Software Requirements Specification

For

Face Detection and Emotion Recognition Using Python and Machine Learning

Prepared by

Members	USN
Prashanth K	4KV22CI037
Ankith K R	4KV22CI006
Keerthan S	4KV22CI023
Anoop Mathew	4KV22CI007

November 12, 2024

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

the **Face Detection and Emotion Recognition System**, which detects human faces and identifies emotions from facial expressions in images or video streams. This system aims to be used in various domains like security surveillance, retail analytics, user interaction systems, and customer experience analysis.

1.2 Document Conventions

- **Bold** text will be used to highlight the titles of system features and key terms.
- *Italic* text will be used for placeholders or examples.
- Requirements are presented using "Shall" (mandatory), "Should" (preferred), and "May" (optional).
- Each requirement will have a unique identifier (e.g., REQ-1, REQ-2).

1.3 Intended Audience and Reading Suggestions

- **Developers**: Detailed functional and non-functional requirements for implementation.
- **Project Managers**: Understanding the product scope, priorities, and deadlines.
- **Testers**: Information on test scenarios, validation criteria, and performance metrics.
- **End Users**: Basic overview of the system and interface guidelines.

Suggested sequence of reading:

1. Start with Section 2 (Overall Description) to get an understanding of the system.
2. Move to Section 4 (System Features) for detailed functionalities.
3. Section 5 (Non-functional Requirements) for performance and security considerations.
4. Section 6 (Other Requirements) for any external dependencies.

1.4 Product Scope

The **Face Detection and Emotion Recognition** system will:

- Detect faces in real-time from images and video streams.
- Classify the detected faces into different emotional categories based on facial expressions.
- Be deployed in environments requiring real-time processing, such as security surveillance and customer interaction analysis.

This system will provide a framework for face detection and emotion recognition, leveraging machine learning models trained on facial expression datasets. It will be compatible with various camera devices and will support cloud-based or edge computing deployment.

1.5 References

- OpenCV Documentation for face detection.
- TensorFlow/Keras Model for Emotion Recognition.

2. Overall Description

2.1 Product Perspective

This system is a **standalone module** for face detection and emotion recognition. It integrates seamlessly with external systems like surveillance cameras, kiosks, or digital signage platforms. The system is intended for real-time operation, processing video feeds from various input devices.
A typical architecture could consist of:

- **Input Devices**: Cameras for image/video capture.
- **Face Detection Module**: Identifies and locates faces in input frames.
- **Emotion Recognition Module**: Classifies the detected faces into predefined emotion categories.
- **Output Devices**: Display emotional labels and detected faces in real-time. The system can be used as an add-on to existing surveillance and customer analytics systems.

2.2 Product Functions

- **Face Detection**: Detect human faces from static images and video streams.
- **Emotion Recognition**: Identify emotions such as happiness, sadness, anger, surprise, fear, and neutral from detected faces.
- **Real-time Processing**: Provide live feedback of detected faces and emotional states in video feeds.
- **Data Logging**: Optionally log detection events and recognized emotions.

- **API for Integration**: Provide an API for other applications to access face detection and emotion recognition capabilities.

2.3 User Classes and Characteristics

- **End Users**: Users interacting with retail or customer service systems that utilize emotion recognition for improving customer experience. They interact indirectly with the system through these applications.
- **Security Personnel**: Users who will use the system for surveillance and security, monitoring faces and recognizing emotional states in real-time.
- **Administrators**: Personnel who will configure, maintain, and monitor the system.
- **Developers**: Integrators who will use the system's API for adding face detection and emotion recognition capabilities to external applications.

2.4 Operating Environment

- **Hardware**: The system will operate on standard PCs, edge devices (e.g., Raspberry Pi), or cloud infrastructure. It requires a **camera** for input, ideally with a resolution of 720p or 1080p.
- **Software**: The system will use Python, with machine learning frameworks such as **TensorFlow** or **Keras** for emotion recognition and **OpenCV** for face detection.
- **Operating System**: Compatible with **Windows**, **Linux** (Ubuntu), or **macOS**.
- **Dependencies**: **CUDA**-enabled GPU for real-time processing, OpenCV for image processing, TensorFlow/Keras for deep learning models.

2.5 Design and Implementation Constraints

- **GPU**: The system requires **NVIDIA GPU** support for hardware acceleration of face detection and emotion recognition.

