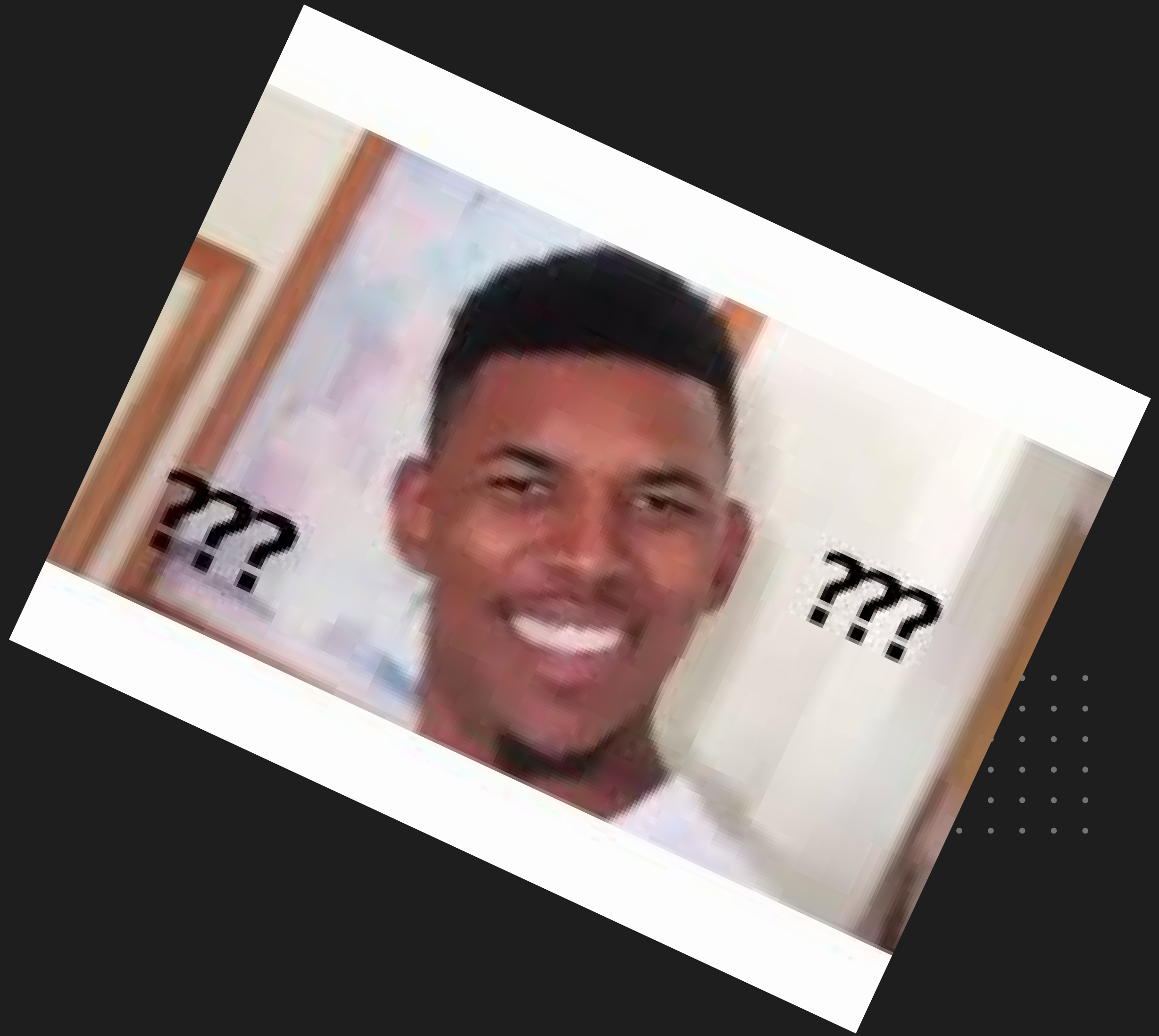
GROUP 11



FACIAL EMOTION RECOGNTION

BY USING CNN

Course Code: AIC205A
Course Title: Machine Learning – 2
Course Instructor: Mrs. Shaista Tarannum

