LIVE EMOTION DETECTOR

Overview

- Abstract
- Problem Definition
- Introduction
- Literature Survey
- Objective
- Data source
- Methodology
- CNN Algorithm
- Model Evaluation & Results
- Future Enhancements
- Conclusion
- References

Abstract

- Emerging in affective computing and social signal processing, automatic emotion recognition via facial expression analysis is a burgeoning field.
- While current research primarily centers on recognizing basic emotions (happy, sad, surprise, disgust, fear, etc.), addressing mixed emotion recognition is less explored due to its inherent complexity.
- Our project aims to discern both simple and compound emotions from facial expressions, conducting a comparative analysis of the model's emotional identification capabilities.

Problem Definition

- Identifying compound emotions presents a formidable challenge, aiming to detect live emotions using Convolutional Neural Network (CNN) models.
- The core objective is to harness the power of deep learning to accurately recognize and interpret facial expressions, providing valuable insights into human emotions.

Introduction

- Facial expression recognition identifies human emotions using both natural and computational methods.
- Unlike facial recognition, which matches faces to datasets, it focuses on understanding emotions.
- Applications include training systems, like chatbots, to respond to human emotions, enabling actions such as capturing a smile-triggered image or adjusting interactions based on user mood in automation systems.

Literature Survey

- Facial Emotion Recognition has been a subject of considerable interest among behavioral scientists since C. Darwin's work in 1872.
- In 1978, Paul Ekman and Wallace V. Friesen pioneered the initial effort in facial emotion analysis with the creation of the Facial Action Coding System.
- A study by Zhang et al delved into facial emotion recognition, exploring two feature types: geometry-based features and Gabor wavelets-based features.

Objective

- This project endeavors to create and deploy a Convolutional Neural Network (CNN) tailored for Live Emotion Detector.
- The CNN model will undergo training using a dataset featuring facial images annotated with diverse emotion labels, including happiness, sadness, anger, surprise, and more.
- The primary objective is to construct a model capable of sensing and interpreting individual emotions, thereby promoting enhanced well-being.
- The overarching aim is to elevate services and cultivate more efficient human-machine interactions.

What is Emotion?

"Emotions find expression through voice, hand and body gestures, with a primary focus on facial expressions. These non-verbal cues play a crucial role in conveying feelings, enriching communication, and enhancing emotional intelligence."

- There are different types of facial emotions are like Happy ,Sad ,Disgust , Fear ,Surprise ,Excitement ,Angry ,Confused ,Neutral etc.

Why Facial Expression?

- Facial nerve disorders
- Computer systems that understand human behavior
- Lie Detection
- Security systems
- Speech recognition
- Emotion for animation
- Behavior assessment of emotion and paralinguistic displays

Importance & Application

- Human beings express emotions in day to day interactions.
- Understandings emotions and knowing how to react to people's expressions greatly enriches the interaction.
- Visual Communication: Enhancing virtual interactions
- Mental Health Monitoring: Detecting signs of stress or mood changes.
- Entertainment: Personalizing user experiences in gaming and content consumption.

Data Source

- For sourcing data, I gathered images depicting various emotions through a web scraping technique.
- Web scraping is a technique used to extract data from websites.
- In this process, I utilized the BeautifulSoup tool, a Python library, known for its efficiency in parsing HTML and XML data on the web.

The Dataset

- Regarding the tasks of recognizing expressions, the dataset necessary for training and testing the network must adhere to specific criteria.
- The data should consist of images where the entire face is predominantly visible, with faces primarily oriented towards the front.
- Excessive rotation along the Y and Z axes should be minimized.
- Additionally, the images should possess a sufficiently high resolution to ensure optimal training conditions.