- **Real-time Processing**: The system must be capable of processing video at a rate of at least 30 frames per second (FPS).
- **Privacy Compliance**: The system must comply with **GDPR** (General Data Protection Regulation) for facial data processing, including encryption and data anonymization.
- **Scalability**: The system should be scalable to handle multiple camera inputs and large-scale data storage.

2.6 User Documentation

- **User Manual**: Includes installation instructions, setup guidelines, and usage instructions for security and retail personnel.
- **API Documentation**: For developers wishing to integrate the face detection and emotion recognition system into their applications.
- **Onboard Help**: Contextual help within the system for users to understand the system's interface and features.

2.7 Assumptions and Dependencies

- The system assumes the availability of a **video feed** from cameras for real-time face detection.
- The face detection model assumes good lighting and visible faces in the frame for optimal performance.
- External APIs (if any) for storing detected data or recognizing emotions may have their own constraints, such as rate limits.

3. External Interface Requirements

3.1 User Interfaces

- **GUI**: The GUI will display detected faces with emotion labels overlaid. The interface will include a live video feed from the camera with bounding boxes around the detected faces and emotion categories displayed near each face.
- **Real-time Feedback**: Users will see real-time emotion detection results for each person in the camera feed.
- **Error Handling**: The interface will show error messages when faces are not detected or if the system is unable to classify emotions.

3.2 Hardware Interfaces

- **Cameras**: The system will interface with standard USB or IP cameras. Cameras must support at least 720p resolution and must be capable of streaming video at a minimum of 30 FPS.
- **GPUs**: The system will interface with NVIDIA GPUs for acceleration of image processing and machine learning tasks.

3.3 Software Interfaces

- **OpenCV**: Used for real-time face detection from images and video streams.
- **TensorFlow/Keras**: Utilized for training and executing deep learning models for emotion classification based on facial expressions.
- **REST API**: Provides services to other systems for face detection and emotion recognition.

3.4 Communications Interfaces

- **HTTPS**: Communication between the system's backend and any external systems (e.g., cloud storage, external APIs) should be encrypted using HTTPS to ensure secure transmission of data.
- **Data Storage**: Data related to detected faces and emotions (if logged) should be stored in a secure database, with encryption enabled.

4. System Features

4.1 Face Detection

4.1.1 Description and Priority

This feature will detect faces in static images and real-time video streams.
Priority: High.

4.1.2 Stimulus/Response Sequences

- **Stimulus**: The user starts a video feed or uploads an image.
- **Response**: The system detects faces and overlays bounding boxes around them.

4.1.3 Functional Requirements

REQ-1: The system shall detect faces with at least 95% accuracy in a standard environment.

REQ-2: The system shall process video input at a minimum of 30 FPS with a maximum latency of 100 milliseconds per frame.

4.2 Emotion Recognition

- **4.2.1 Description and Priority**

  Once faces are detected, the system classifies emotions such as happiness, sadness, anger, surprise, fear, and neutral. **Priority**: High.
- **4.2.2 Stimulus/Response Sequences**
- **Stimulus**: A face is detected.
- **Response**: The system classifies the detected face into an emotion category (e.g., happy, sad, etc.).
- **4.2.3 Functional Requirements**
- **REQ-3**: The system shall classify detected faces into one of six emotional categories with 4o mini

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- The system shall process at least 30 FPS for video input.
- The face detection process should not exceed 100 milliseconds per frame.

5.2 Safety Requirements

- The system should avoid storing or sharing sensitive data such as faces without the user's explicit consent.
- Data should be encrypted both during transmission and at rest.

5.3 Security Requirements

- The system should authenticate users accessing administrative or configuration features.
- All video data and emotional analysis results should be encrypted during transmission.

5.4 Software Quality Attributes

- **Reliability**: 99.9% uptime for critical applications.
- **Usability**: Intuitive user interface with minimal training required for operation.

6. Other Requirements

- **Legal Compliance**: The system shall comply with relevant laws regarding privacy (e.g., GDPR).
- **Multi-language Support**: The system should support multiple languages for diverse international deployments.

Appendix A: Glossary

Face Detection: The process of identifying human faces in images or video streams.

Emotion Recognition: Classifying a person’s emotional state based on facial expressions.

Appendix B: Analysis Models

Data flow diagrams, state transition diagrams for the flow from image capture to emotion recognition.

Appendix C: To Be Determined List

TBD: Final choice of emotion recognition model architecture.

TBD: Final integration with video management systems for larger deployments.