

CAPSTONE PROJECT

REAL TIME FACIAL EMOTION DETECTION USING DEEP LEARNING

PRESENTED BY

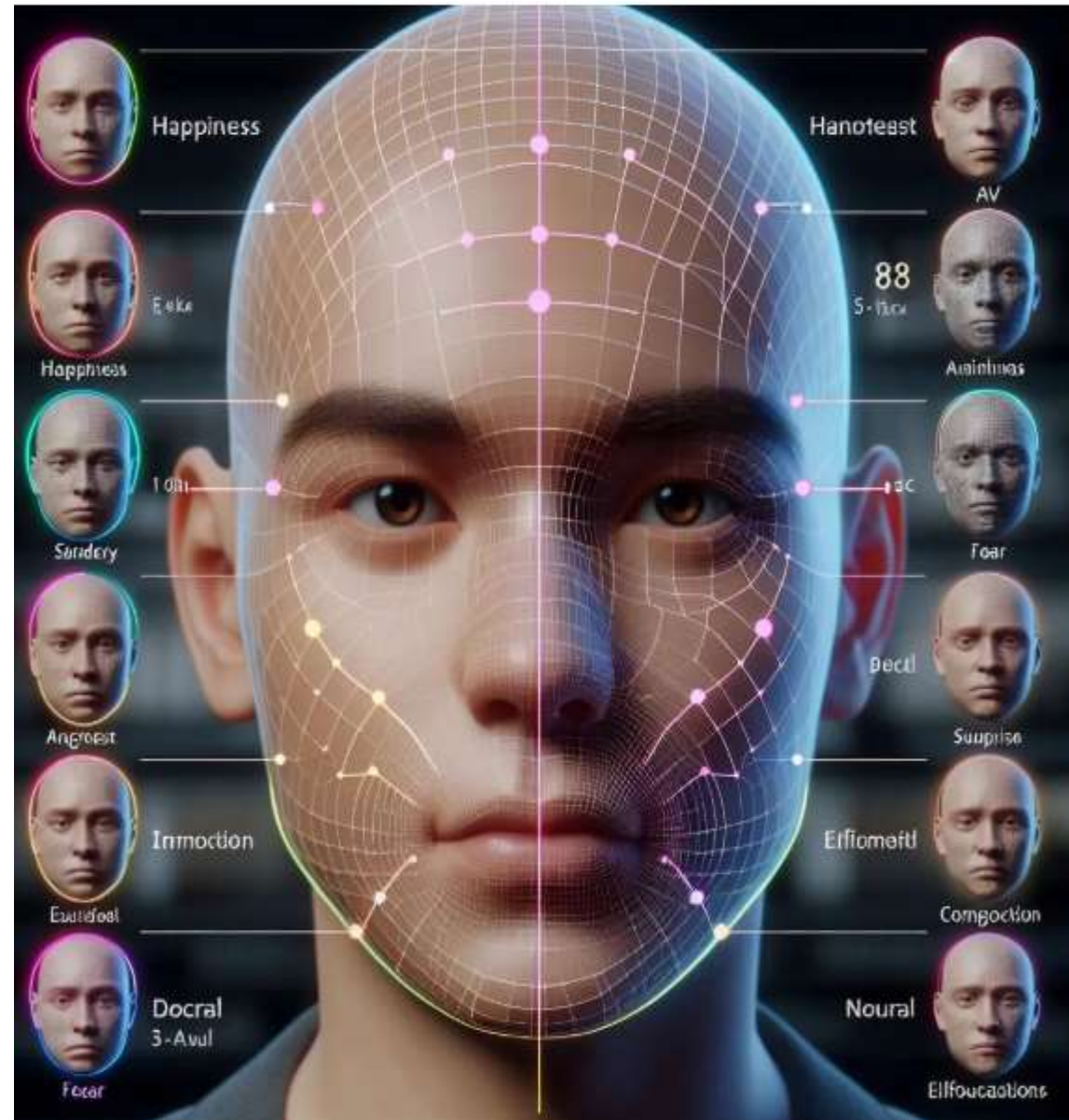
STUDENT NAME: SURYANSH DHAMA

COLLEGE NAME: KR.MANGALAM UNIVERSITY

DEPARTMENT: BTECH CSE AI/ML

EMAIL ID: YDVRYO@GMAIL.COM

AICTE STUDENT ID: STU67E697B2203411743165362



OUTLINE

- **Problem Statement**
- **Proposed System/Solution**
- **System Development Approach**
- **Algorithm & Deployment**
- **Result (Output Image)**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

- Humans express emotions using facial expressions. Recognizing these emotions manually is subjective and inefficient.
- There is a growing need for intelligent systems that can automatically detect emotions in real-time, especially in applications like healthcare, online education, human-computer interaction, and mental wellness.
- The challenge lies in accurately detecting and classifying emotions using facial features in real-time video feeds.
- **DataSet link :-** <https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge>

This dataset contains 35887 grayscale 48x48 pixel face images with seven emotions.