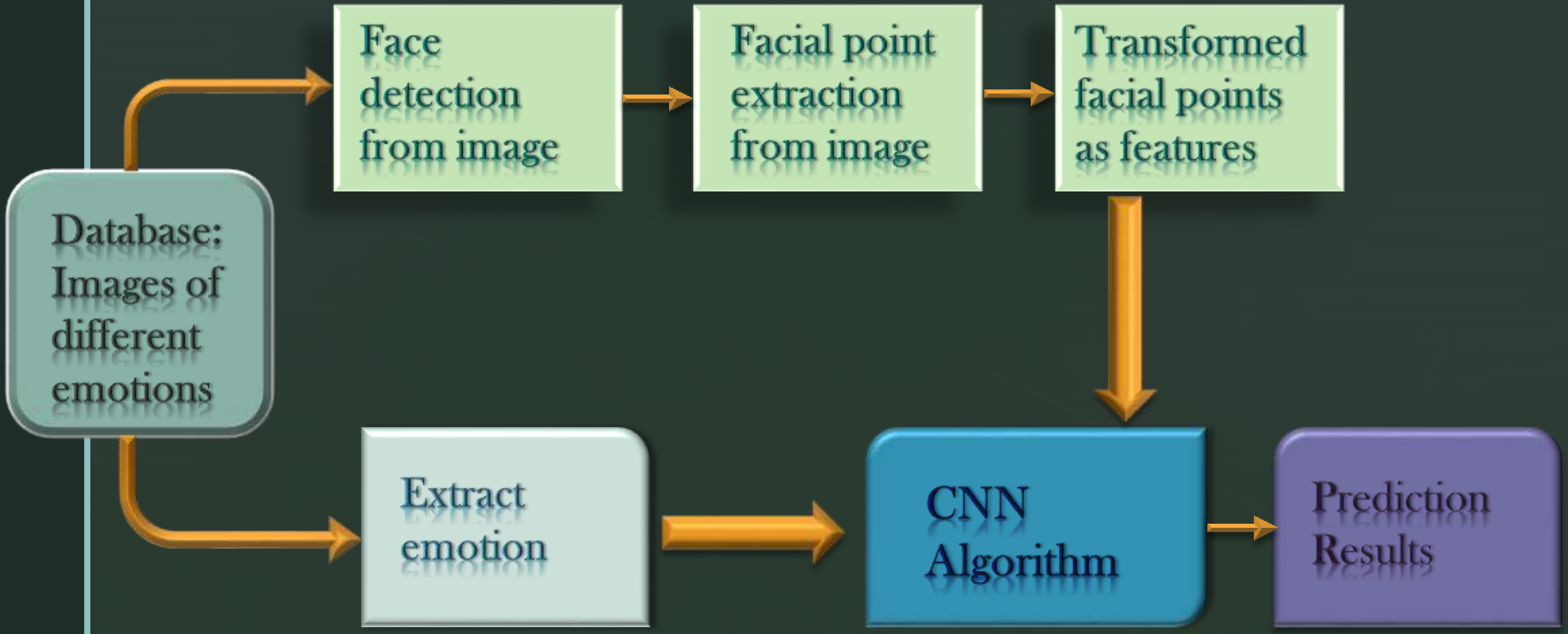
Dataset Description

- The dataset comprises 250 * 250 pixel face images categorized into nine emotion classes: 'Happy,' 'Sad,' 'Disgust,' 'Confused,' 'Fear,' 'Excitement,' 'Surprise,' 'Neutral,' and 'Angry.’
- To facilitate model development, the dataset was divided, dedicating 80% for training and 20% for testing.
- This organization ensures a diverse and representative dataset for effective training and evaluation.

Proposed Methodology

- Capture Images
- Image pre-processing
  - RGB to Gray scale conversion
  - Scale -Normalization
- Feature Extraction
- Building the Model
- Model Training
- Simple and Compound Facial Emotion Detection
- Measuring the performance of the model