Faculty of Engineering and Technology
Ramaiah University of Applied Sciences

GROUP11

FACIAL EMOTION CLASSIFCATION

SUPERVISED IMAGE CLASSIFICATION USING CNN WORKFLOW

Objective: Automatically recognize human facial emotions from images

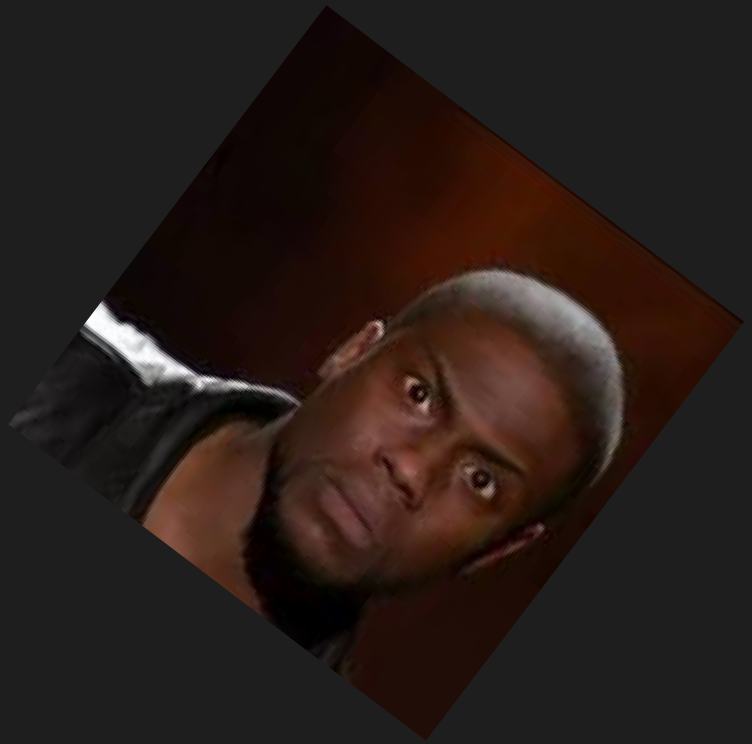
Input: 48 × 48 grayscale facial images

Output: One of 7 emotion classes

- Angry
- Disgust
- Fear
- Happy
- Sad
- Surprise
- Neutral

Dataset Used: FER2013 (Kaggle)

MOTIVATION AND APPLICATIONS



- Facial emotions convey important non-verbal human information
- Automatic emotion recognition improves human–computer interaction

Useful in:

- Mental health monitoring Online learning & student engagement analysis
- Customer feedback analysis
- Surveillance and security systems
- CNNs are effective because they learn spatial facial features automatically

MAIN CONTENT

- Dataset Name: FER2013 (Facial Expression Recognition 2013)
- Source: Kaggle
- Image Type: 48 × 48 grayscale facial images
- Number of Classes: 7 emotion categories

Dataset Split:

Training set: ~28,700 images

Test set: ~7,100 images

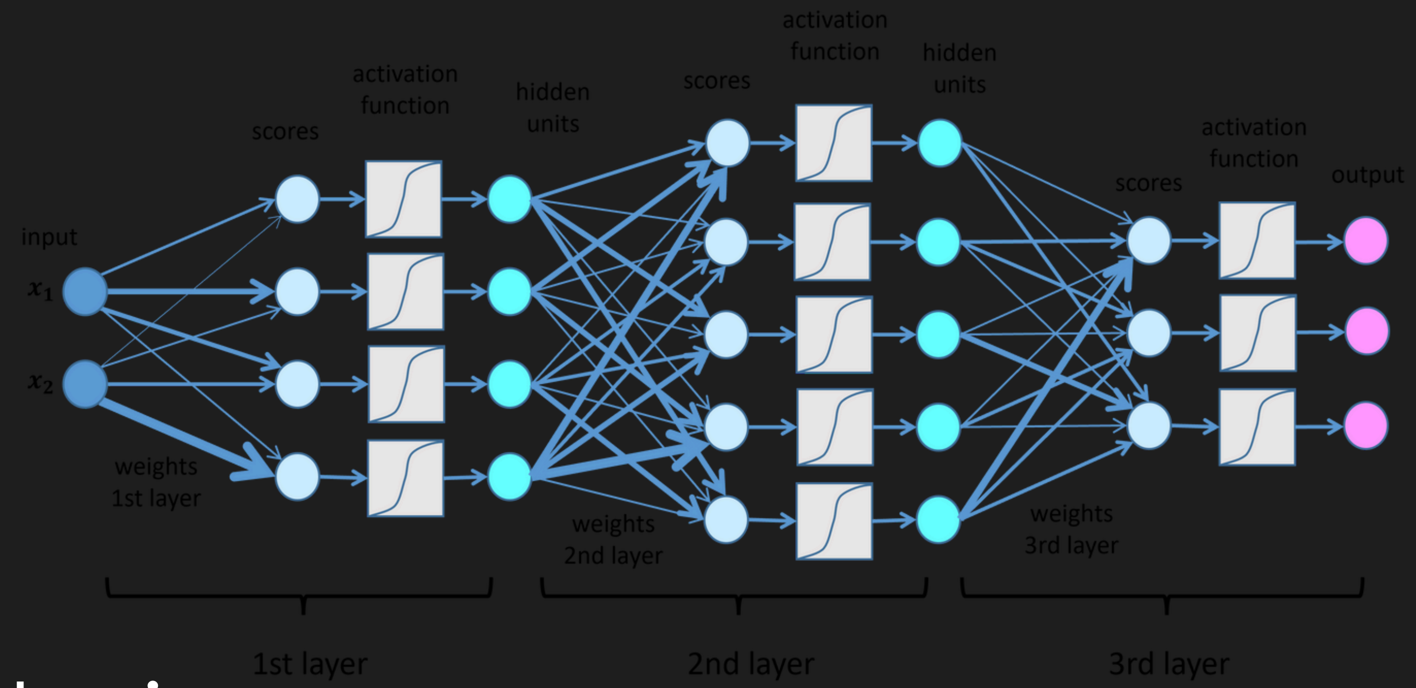
DATASET DESCRIPTION FER2013

Challenges in Dataset

- Low-resolution facial images
- Class imbalance across emotions
- Similar facial expressions between some classes (e.g., Fear vs Surprise)

GROUP 11

CONCEPTUAL UNDERSTANDING



Types of Neural Networks

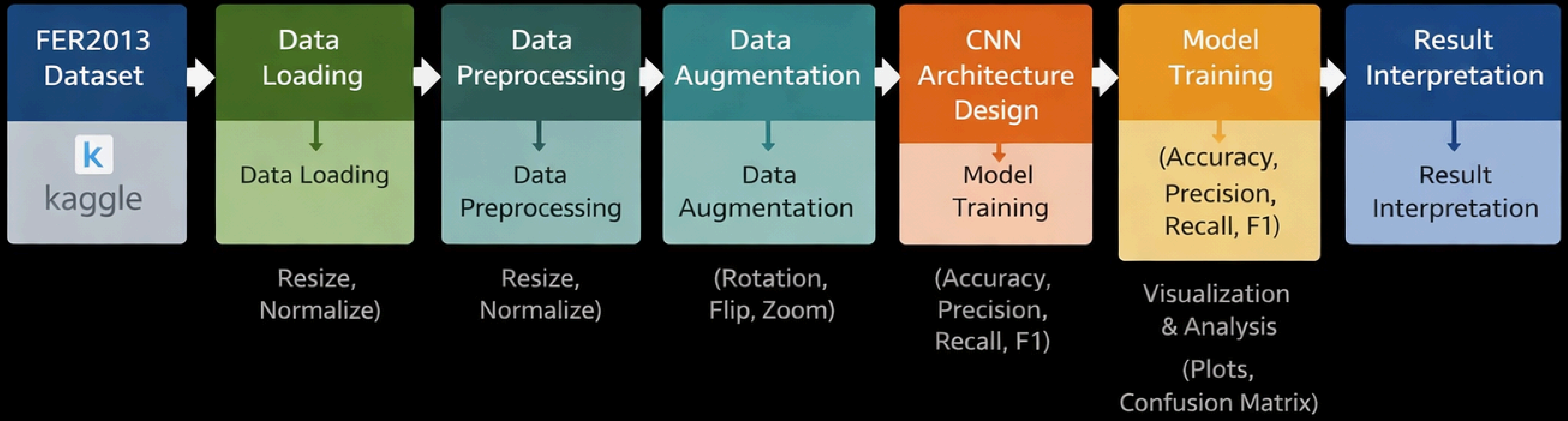
- ANN (Artificial Neural Network): Used for tabular data and basic classification/regression problems
- CNN (Convolutional Neural Network): Best suited for image data due to spatial feature extraction
- RNN (Recurrent Neural Network): Used for sequential data such as text, speech, and time-series

