**Emotion Detection GUI System — Full Technical Report**

**Overview**

This **Emotion Detection GUI System** is a real-time facial expression analysis platform combining deep learning and interactive GUI design. Built with TensorFlow, OpenCV, and Tkinter, the system enables end-users to analyze emotional states from webcam video input with intuitive visual feedback. It uses a trained Convolutional Neural Network (CNN) to recognize emotions from facial features and presents a detailed, user-friendly interface with live video, prediction overlays, emotion probability charts, and performance metrics.

**Features Summary**

* **Real-Time Emotion Recognition**: Classifies emotional expressions frame-by-frame from a live video stream.
* **Tkinter GUI**: Compact, responsive interface integrating video feed, controls, and analytics.
* **Live Probability Bar Chart**: Visual feedback using Matplotlib showing predicted probabilities of all emotions.
* **FPS (Frames Per Second) Tracker**: Helps monitor system performance.
* **Customizable Input**: Can run from webcam or external video file.
* **Customizable Model**: Accepts any trained Keras model with the same architecture.
* **Cross-platform and GPU-Compatible**: Designed for desktop systems with optional TensorFlow GPU acceleration

**Mathematical Logic Behind CNN-based Emotion Recognition**

**1. Convolutional Layers**

* Apply a filter (kernel) over small patches of the input image to detect local features (edges, textures).
* Mathematically, convolution is defined as: Where is the input image, is the kernel, and is the output feature map.

**2. ReLU Activation Function**

* Applies non-linearity after each convolution: This avoids vanishing gradients and allows CNNs to learn complex patterns.

**3. Max Pooling**

* Downsamples the feature maps by selecting the max value in a neighborhood: Reduces computational cost and overfitting.

**4. Fully Connected Layers**

* Flatten the feature maps and apply matrix operations: Used for final classification.

**5. Softmax Layer**

* Converts logits (raw scores) to probabilities:

**6. Loss Function (Categorical Cross Entropy)**

* Measures error between predicted and true :

This mathematical pipeline enables the system to effectively learn and distinguish between subtle emotional patterns in facial expressions.

**Core Components & Logic**

**1. Face Detection**

* Utilizes OpenCV's Haar cascade classifier (haarcascade\_frontalface\_default.xml)
* Detects multiple faces per frame; uses the largest one
* Converts video to grayscale for processing

**2. Emotion Classifier**

* CNN model (emotion\_cnn\_gpu.h5) trained on 48x48 grayscale images
* Input: Normalized, resized face ROI (Region of Interest)
* Output: Probability vector over 7 emotions:
  + Angry, Disgust, Fear, Happy, Neutral, Sad, Surprise

**3. GUI (Tkinter)**

* Main window includes:
  + Live webcam video
  + Real-time FPS label
  + Emotion probability chart (Matplotlib embedded)
  + Quit button
* Auto-updates frames and chart every ~10ms
* Displays top-predicted label and confidence on the frame

**4. Matplotlib Integration**

* Uses FigureCanvasTkAgg to embed chart in GUI
* Updates bar heights for each frame with current emotion probabilities

**5. Performance Handling**

* Calculates FPS every 10 frames
* Uses ImageTk to convert OpenCV frames for Tkinter display

**Launch Options**

Run with webcam (default):

python main.py

Run with custom video source:

python main.py --source path/to/video.mp4

Run with custom-trained model:

python main.py --model path/to/emotion\_model.h5

**Dataset & Training**

**Dataset Structure:**

models/

├── train/

│ ├── angry/

│ ├── disgust/

│ └── ...

└── test/

└── [same structure]

**Model Architecture:**

* Multiple convolutional layers + batch normalization
* Max pooling for feature downsampling
* Fully connected layers with dropout
* Output softmax layer for emotion classification

**Use Cases**

* Real-time emotion feedback for presenters or streamers
* UX testing and emotional response tracking
* Classroom engagement monitoring
* Interactive art installations or games
* Assistive tools for emotion recognition training

**Strengths**

* Clean modular architecture
* Accurate and smooth classification
* Easy to install and run
* Fully GPU compatible (via TensorFlow GPU backend)

**Limitations**

* Limited to 7 discrete emotions
* No multi-face tracking (only processes largest face)
* Haar cascades less accurate in varied lighting/angles
* No logging or analytics support for tracking over time

**Future Opportunities**

* Add timeline chart for emotion over time
* Support multiple face tracking and recognition
* Include face ID linking with emotional profiles
* Enable export of emotion data to CSV or JSON
* Add voice/audio interaction based on emotion state

**Conclusion**

This system is a compact yet powerful demonstration of real-time emotion recognition using accessible technologies. Its combination of performance, usability, and extensibility makes it a strong base for academic projects, research prototypes, or user-facing emotion-aware tools.

**Created: April 29, 2025**  
**Lead Developer: Nour Benzahra**