Face Expression Analysis (using NN)

Presented by: Ashish Kumar (23B1028) and
Velamakanni Satwik (23B0998)

Introduction

- Facial expressions play a major role in human communication
- The objective of this project is to automatically detect emotions from facial images
- Deep learning enables robust recognition even under variations in lighting, angle, or background
- Applications include human–computer interaction, surveillance, healthcare, and emotion-aware AI systems

Problem Statement

- Build a system that classifies images into 7 emotion categories:
  Angry, Disgust, Fear, Happy, Neutral, Sad, Surprise

- Handle challenges like low-resolution images, noise, and class imbalance.

- Achieve a high accuracy and reliable performance on unseen data.

Dataset (FER-2013)

- ➢ Dataset used: **FER2013**, a widely used benchmark for emotion recognition (here)
- ➢ Contains **35k+ grayscale images** in **48×48 resolution** and 7 emotion labels
- ➢ Images include variations in lighting, pose, age, gender.



Model Architecture 🤖

A custom Convolutional Neural Network with:

- **3 Convolutional Blocks**
  - Conv2D → BatchNorm → Conv2D → BatchNorm → MaxPool → Dropout
- Filter sizes: **64**, **128**, **256**
- Dropout used to reduce overfitting

- **Fully Connected Layers**
  - Dense(512)
  - Dense(256)
  - Dropout layers
- **Output Layer:** Softmax over 7 classes
- Model trained end-to-end on 48×48×3 input images

Reference: https://arxiv.org/pdf/2105.03588

Training Setup

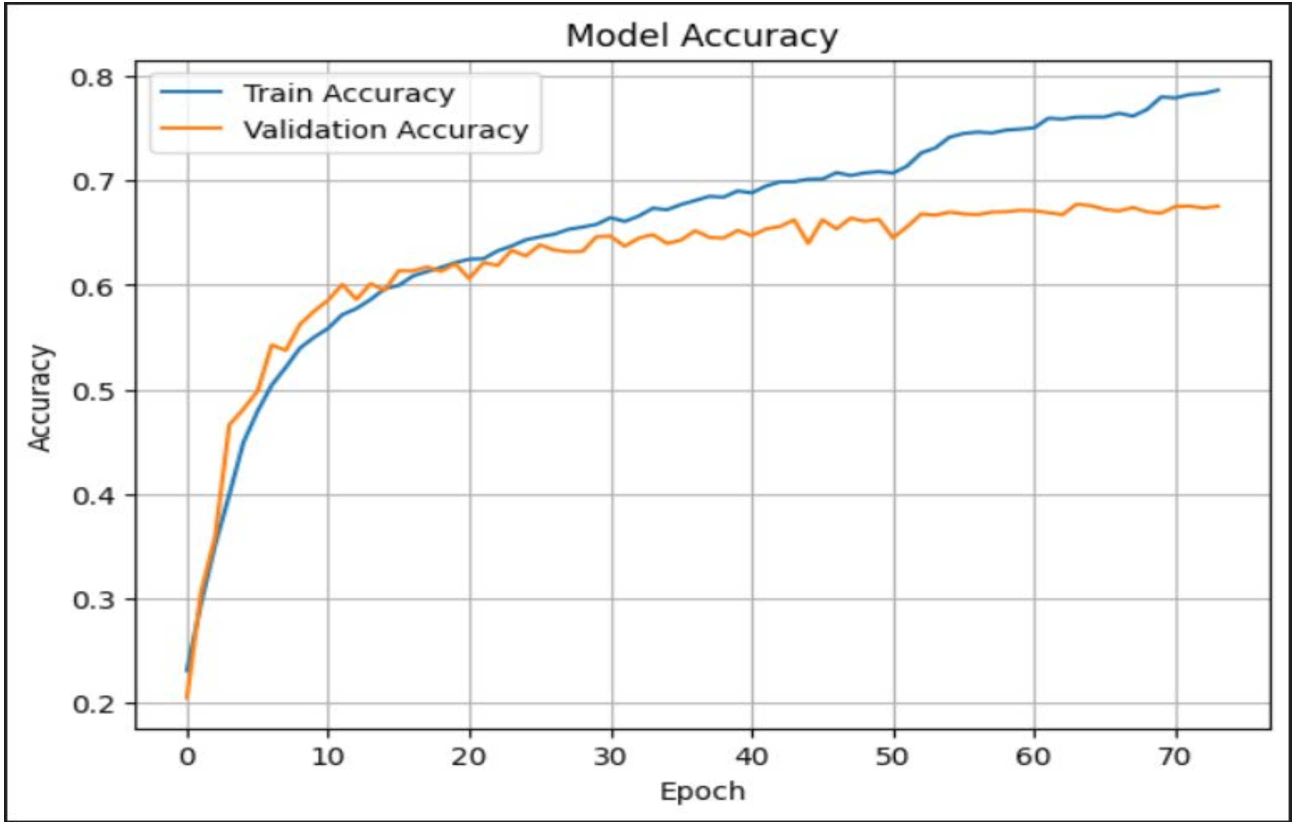
- **Optimizer:** SGD with momentum + Nesterov
- **Learning Rate:** 0.01 with ReduceLROnPlateau
- **Loss Function:** Sparse Categorical Crossentropy
- **Batch Size:** 32
- **Epochs:** Up to 200 (EarlyStopping prevents overtraining)
- **Callbacks:**
  - EarlyStopping
  - ReduceLROnPlateau

