Metropolitan University

Department of Software Engineering

Project Documentation

Course Title: NEURAL NETWORK AND DEEP LEARNING LAB

Course code: SWE-458

Project Title: Custom CNN Models for Facial Emotion Recognition

Dataset: FER2013 (Facial Expression Recognition 2013)

Submitted to:

Al Akram Chowdhury

Lecturer

Dept. of Software Engineering

Submitted by:

Tushar Sinha

221-134-005

Dipto Deb Nath

221-134-013

Software Engineering, 2nd Batch.

Date of Submission: 26th April 2025

Overview

This project implements and compares five custom Convolutional Neural Network (CNN) models to classify facial emotions using the FER2013 dataset. Each model has different architectural designs and complexities to evaluate their performance and efficiency for emotion recognition tasks.

The models developed are:

1. **VisageNet**
2. **EmotiNet**
3. **MoodCNN**
4. **NeuroMood**
5. **FERLight**

All models are built using **TensorFlow/Keras** frameworks.

Dataset: FER2013

Source: Kaggle FER2013

Description:

- 48x48 grayscale facial images.
- Labeled with one of 7 emotions:
    - 0 = Angry
    - 1 = Disgust
    - 2 = Fear
    - 3 = Happy
    - 4 = Sad
    - 5 = Surprise
    - 6 = Neutral

Dataset is split into training, validation, and test sets.

Model 1: VisageNet

File: `VisageNet_model.ipynb`

Architecture:

- Sequential CNN layers with batch normalization after convolutions.
- Aggressive downsampling through pooling.
- Smaller dense layers compared to NeuroMood.
- Higher dropout rates.

Highlights:

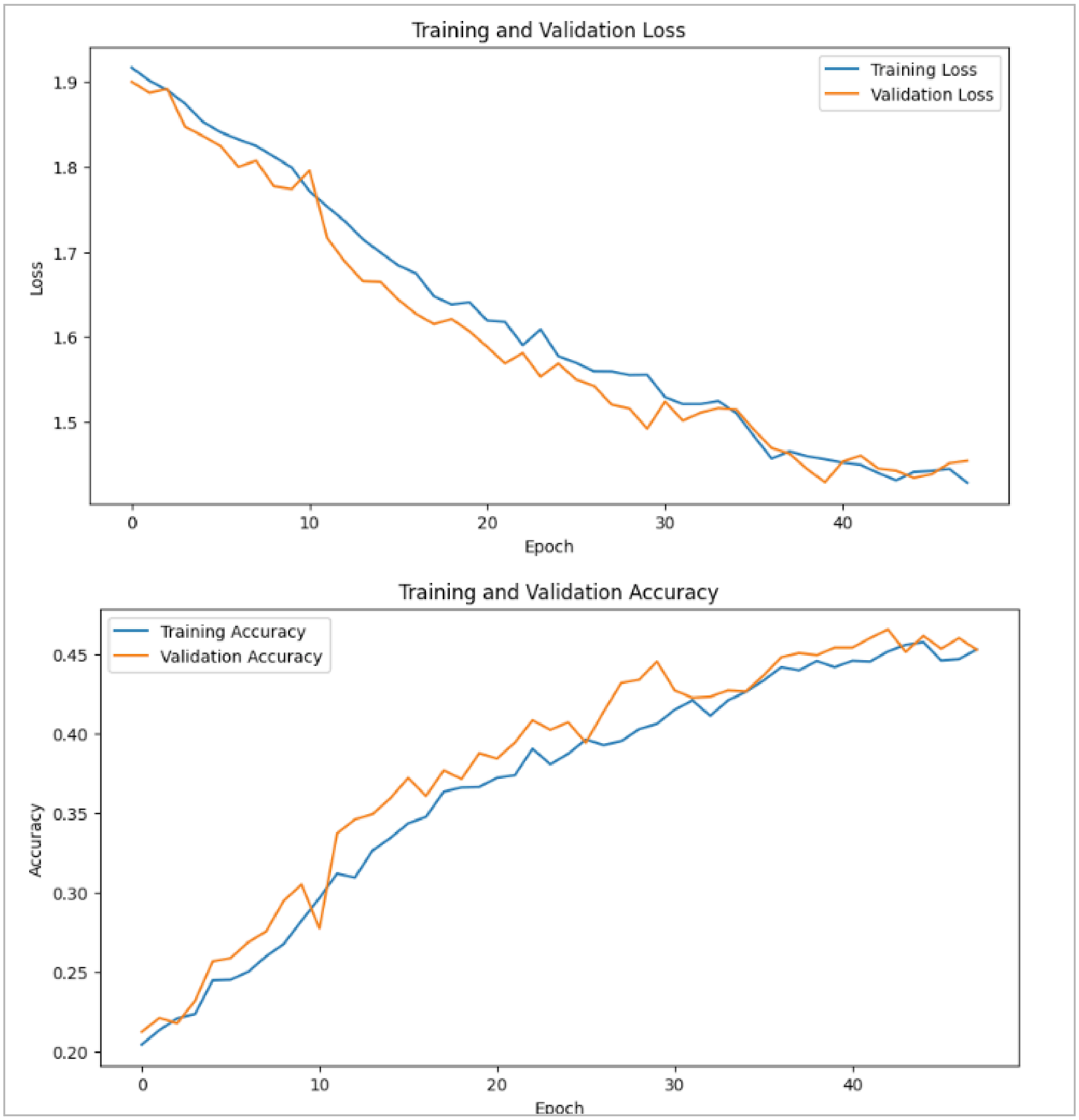
- Faster convergence thanks to batch normalization.
- Lighter and faster than NeuroMood.
- Good balance between performance and efficiency.