IDE USED:

Essential Libraries Used

- TensorFlow / Keras – Framework to build, train, and evaluate CNN models
- NumPy – Handles numerical operations and array manipulation
- Matplotlib / Seaborn – Used for plotting accuracy, loss, and confusion matrix
- OpenCV (cv2) – Image loading, resizing, and preprocessing
- Scikit-learn – Computes evaluation metrics like precision, recall, and F1-score

End-to-End Deep Learning Workflow



This workflow was implemented manually and also using AI assistance for comparison.

GROUP 11

DATA PREPARATION AND PREPROCESSING

Image Resizing:

All images resized to 48 × 48 pixels to maintain uniform input size

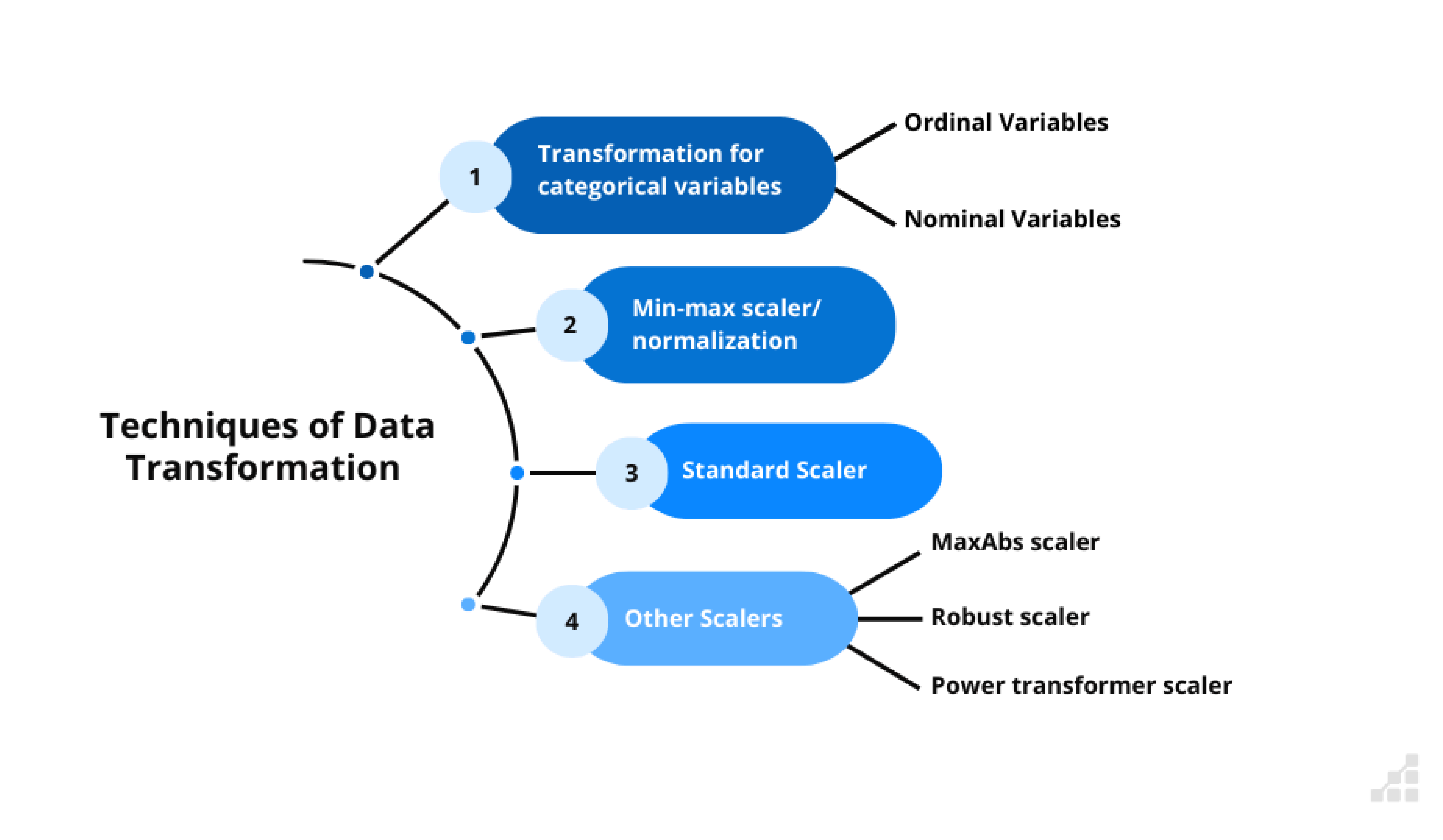
Normalization:

Pixel values scaled to the range [0, 1] for faster and stable training

Data Augmentation:

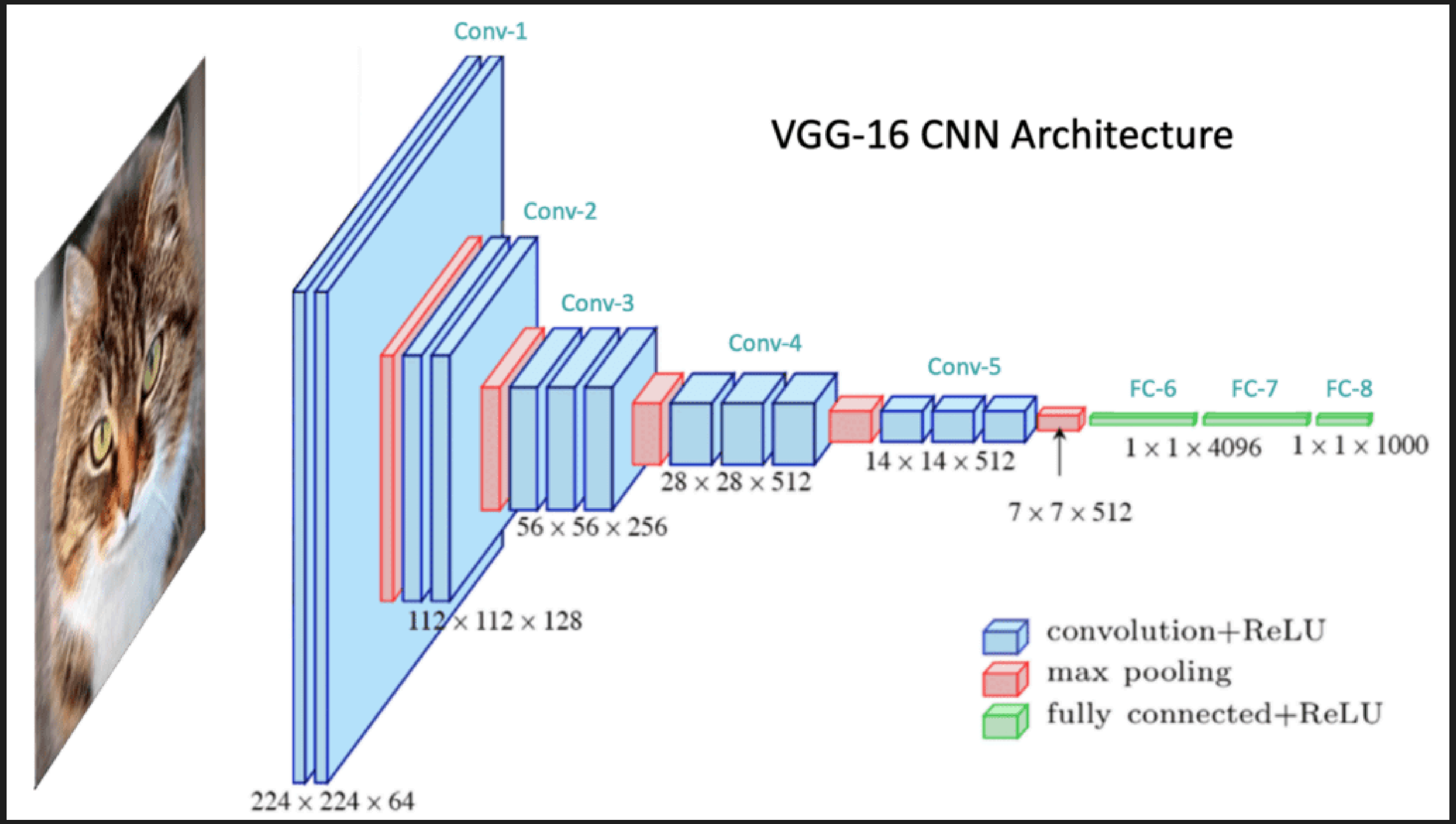
Applied to improve generalization and reduce overfitting:

- Random rotations
- Horizontal flipping
- Zoom and shift operations



Reason for Preprocessing:

- Handle low-resolution images
- Improve robustness to pose and lighting variations



MANUAL CNN ARCHITECTURE DESIGN

Designed manually with minimal help of AI to improve accuracy and generalization on FER2013

Input:

48 × 48 grayscale facial images

Feature Extraction Layers:

- Convolution layers with increasing filters (64 → 128 → 256 → 512)
- Batch Normalization for stable training
- LeakyReLU activation for better gradient flow

Advanced Components:

- Residual Connections to prevent vanishing gradients
- Squeeze-and-Excitation (SE) blocks to emphasize important facial features

Classification Head:

- Global Average Pooling
- Fully connected Dense layer
- Dropout for regularization
- Softmax output layer (7 classes)

MODEL TRAINING STRATEGY

Optimizer: Adam

Loss Function: Categorical Cross-Entropy

Label smoothing applied to reduce overconfidence

Learning Rate Scheduling:

Cosine decay learning rate for smooth convergence

Training Control:

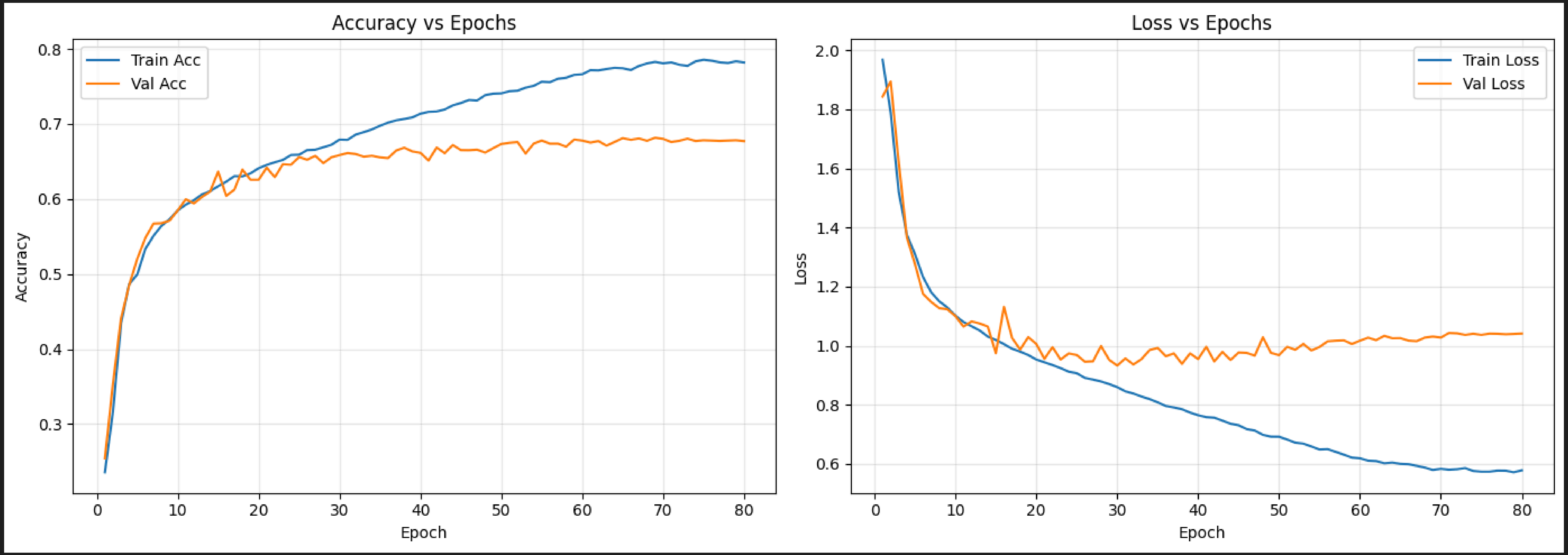
- Early stopping to prevent overfitting
- Model checkpointing to save best weights