HAPPY ANGRY NATURAL FEAR SAD DISGUST SURPRISE

PROPOSED SOLUTION

- **The proposed system captures real-time facial expressions using a webcam and classifies emotions using a deep learning model.**
- **Key Features:**
 - **- Real-time emotion detection using OpenCV and Keras**
 - **- CNN model trained on facial emotion dataset (48x48 grayscale images)**
 - **- Integration of computer vision for face detection**
 - **- Seven emotion classes: Angry, Disgust, Fear, Happy, Neutral, Sad, Surprise**

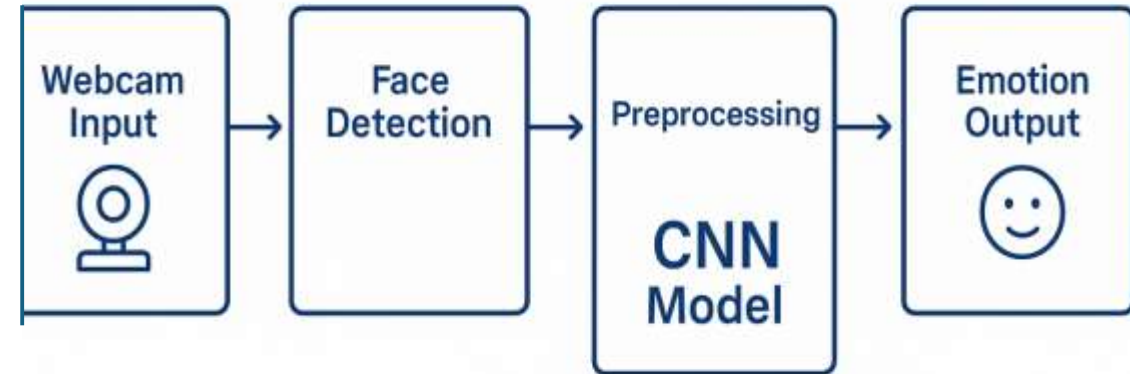
SYSTEM APPROACH

Technologies Used:

- Python
- OpenCV for face detection
- Keras (TensorFlow backend) for emotion classification
- NumPy for preprocessing
- Haar Cascade for detecting face regions

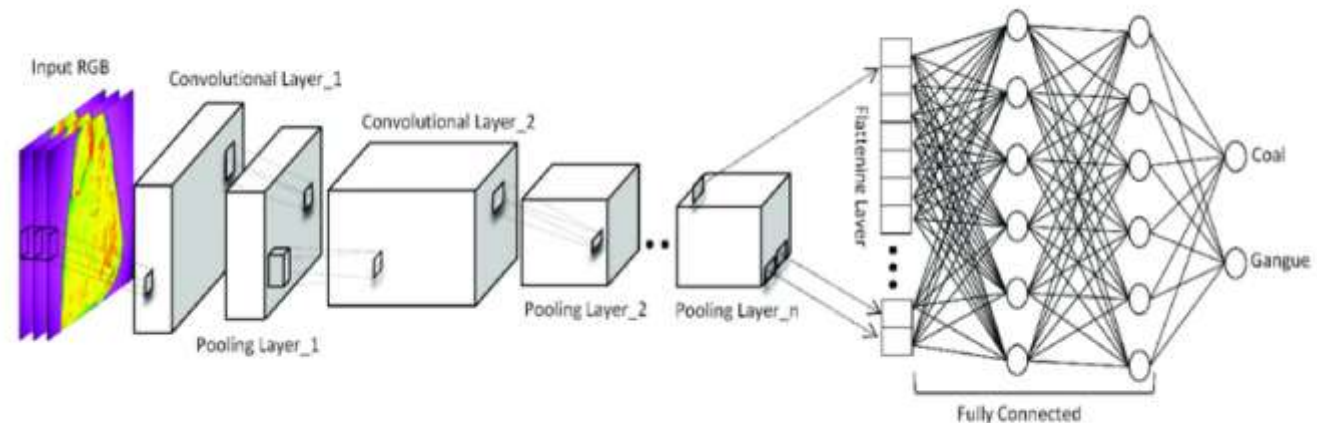
Libraries Required:

- keras
- opencv-python
- numpy



ALGORITHM & DEPLOYMENT

- Algorithm Used: Convolutional Neural Network (CNN)
- Training Process:
 - - Dataset: FER2013 (facial emotion recognition)
 - - Model Input: 48x48 grayscale images
 - - Output: 7 emotion classes via softmax activation
- Deployment:
 - - Model saved as .json and .h5 files
 - - Real-time video captured via OpenCV
 - - Model prediction displayed live on screen

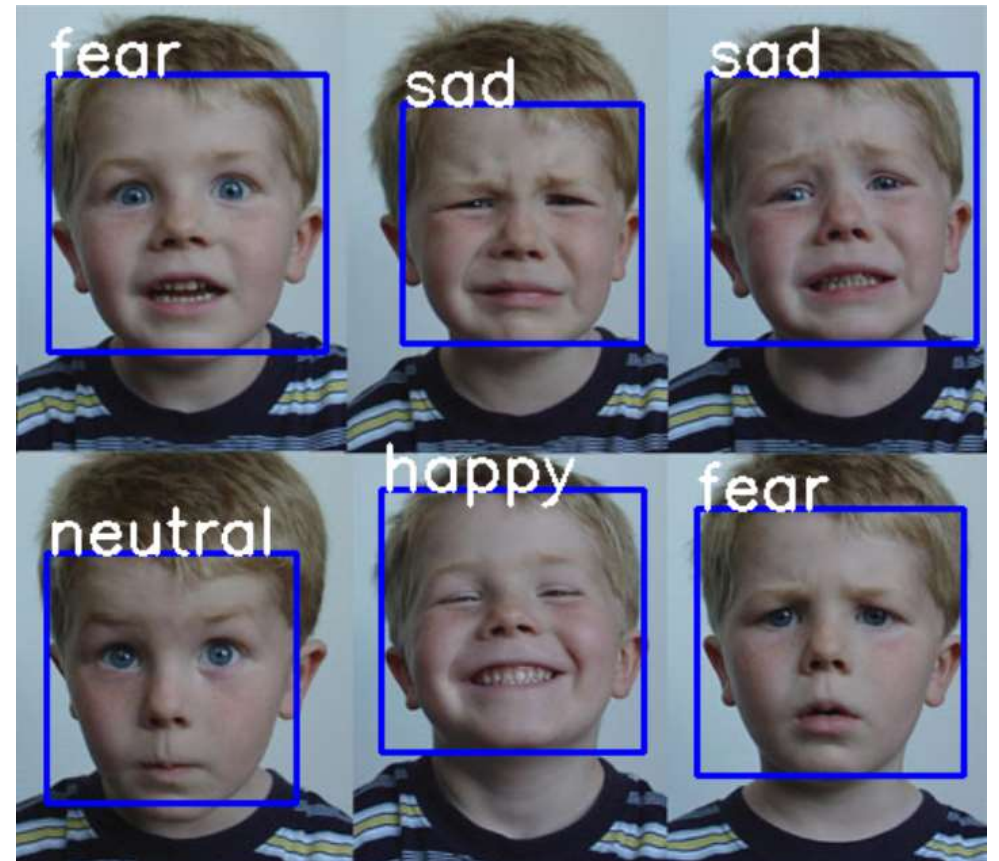


RESULT

The model successfully classifies emotions from real-time webcam input with high accuracy.

Example Output:

- Detected Face: Label = "Happy"
- Detected Face: Label = "Sad"
- Detected Face: Label = "Fear"
- Detected Face: Label = "Surprise"
- Detected Face: Label = "Neutral"
- Detected Face: Label = "Angry"



CONCLUSION

This system demonstrates a practical application of AI and deep learning for emotion recognition.

The project provided hands-on experience with:

- AI model training
- Computer vision integration
- Real-time deployment and testing

It shows how AI can enhance user interaction and mental health tracking through emotion-aware systems.



FUTURE SCOPE

- Train on larger and diverse datasets to improve accuracy
- Deploy as a web or mobile app using Flask or Android
- Integrate real-time dashboards or analytics
- Combine voice + facial emotion detection
- Explore edge AI for on-device emotion detection

REFERENCES

- [1] FER2013 Dataset
- [2] Keras Documentation
- [3] OpenCV Face Detection
- [4] Microsoft Azure AI Services
- GitHub Repository: <https://github.com/Suryansh-Dhama/REAL-TIME-facial-emotion-detection-USING-DEEP-LEARNING>
- Github Profile : <https://github.com/Suryansh-Dhama>

Thank you

A thick, hand-drawn orange line that spans the width of the text "Thank you" and extends slightly beyond it on both sides.