Model Architecture Summary

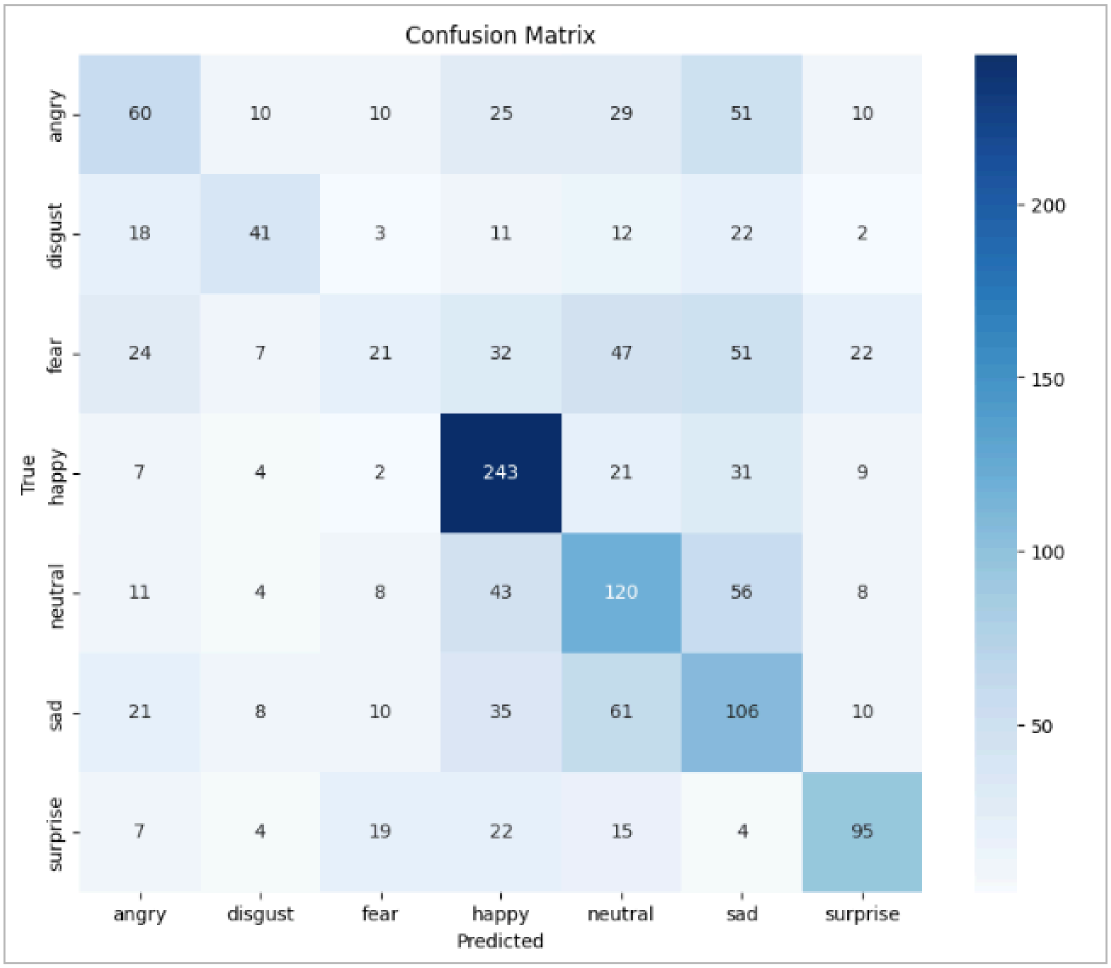
Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d (MaxPooling2D)	(None, 31, 31, 32)	0
conv2d_1 (Conv2D)	(None, 29, 29, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 14, 14, 64)	0
conv2d_2 (Conv2D)	(None, 12, 12, 128)	73,856
max_pooling2d_2 (MaxPooling2D)	(None, 6, 6, 128)	0
flatten (Flatten)	(None, 4608)	0
dense (Dense)	(None, 128)	589,952
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 7)	903