Regularization Techniques:

- Dropout layers
- Data augmentation

GROUP 11

EVALUATION METRICS.



Performance drop caused by:
Similar facial expressions
Imbalanced class distribution

Overall Accuracy: 70-71%
Weighted F1-score: 0.68
Macro F1-score: 0.66
→ Indicates class imbalance affects minority classes

Training vs Validation Curves
Training accuracy steadily increases (~78%)
Validation accuracy plateaus (~68%)
Validation loss stops decreasing after ~25 epochs
Indicates mild overfitting but stable generalization

Best performing emotions:
Happy → Precision 0.88, Recall 0.89
Neutral → F1-score 0.78

Challenging emotions:
Fear → Low recall (0.42)
Surprise → Confused with Fear & Happy

GROUP 11

CONFUSION MATRIX AND FINAL DISCUSSION

Key Observations from Confusion Matrix

- Strong diagonal values for Happy and Neutral
- Frequent confusion between:

Fear ↔ Surprise

Sad ↔ Neutral

- Minority classes like Disgust show lower recall due to fewer samples

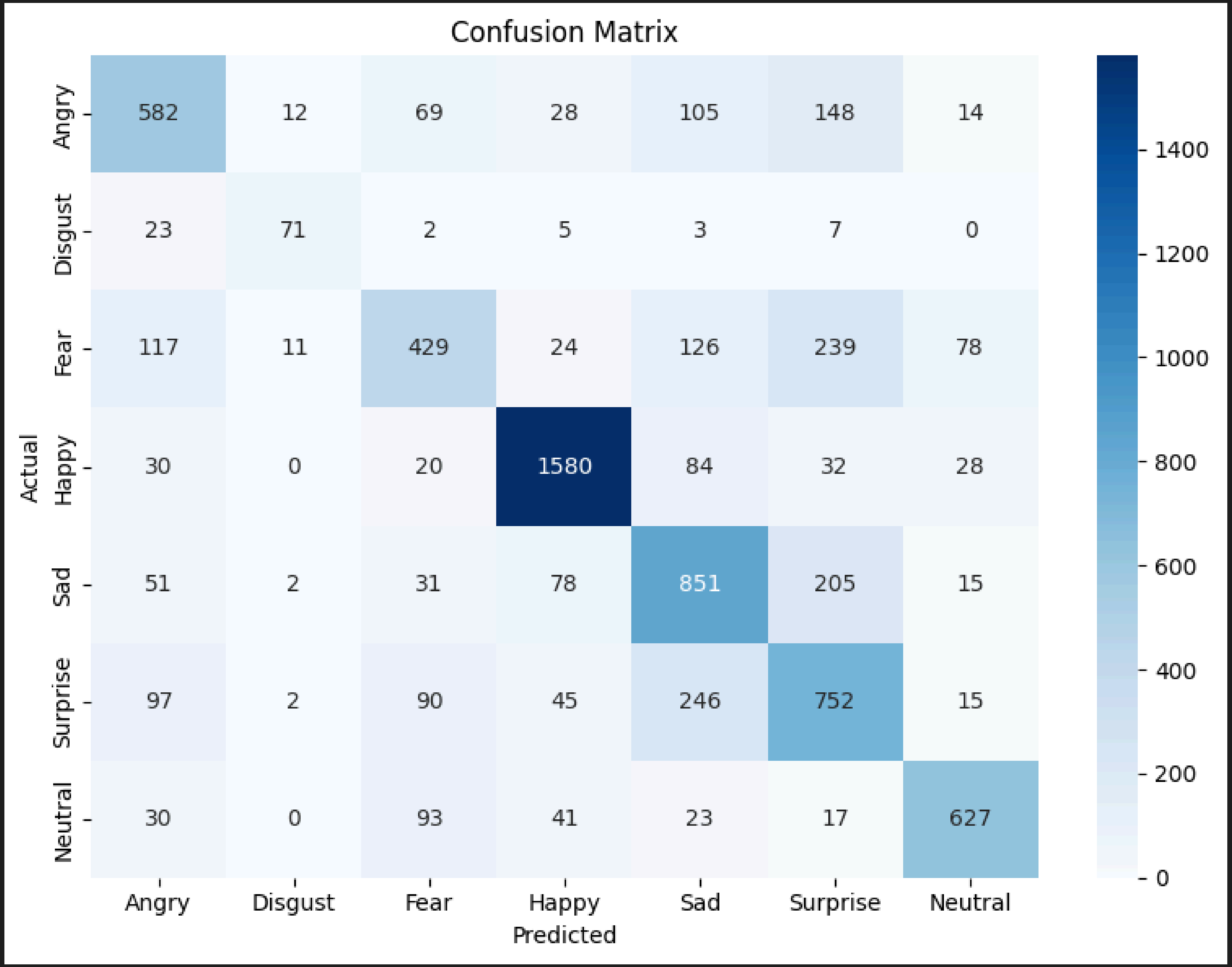
Final Discussion

- Model performs well on distinct facial expressions
- Performance drops for subtle or visually similar emotions
- Confusion is mainly caused by:

Low-resolution (48×48) images

Class imbalance in FER2013

Overlapping facial features



GROUP 11

AI-ASSISTED CNN IMPLEMENTATION

AI Tool Used: ChatGPT 5.2

AI was used to:
- Generate a baseline CNN architecture
- Assist in data loading and preprocessing
- Suggest early stopping and basic regularization

Nature of AI-Generated Model:
- Shallow CNN architecture
- Fewer convolution layers
- No residual connections or attention mechanisms

Outcome
- Faster implementation
- Easier debugging and understanding