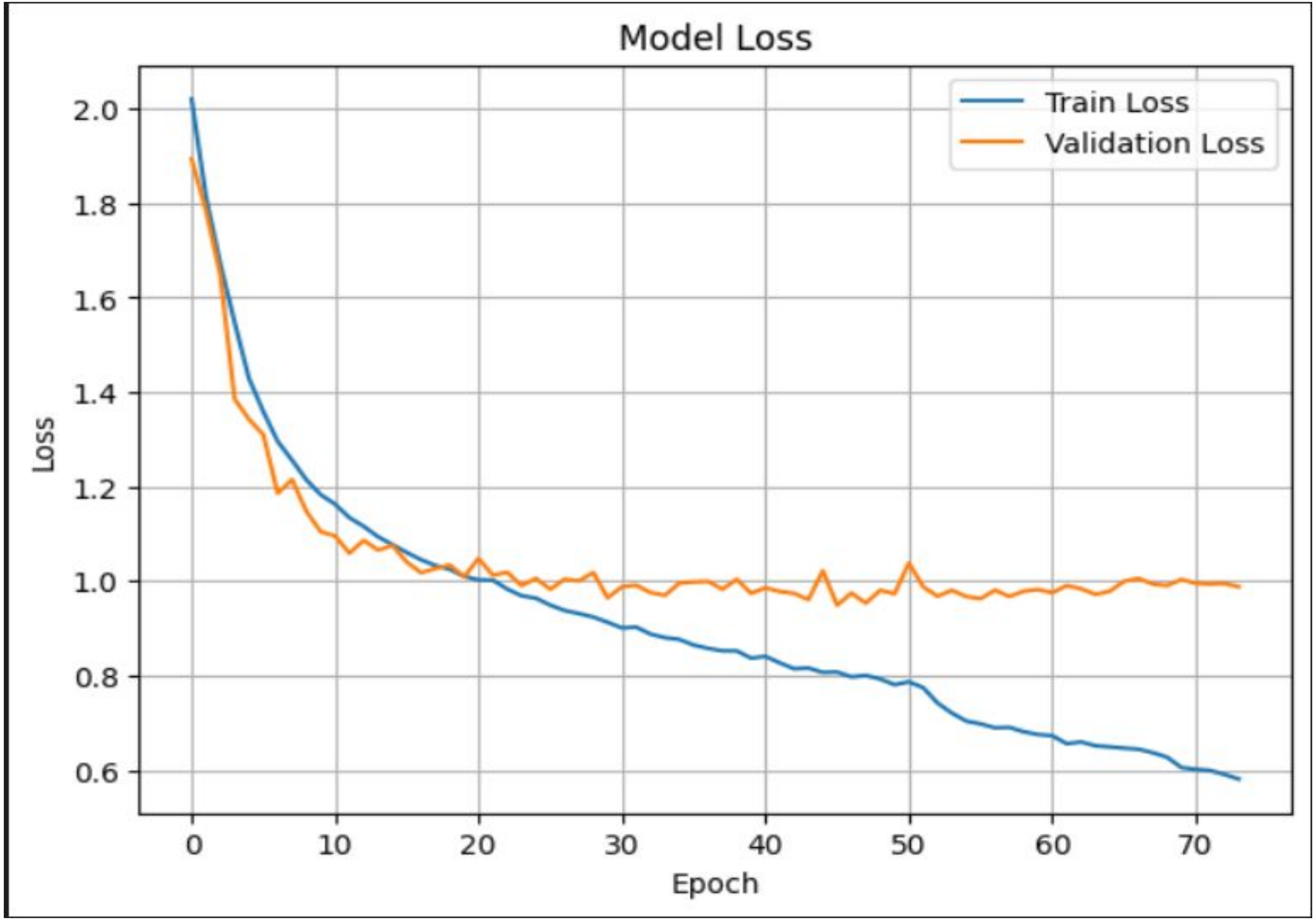
Training Results

- **Final Test Accuracy: 67.55%**
- **Final Test Loss:** 0.9881
- Model shows strong generalization for “Happy”, “Angry” and “Surprise”
