ECEN 5743 Deep Learning

Final Project

Convolutional Neural Networks for Emotion Classification

Dr. Martin Hagan

Claudia Pauyac

Hridi Prova Debnath

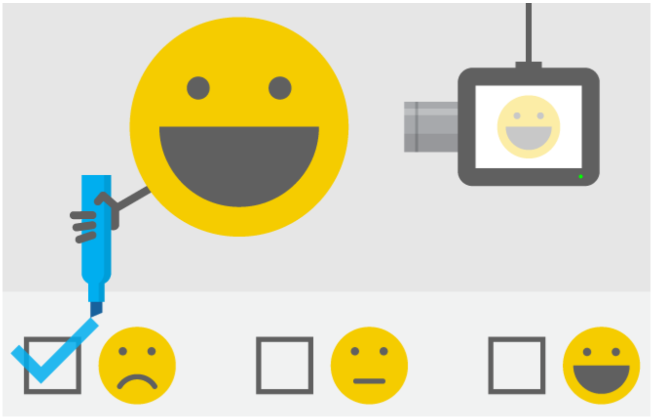
*** Outline**

I. Introduction

* Problem statement

Why emotion recognition?

- Critical for applications like human-computer interaction (HCI), mental health monitoring, and customer experience analysis.
- Challenges: Variability in facial expressions, lightning, pose, and dataset limitations (e.g., dataset's class imbalance, low-resolution images).



* Objective

Goal: Develop a robust model for emotion recognition by:

1. Leveraging pre-trained model for feature extraction
2. Integrating transformer layers to capture spatial attention
3. Enhancing generalization with data augmentation
4. Adding custom CNN/dense layers to refine predictions

Target outcome: Improve accuracy over baseline models and address dataset challenges.

II. Dataset Overview

* Dataset Source:

FER2013: Public benchmark dataset for facial emotion recognition, introduced in ICML, 2013

* Key Statistics:

- Total images: 35 887 grayscale faces (48x48 pixels)
- Emotion classes: 7 (**angry**, **disgust**, **fear**, **happy**, **sad**, **surprise**, **neutral**)
- Resolution: 48x48 pixels (low resolution)
- Split:
  Training: 22 967 images
  Validation: 5 742 images
  Test: 7 178 images

* Preprocessing:

- Resizing for desired pre trained model compatibility
- If required, Grayscale → RGB conversion via channel duplication

* Sample Images:

Angry (4953 images)

Disgust (547 images)

Fear (5121 images)