Methodology



RGB – Grey Scale Conversion

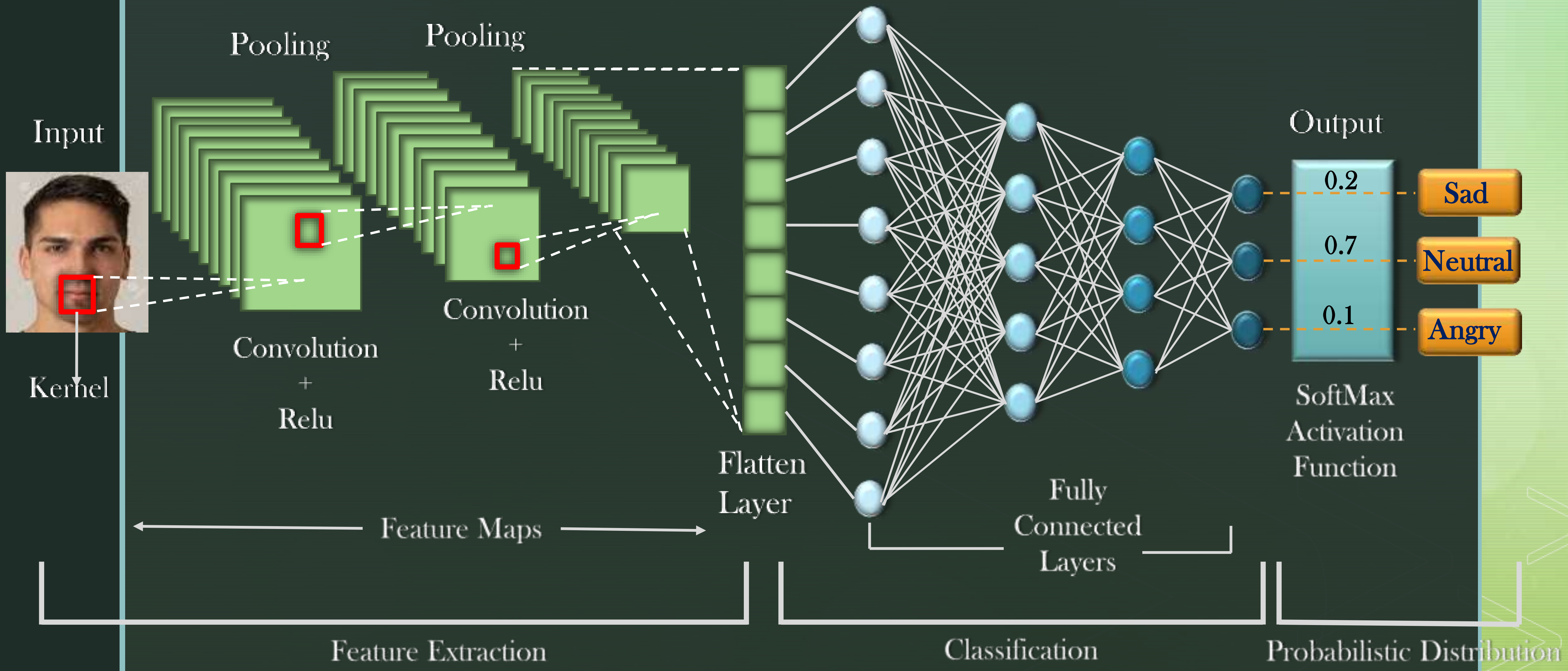
- Conversion from RGB to grayscale simplifies images to a single channel, emphasizing luminance while removing color complexities.
- Grayscale facilitates efficient brightness and contrast analysis, crucial for various computer vision tasks.
- OpenCV simplifies RGB to grayscale conversion in Python through efficient library import, image loading, conversion with `cv2.cvtColor()`, and display or saving using `cv2.imshow()` or `cv2.imwrite()`.

Convolution Neutral Network

- A Convolutional Neural Network (CNN) is a type of neural network specialized in extracting features from images.
- This fundamental operation is crucial in machine learning and serves as a foundational model in popular networks like GoogleNet, VGG19, and others.
- CNNs are extensively employed for tasks such as object detection and image classification due to their effectiveness in capturing meaningful patterns and features from visual data.

- CNN has the following five basic components:
  - Convolution Layer: Detects features using filters that slide across the input image.
  - ReLU Activation: Introduces non-linearity by applying Rectified Linear Unit activation to the convolutional outputs.
  - Pooling Layer: Reduces spatial dimensions, retaining the most important features.
  - Flattening: Transforms the pooled feature maps into a 1D vector, preparing them for the fully connected layers.
  - Fully Connected Layer: Performs classification based on the learned features from the previous layers.