- Lower final accuracy compared to manual model

MANUAL VS AI-ASSISTED CNN COMPARISON

Aspect	Manual CNN	AI-Assisted CNN
Architecture	Deep CNN with residual & SE blocks	Shallow CNN
Design Control	Full manual control	Limited
Training Strategy	Advanced (LR scheduling, label smoothing)	Basic
Validation Accuracy	~70–71%	~60%
Generalization	Better	Moderate
Development Time	Higher	Lower

Key Takeaway

- AI is useful for fast prototyping
- Manual design leads to better performance and understanding

Manual vs AI Experience:

- Manual implementation required deeper understanding of CNN design
- AI-assisted approach was faster and more guided

Trust in AI-Generated Code:

- AI-generated code cannot be fully trusted
- Requires verification, testing, and validation

Ethical Considerations:

- AI usage was clearly disclosed
- Prompts and outputs were documented
- Academic integrity was maintained

AI is a supporting tool, not a replacement for learning

REFLECTION AND ETHICAL AWARENESS

CONCLUSION

- Successfully implemented an end-to-end CNN-based emotion recognition system
- Manual CNN achieved better performance than AI-assisted baseline
- Model generalizes reasonably well despite FER2013 limitations
- Proper preprocessing and architecture design significantly improved results

Future Work

- Use deeper architectures (EfficientNet)
- Apply transfer learning for better feature extraction
- Address class imbalance using:

Class weighting

Oversampling

Experiment with higher-resolution datasets

GROUP 11

THANK YOU

PRESENTED TO YOU BY: -

PRATYUSH PATTANAIK 23ETAI410035

MANOJ K BHANDARE 23ETAI410023

NARAHARI N 23ETAI410028

MSRUAS