Training Accuracy and Loss Curves



Confusion Matrix



Model 2: EmotiNet

File: `EmotiNet_model.ipynb`

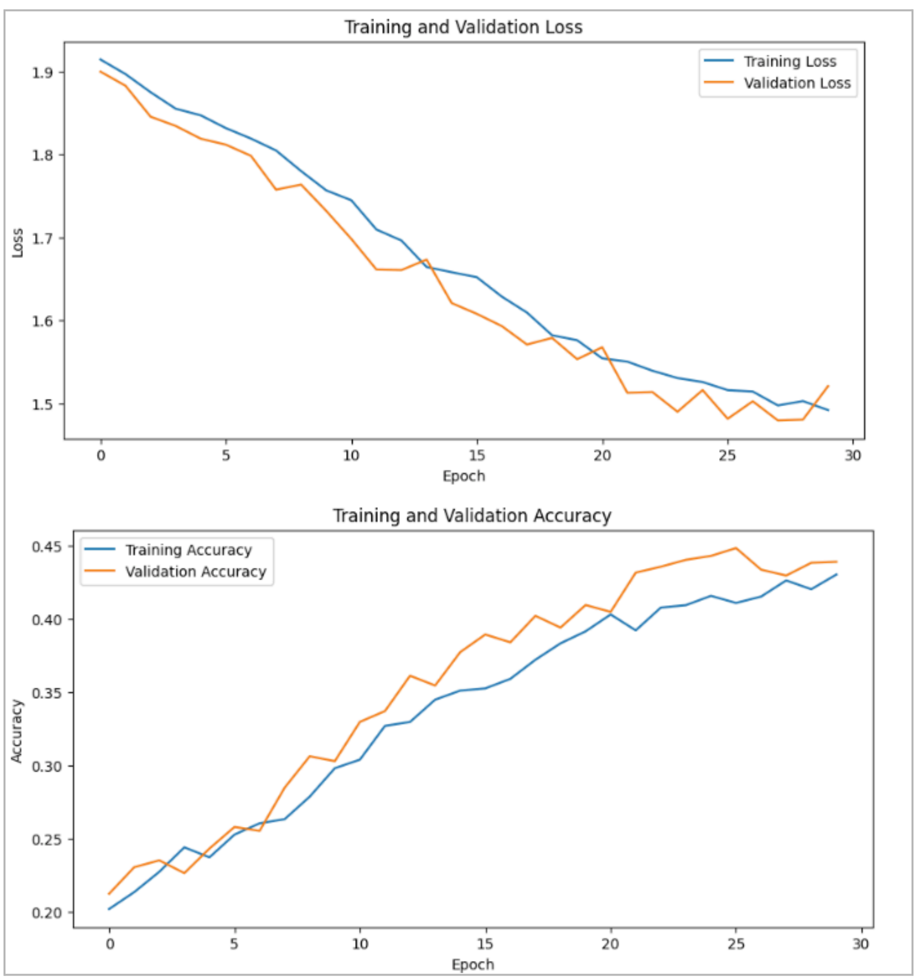
Architecture:

- CNN layers with smaller kernel sizes.
- GlobalAveragePooling2D instead of Flatten to reduce parameter count.
- Compact dense layers.
- Final softmax classification.

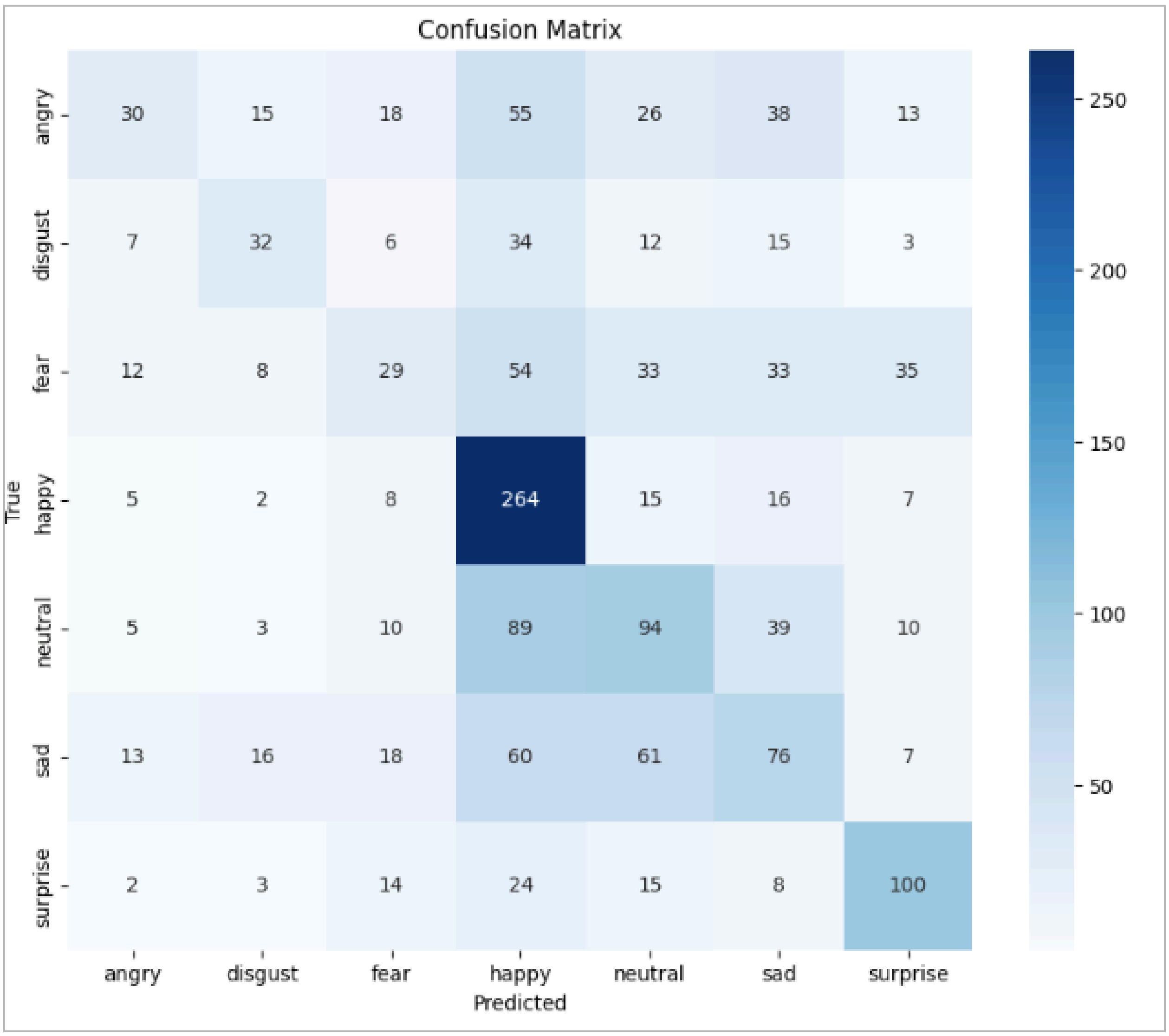
Highlights:

- Most lightweight model among the three.
- Suitable for mobile or embedded applications.
- Quick training and low memory footprint.

Training Accuracy and Loss Curves



Confusion Matrix



Model 3: MoodCNN

File: `MoodCNN_model.ipynb`

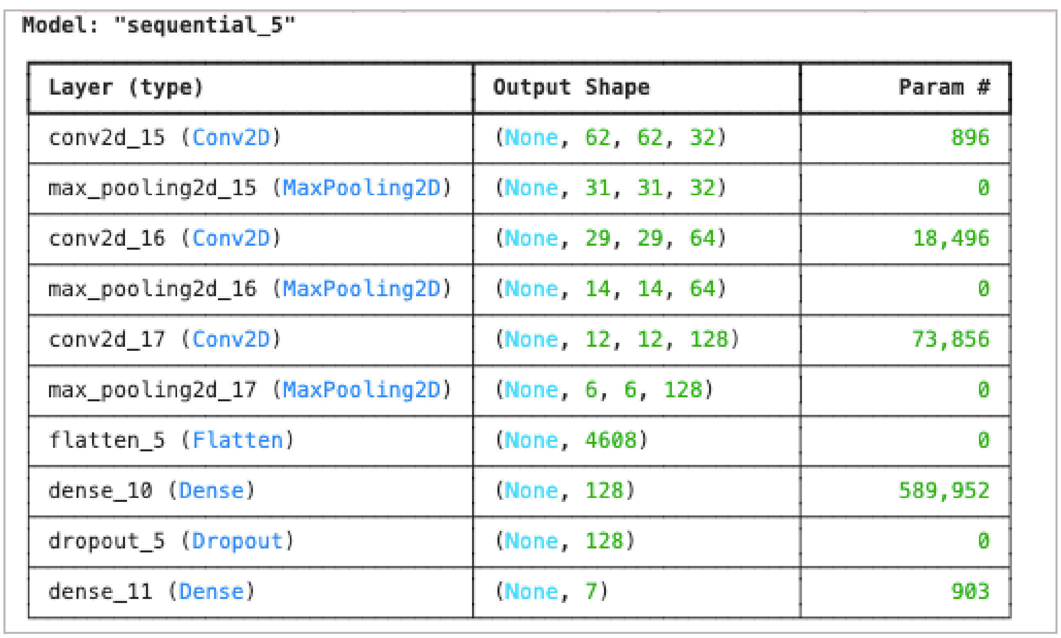
Architecture:

- Fewer convolutional layers but deeper filters.
- Uses both MaxPooling and AveragePooling.
- Introduces early dropout in convolution blocks.
- Larger dense layers for final prediction.

Highlights:

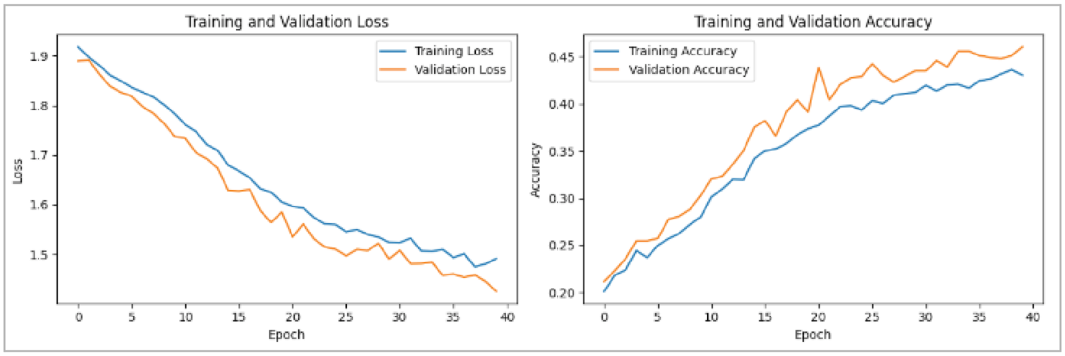
- Balances depth and width of the network.
- Early dropout helps in regularization during convolution stages.
- Designed to perform well without being overly deep.

