**A Project Report on**

**Sentiment Analysis using OpenCV**

**Submitted to :-** Mr Ashish Bajaj

**Submitted By:**

Kanav Mahajan (102303102)

Anush Sharma (102303103)

Ayush Kaushik (102303104)

**Date of Submission:** 6th May, 2025



**ACKNOWLEDGEMENT**

I would like to express my sincere gratitude to everyone who has contributed to the successful completion of this project on “emotion detection Using Machine Learning”.

First and foremost, I extend my deepest appreciation to **Mr** Ashish Bajaj**,** my project supervisor, for their invaluable guidance, insightful suggestions, and continuous encouragement throughout this journey. Their expertise and support have been instrumental in shaping this report.

Furthermore, I am grateful to my peers and colleagues for their constant motivation and constructive feedback. Their support has been crucial in refining the ideas and methodologies applied in this project.

Last but not least, I would like to thank my family and friends for their unwavering encouragement and belief in my abilities. Their support has been a source of strength throughout this endeavor

Real-Time Face Emotion Detection Using Python and CNN

# Abstract

Recognizing emotions from facial expressions has always been a fascinating challenge in computer vision. With the help of deep learning and tools like Python, we can now create systems that not only detect faces but also identify what emotions people are feeling in real time. This project focuses on building a simple, yet effective, emotion detection system using Convolutional Neural Networks (CNNs). It uses Python, OpenCV, and TensorFlow to process live video, detect faces, and classify emotions such as happy, sad, angry, and more. This kind of system can be useful in places like classrooms, healthcare, or even customer service.

# Introduction

Think about how we communicate every day—our words matter, but so do our facial expressions. Computers that can understand emotions can be much more helpful and responsive. For example, a teaching app that knows when a student looks confused can offer extra help.

This project explores how to use a deep learning model (specifically, a CNN) to read emotions from facial expressions in real time. The goal is to build something that works reliably using just a webcam and open-source libraries.

Emotion recognition is becoming increasingly important in areas such as online education, gaming, surveillance, and virtual assistants. Systems that understand human emotions can improve experiences, adapt in real-time, and even support mental health monitoring.

# Problem Statement

It's not easy for machines to understand human emotions. Faces can look different under different lighting, from different angles, or when people are wearing glasses or masks. Some emotions—like fear and disgust—are especially hard to tell apart. This project aims to create a system that can work through these challenges and still detect emotions reasonably well, even in real-time video.

Moreover, many existing emotion recognition systems are either too slow for real-time use or require high-end hardware. There's a clear need for a solution that balances speed, accuracy, and efficiency for everyday devices.

# Objectives

- Build a simple, real-time emotion detection system using Python and CNNs.

- Use a popular dataset to train the model to recognize basic emotions.

- Connect the model with a webcam so it can classify emotions live.

- Test how well the system works in different settings.

- Think about where this kind of system might actually be useful.

- Make the system light enough to run on regular laptops.

# Methodology

\*\*1. Tools Used\*\*

- Python

- OpenCV

- TensorFlow/Keras

- NumPy and Matplotlib

\*\*2. Dataset and Preprocessing\*\*

We used the FER2013 dataset...

\*\*3. CNN Model Structure\*\*

- Conv2D (32 filters) + ReLU + MaxPooling

- Conv2D (64 filters) + ReLU + MaxPooling

- Dropout (0.25)...

\*\*4. Training\*\*

- Loss: Categorical cross-entropy

- Optimizer: Adam

- Epochs: 50

\*\*5. Real-Time Application\*\*

- OpenCV captures webcam input...

\*\*6. Evaluation Metrics\*\*

- Accuracy

- Confusion Matrix

- FPS

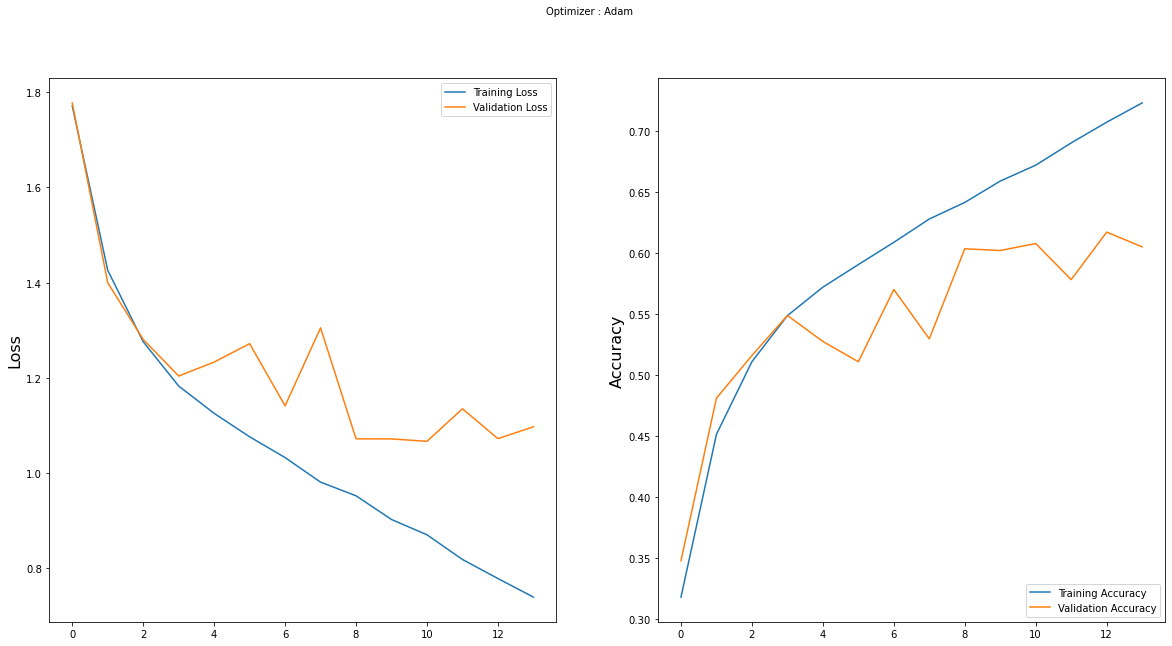
# Results

The model reached about 70% accuracy on the validation set. It did best on emotions like “happy” and “neutral.” Emotions like “fear” and “disgust” were harder to get right.

- Real-time: ~15–20 FPS

- Best in good lighting

- ~72% accuracy in live tests



# Screenshots and Visuals

(Figure 1) Model Architecture Diagram

(Figure 2) Training vs Validation Accuracy Plot

(Figure 3) Sample Predictions

(Figure 4) Confusion Matrix Heatmap

# Conclusion

This project proves that with Python and open-source tools, real-time emotion recognition is achievable.

\*\*Future Scope:\*\*

- Mobile deployment

- Facial landmarks

- Voice + face integration

- Group emotion recognition

# References

1. Goodfellow, I., Bengio, Y., & Courville, A. (2016). \*Deep Learning\*.

2. FER2013 Dataset. Kaggle

3. OpenCV Docs

4. TensorFlow Docs

5. Chollet, F. (2017). \*Deep Learning with Python\*.

6. Ekman, P. & Friesen, W. (1978). \*Facial Action Coding System\*.