- Performance drops slightly on separating classes like “Neutral” and “Sad”
- Training & validation accuracy curves indicate good convergence

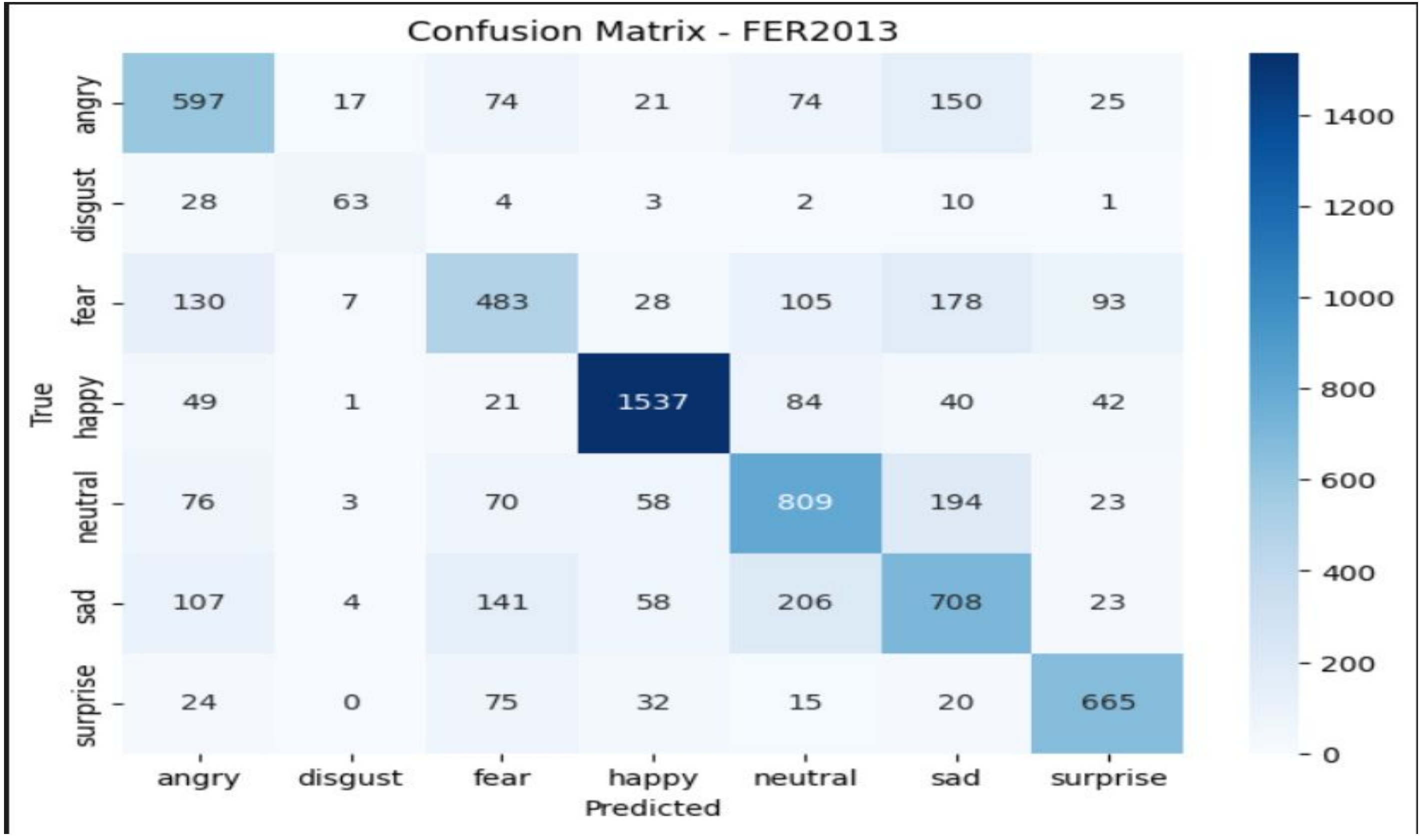
Fun Fact: The typical human performance is an estimated 65% ± 5%