Model Training using CNN



Optimizer, Loss Function & Metrics

- Utilizing categorical cross-entropy, a loss function measuring the difference between predictions and actual values, especially in classification models with probability outputs.
- This function penalizes divergence from true labels, with higher loss for significant deviations, e.g., predicting 0.012 when the actual label is 1.

```
# Compile the model
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
```

- The optimizer used is Adam().
- Adam stands for Adaptive Moment Estimation.
- This method dynamically computes adaptive learning rates for individual parameters by maintaining an exponentially decaying average of past squared gradients (similar to AdaDelta) and past gradients (similar to momentum), enhancing training efficiency in deep learning models.
- Accuracy, a common metric, quantifies the proportion of correctly predicted instances out of the total.
- It provides a straightforward measure of model performance in classification tasks, gauging the overall correctness of predictions.

Validation

- In the validation phase, a combination of OpenCV and Keras functions was employed.
- Initially, the video frames were stored in a video object. Each frame underwent a transformation, converting it to grayscale and resizing it through NumPy.
- This resized image was then reshaped and fed into a predictor, loaded using the `keras.model.load_model()` function.
- The maximum output was extracted. Subsequently, facial regions were identified, and a rectangle was drawn around them.
- The output was then formatted above the rectangle box.

Model Performance Evaluation

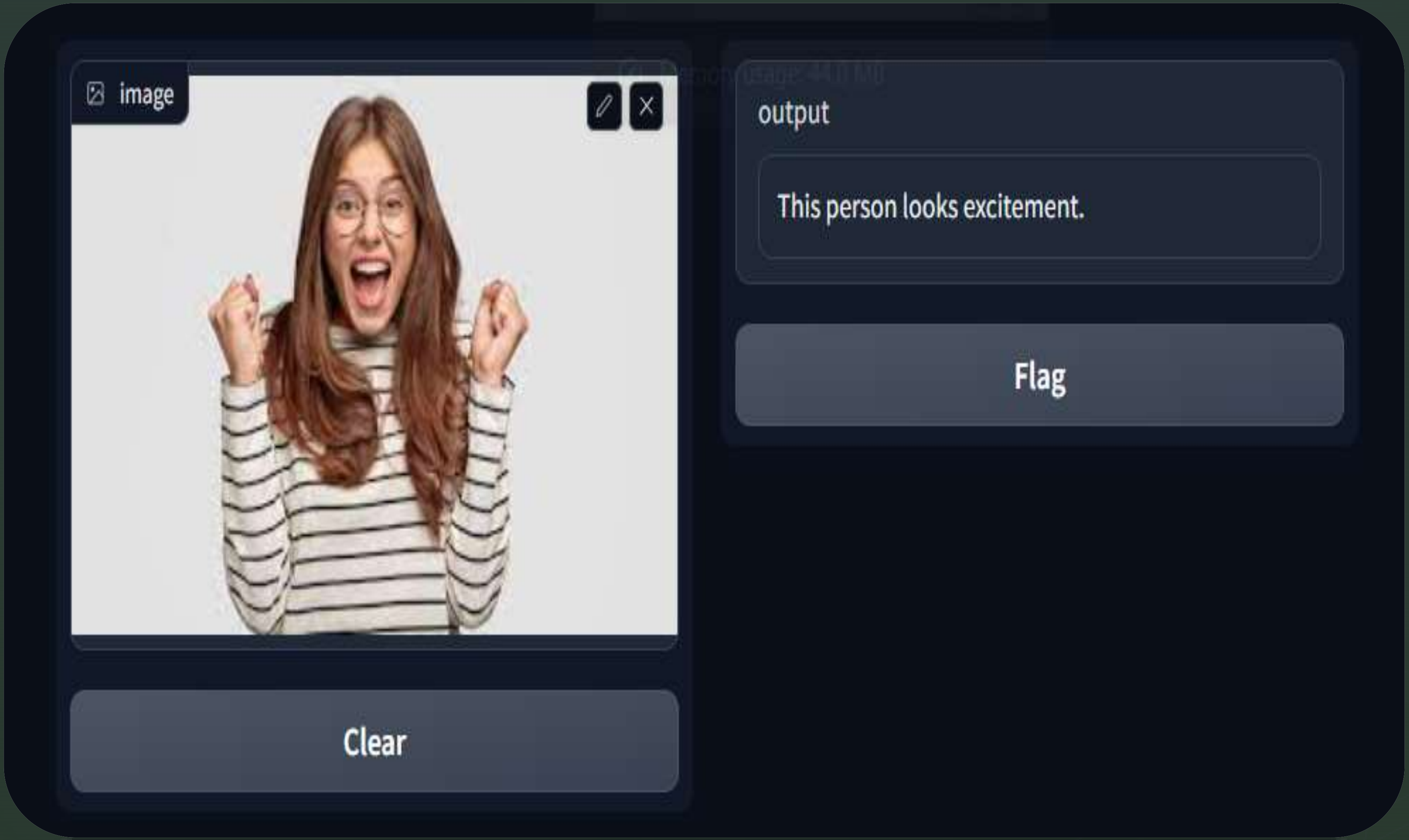
- During training, the CNN model achieves a validation accuracy of 79%, indicating its proficiency in learning emotional representation features from the training images.
- The training process unfolds over several epochs, each utilizing a batch size of 32.

```
Epoch 6/10
115/115 [==============================] - 66s 573ms/step - loss: 1.4710 - accuracy: 0.5333 - val_loss: 2.2904 - val_accuracy: 0.1902
Epoch 7/10
115/115 [==============================] - 68s 588ms/step - loss: 1.2668 - accuracy: 0.6115 - val_loss: 2.3738 - val_accuracy: 0.1913
Epoch 8/10
115/115 [==============================] - 69s 597ms/step - loss: 1.0955 - accuracy: 0.6721 - val_loss: 2.5189 - val_accuracy: 0.1781
Epoch 9/10
115/115 [==============================] - 66s 573ms/step - loss: 0.9253 - accuracy: 0.7361 - val_loss: 2.7096 - val_accuracy: 0.1825
Epoch 10/10
115/115 [==============================] - 63s 550ms/step - loss: 0.7839 - accuracy: 0.7973 - val_loss: 2.8429 - val_accuracy: 0.1781
<keras.src.callbacks.History at 0x7b0a4eccdd50>
```