Model Architecture Summary

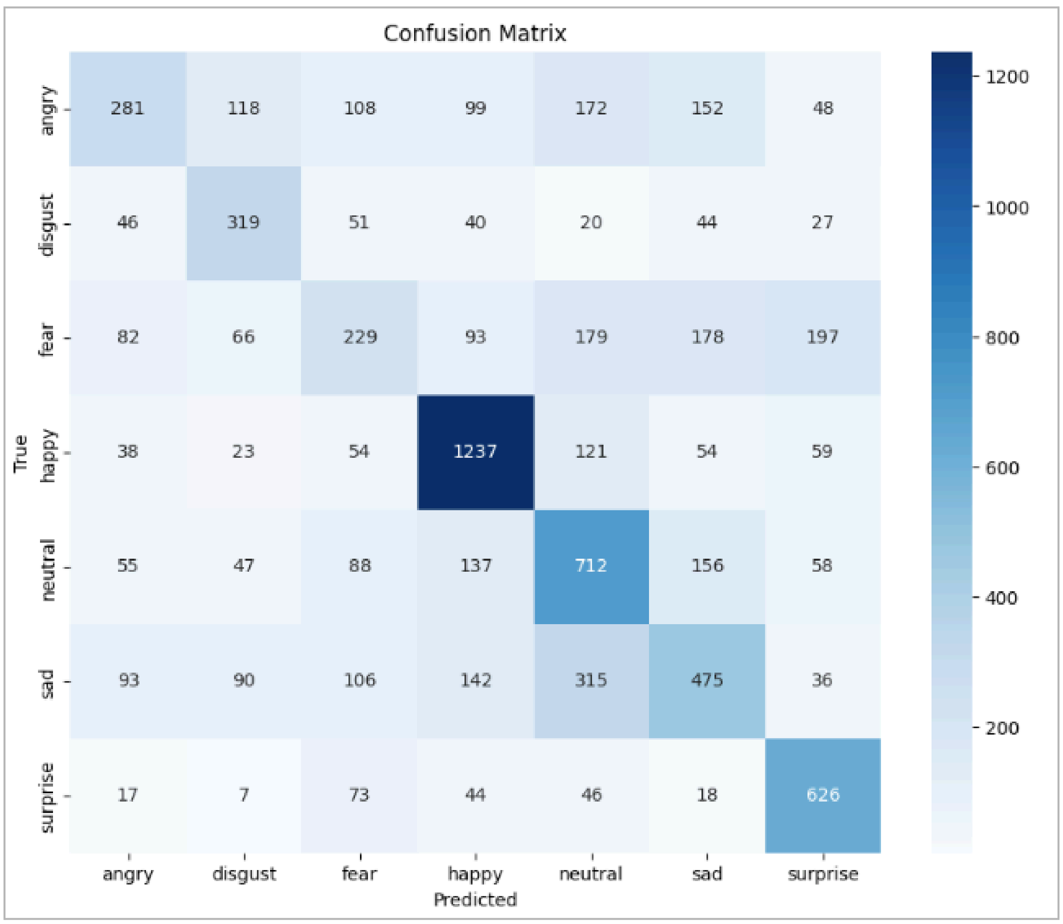

Model: "sequential_5"

Layer (type)	Output Shape	Param #
conv2d_15 (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d_15 (MaxPooling2D)	(None, 31, 31, 32)	0
conv2d_16 (Conv2D)	(None, 29, 29, 64)	18,496
max_pooling2d_16 (MaxPooling2D)	(None, 14, 14, 64)	0
conv2d_17 (Conv2D)	(None, 12, 12, 128)	73,856
max_pooling2d_17 (MaxPooling2D)	(None, 6, 6, 128)	0
flatten_5 (Flatten)	(None, 4608)	0
dense_10 (Dense)	(None, 128)	589,952
dropout_5 (Dropout)	(None, 128)	0
dense_11 (Dense)	(None, 7)	903

Training Accuracy and Loss Curves



Confusion Matrix



Model 4: NeuroMood

File: `NeuroMood_model.ipynb`

Architecture:

- Multiple convolutional layers with increasing filter sizes.
- Max pooling after each convolution block.
- Dropout layers for regularization.
- Fully connected dense layers.
- Final softmax activation for classification.

Highlights:

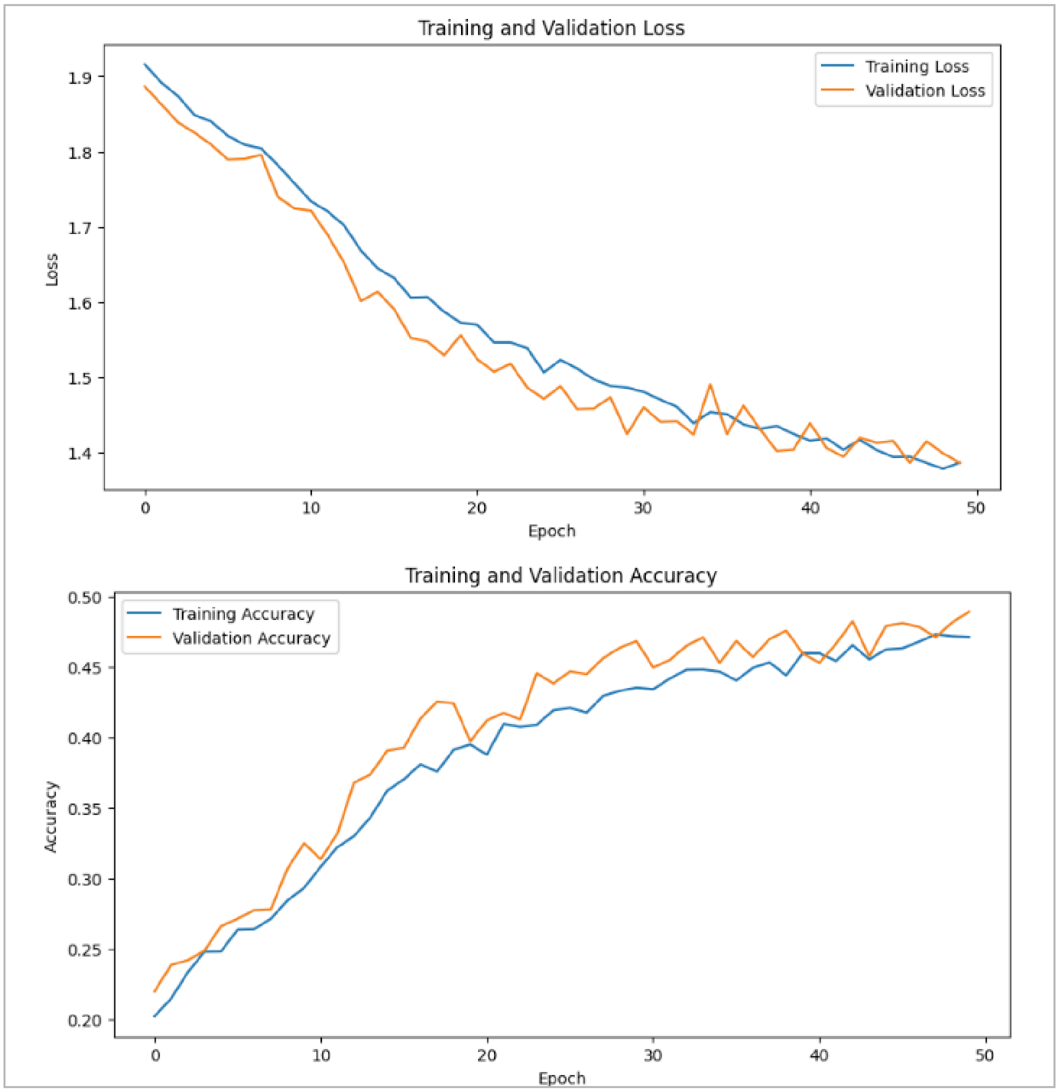
- Deeper network structure.
- Strong feature extraction focus.
- Regularized for better generalization.

Model Architecture Summary

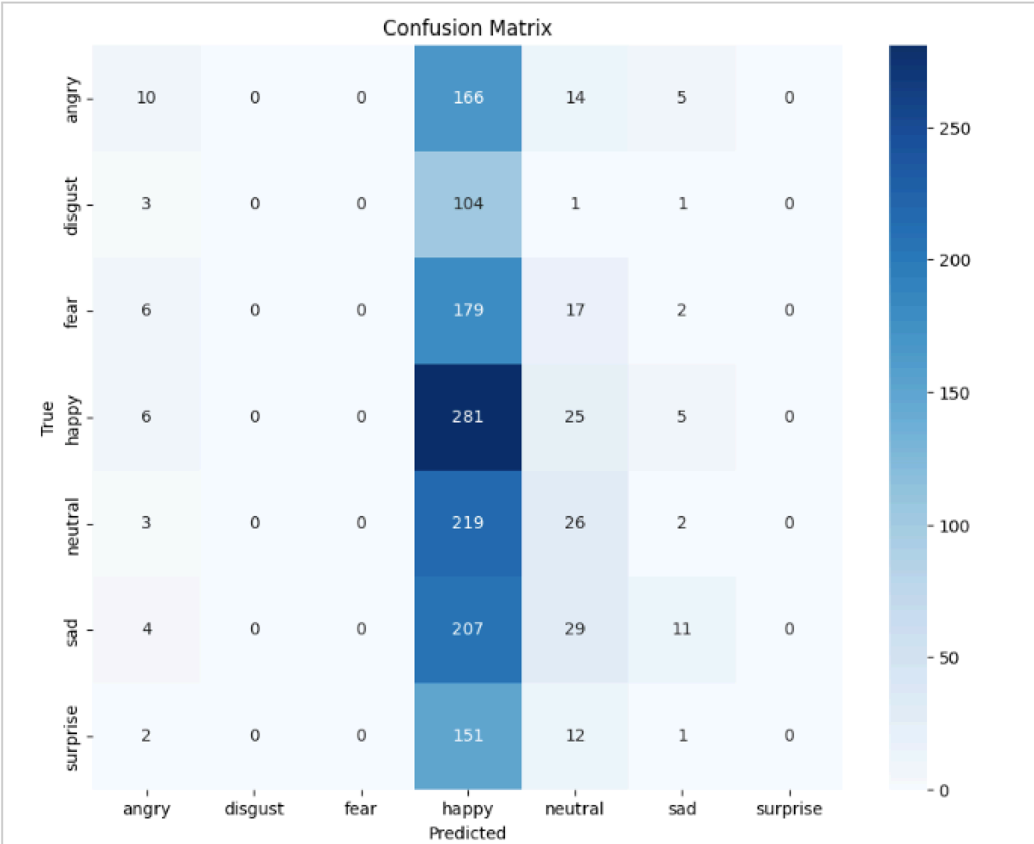
Model: "sequential_3"

