the face region from the input image.

8.3 Preprocessing

- Image is resized to a fixed size
- Converted to grayscale
- Pixel values are normalized

8.4 Model Training

A Convolutional Neural Network (CNN) is trained using the FER-2013 dataset to learn emotion patterns from facial features.

8.5 Emotion Prediction

The trained model predicts the emotion of the detected face and displays the result.

---

9. Model Description

The CNN model consists of:

- Convolutional layers for feature extraction
- Max-pooling layers to reduce dimensions
- Fully connected (Dense) layers
- Softmax activation function for multi-class classification

---

10. Implementation Details

- The project is implemented in Python using Jupyter Notebook
- The dataset is loaded and preprocessed
- The CNN model is trained and saved
- The saved model is used for emotion prediction on new images

---

11. User Manual

Steps to Run the Project:

1. Open Anaconda Navigator
2. Launch Jupyter Notebook
3. Open the project notebook file
4. Load the dataset in the specified directory
5. Run all code cells sequentially
6. Upload or select an image
7. View the detected emotion on the screen

---

12. Advantages

- Easy to use and run within Anaconda

- Works offline after installation
- Accurate emotion detection
- Good learning project for AI and computer vision
- Can be extended easily

---

13. Limitations

- Works best with clear and front-facing images
- Accuracy depends on dataset quality
- Not implemented for real-time video detection

---

14. Future Enhancements

- Real-time emotion detection using webcam
- Graphical user interface using Tkinter or Streamlit
- Improve accuracy with larger datasets
- Add voice-based emotion detection

---

15. Conclusion

The Emotion Detection AI System effectively illustrates how computer vision and artificial intelligence can be used to recognize human emotions from facial images. The project is easy to complete, efficient, and appropriate for academic use. It offers a solid base for upcoming improvements and practical uses.

---

16. References

- FER-2013 Facial Expression Dataset
- Python Documentation
- TensorFlow and Keras Official Documentation