Model Summary

- The model follows a sequential architecture with a Conv2D layer for feature extraction, max pooling for downsampling, and dense layers for classification.
- It has a total of 14,259,785 parameters, offering flexibility for capturing complex relationships.
- The output layer comprises 9 neurons for emotion classes.
- The model is trainable, optimizing for enhanced performance during training.
- With a compact size of 54.40 MB, it ensures efficiency in deployment.
- In summary, the model is well-structured for facial emotion recognition.

Result



Future Enhancements

- **Multi-Model Integration:** Incorporate voice and body language for a comprehensive understanding, enhancing emotion detection accuracy.
- **Emotion Intensity:** Focus on quantifying emotion intensity for nuanced insights, improving interaction and mental health applications.

Conclusion

- In conclusion, our live emotion detection project leverages advanced technologies for precise emotion identification, promising applications in diverse fields, particularly enhancing human-computer interaction.
- Recognizing current achievements, future enhancements focus on refining capturing processes and simplifying edge detection, ensuring sustained progress in affective computing.

References

- Ekman, P., & Friesen, W. V. (1971). Constants across cultures in the face and emotion. Journal of Personality and Social Psychology, 17(2), 124–129.
- Andre Teixeira Lopes, Thiago Oliveira-Santos & Alberto F. de Souza, Rodrigo F. Berriel (2017). Deep learning for emotion recognition: A comprehensive review. Computer Vision and Image Understanding, 166, 6–20.
- Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). Deep learning (Vol. 1). MIT press Cambridge.
- Liu, Z., Song, Y., & Tao, D. (2014). Facial expression recognition with three-dimensional model based CNN. In European Conference on Computer Vision (pp. 508–515).

THANK YOU