Layer (type)	Output Shape	Param #
conv2d_9 (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d_9 (MaxPooling2D)	(None, 31, 31, 32)	0
conv2d_10 (Conv2D)	(None, 29, 29, 64)	18,496
max_pooling2d_10 (MaxPooling2D)	(None, 14, 14, 64)	0
conv2d_11 (Conv2D)	(None, 12, 12, 128)	73,856
max_pooling2d_11 (MaxPooling2D)	(None, 6, 6, 128)	0
flatten_3 (Flatten)	(None, 4608)	0
dense_6 (Dense)	(None, 128)	589,952
dropout_3 (Dropout)	(None, 128)	0
dense_7 (Dense)	(None, 7)	903

Training Accuracy and Loss Curves



Confusion Matrix



Model 5: FERLight

File: `FERLight_model.ipynb`

Architecture:

- Very shallow CNN model with minimal parameters
- Small convolutional blocks with aggressive pooling.
- Direct use of GlobalAveragePooling2D for flattening.
- Lightweight dense layers.

Highlights:

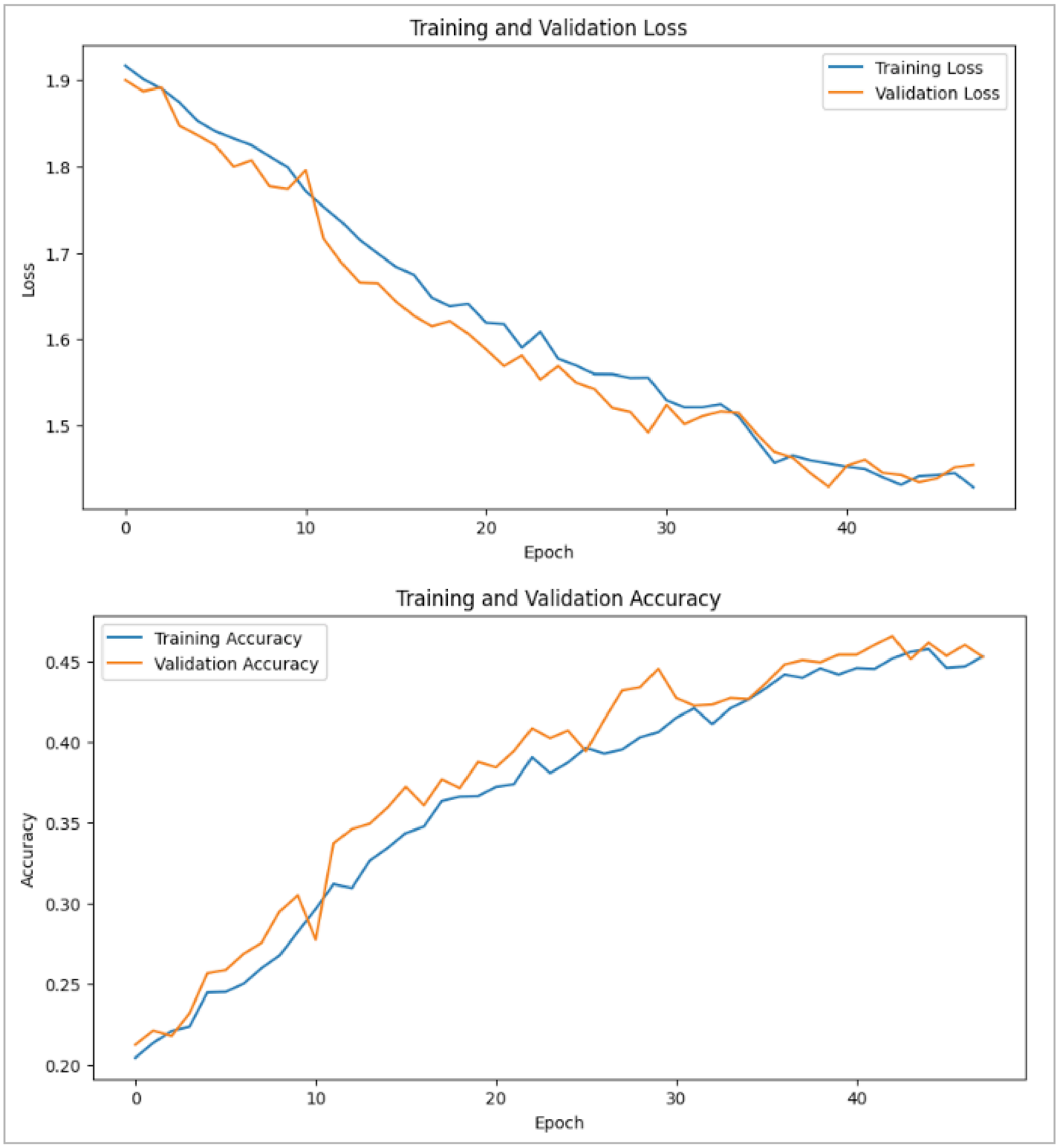
- **Smallest and fastest** among all models.
- Prioritizes efficiency and fast inference time.
- Best suited for low-resource environments like web apps or mobile devices.