Let’s see some graphs 🥳




Model Loss
Train Loss
Validation Loss
Loss
Epoch
2.0
1.8
1.6
1.4
1.2
1.0
0.8
0.6
0
10
20
30
40
50
60
70

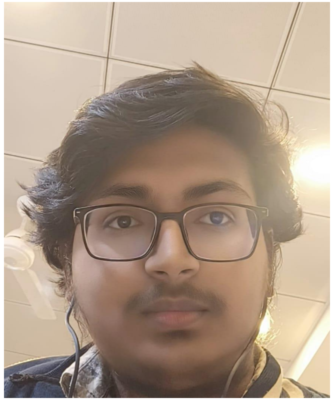
Confusion Matrix & Report



Classification Report:

	precision	recall	f1-score	support
angry	0.59	0.62	0.61	958
disgust	0.66	0.57	0.61	111
fear	0.56	0.47	0.51	1024
happy	0.88	0.87	0.88	1774
neutral	0.62	0.66	0.64	1233
sad	0.54	0.57	0.56	1247
surprise	0.76	0.80	0.78	831
accuracy			0.68	7178
macro avg	0.66	0.65	0.65	7178
weighted avg	0.68	0.68	0.68	7178

Let's see some photos 👻



```
Enter image path:  test-1.jpeg
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 160ms/step
INPUT IMAGE: test-1.jpeg
Predicted Emotion: Neutral (78.04% confidence)

Class-wise Confidence:
Angry       : 0.24%
Disgust     : 0.01%
Fear        : 0.35%
Happy       : 18.79%
Neutral     : 78.04%
Sad         : 2.24%
Surprise    : 0.35%
'neutral'
```

```
Enter image path:  t1.jpg
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 163ms/step
INPUT IMAGE: t1.jpg
Predicted Emotion: Happy (99.80% confidence)

Class-wise Confidence:
Angry       : 0.00%
Disgust     : 0.00%
Fear        : 0.00%
Happy       : 99.80%
Neutral     : 0.20%
Sad         : 0.00%
Surprise    : 0.00%
'happy'
```

Demo Time !!!

Conclusion

- ➢ Developed a deep learning-based Facial Expression Recognition system
- ➢ Achieved **67.55% accuracy** on unseen (test) FER2013, which is competitive for this dataset
- ➢ Built a robust CNN with augmentation, batch normalization, and dropout
- ➢ Results demonstrate strong performance on common expressions

Future Work

- ➢ Use advanced architectures: ResNet, EfficientNet, MobileNet
- ➢ Apply face detection before classification (MTCNN / RetinaFace)
- ➢ Train on larger, cleaner datasets (AffectNet, RAF-DB)
- ➢ Deploy as a real-time system (webcam / mobile app)

Thank You 😊