Model Architecture Summary

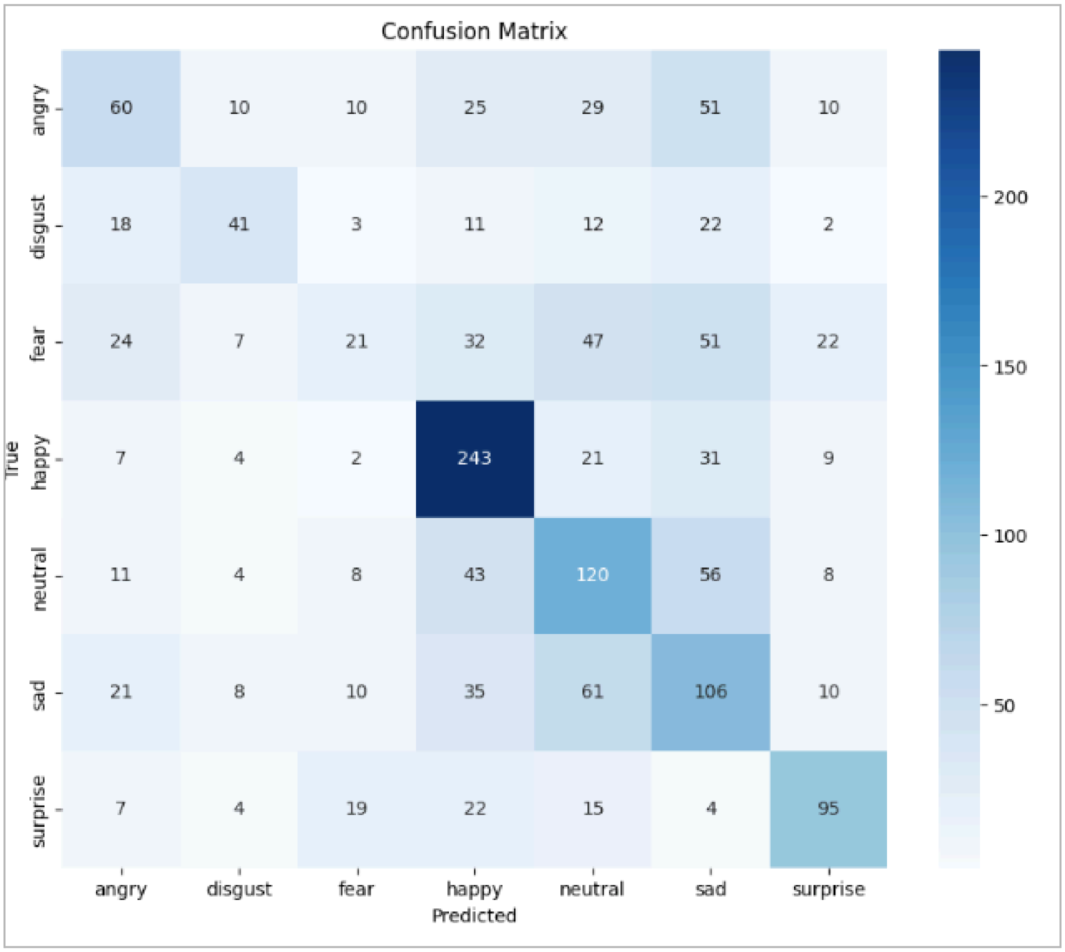
Model: "sequential_2"

Layer (type)	Output Shape	Param #
conv2d_9 (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d_6 (MaxPooling2D)	(None, 31, 31, 32)	0
conv2d_10 (Conv2D)	(None, 29, 29, 64)	18,496
max_pooling2d_7 (MaxPooling2D)	(None, 14, 14, 64)	0
conv2d_11 (Conv2D)	(None, 12, 12, 128)	73,856
max_pooling2d_8 (MaxPooling2D)	(None, 6, 6, 128)	0
flatten_2 (Flatten)	(None, 4608)	0
dense_4 (Dense)	(None, 128)	589,952
dropout_5 (Dropout)	(None, 128)	0
dense_5 (Dense)	(None, 7)	903

Training Accuracy and Loss Curves



Confusion Matrix



Common Training Details

- **Loss Function:** Categorical Crossentropy
- **Optimizer:** Adam
- **Metrics:** Accuracy
- **Input Shape:** (48, 48, 1) — grayscale images

Conclusion

This project explores and contrasts multiple CNN architectures for facial emotion recognition using the FER2013 dataset. Each model targets specific needs:

Model	Focus
NeuroMood	Deep feature extraction and high accuracy
VisageNet	Balanced speed and performance
EmotiNet	Lightweight and fast
MoodCNN	Balanced architecture with early regularization
FERLight	Minimalist design for fastest performance

Comparative performance (accuracy, training speed, confusion matrices) helps users choose the best model depending on requirements like accuracy, speed, or deployment constraints.

Appendix

Link to GitHub: [Custom CNN Models for Facial Emotion Recognition]