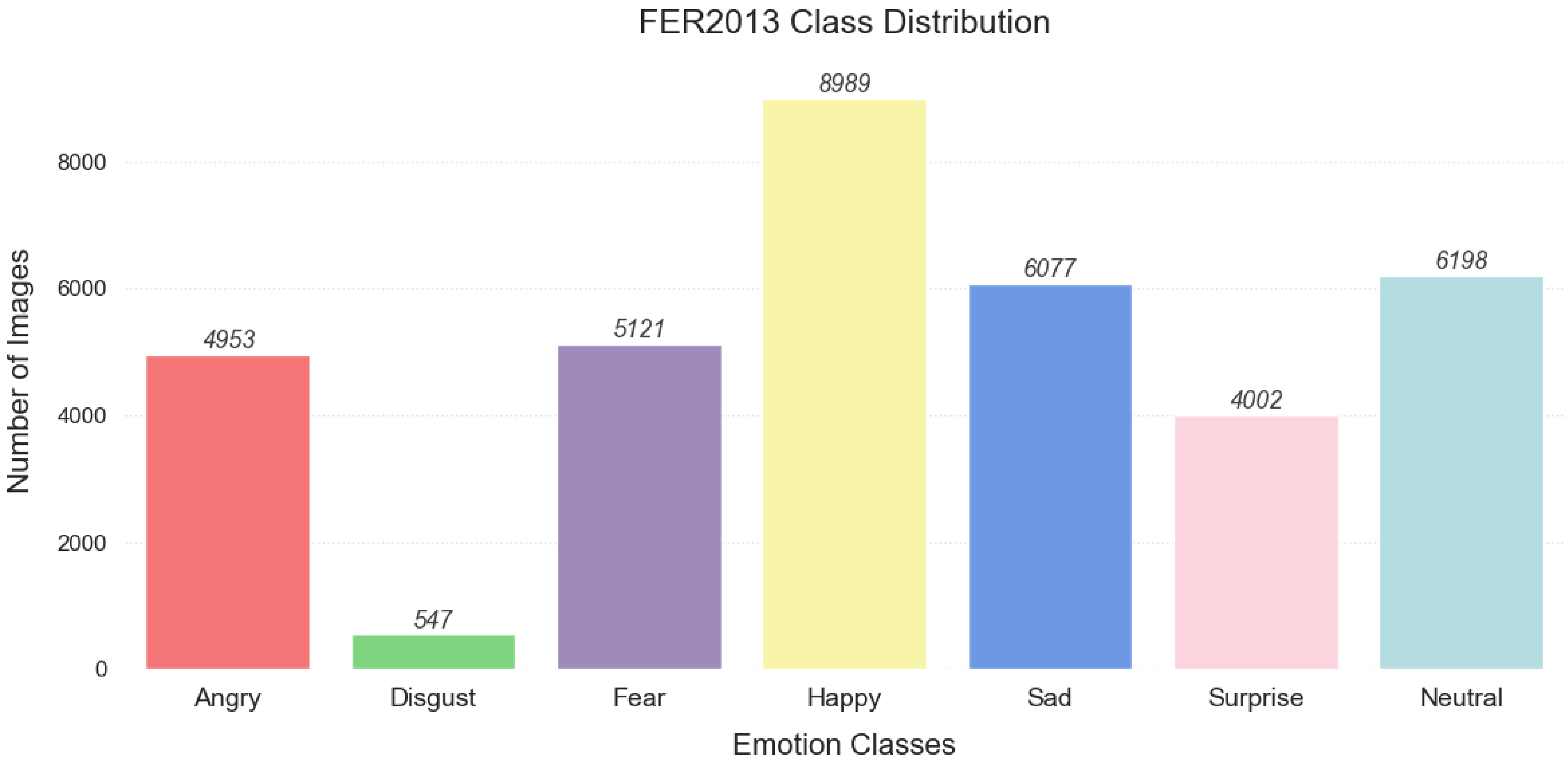
Happy (8989 images)

Sad (6077 images)

Surprise (4002 images)

Neutral (6198 images)

* **Challenges:**
  - Class imbalance (Disgust <5% - only 547 images)
  - Low resolution (e.g., upscaled to 224x224 for ResNet50 compatibility)
  - Generalization (Variability in poses, lighting, and occlusions)



III. Methodology

3.1. Data Preparation & Augmentation

Augmentation techniques:

- **Rotation** ($\pm 35°$)

  Purpose: Simulates head tilts and camera angle variations

  Why 35°: Large enough to capture natural head movements without distorting facial landmarks

- **Weight and height shift range** ($\mathbf{0.25}$)

  Purpose: Account for imperfect face alignment in the dataset

  Why 25%: Matches typical face detection errors (e.g., off-center cropping) and Prevents excessive shifts that would crop out critical facial regions

- **Brightness range** ($[\mathbf{0.5}, \mathbf{1.5}]$)

  Purpose: Simulates varying lighting conditions (dim to bright environments)

  Why $0.5 - 1.5$: Covers natural lighting extremes without over-saturating grayscale pixels

- **Shear range** ($\mathbf{0.4}$)

  Purpose: Mimics perspective changes (e.g., head leaning forward/backward)

  Why 0.4 (~23°): Represents moderate head tilts without warping key emotion features

- **Zoom range** ($\mathbf{0.4}$)

  Purpose: Handles varying face sizes (e.g., distance from the camera)

  Why 40%: Avoids over-zooming (faces become pixelated) or under-zooming (irrelevant background)

- **Horizontal flip** ($\boldsymbol{True}$)

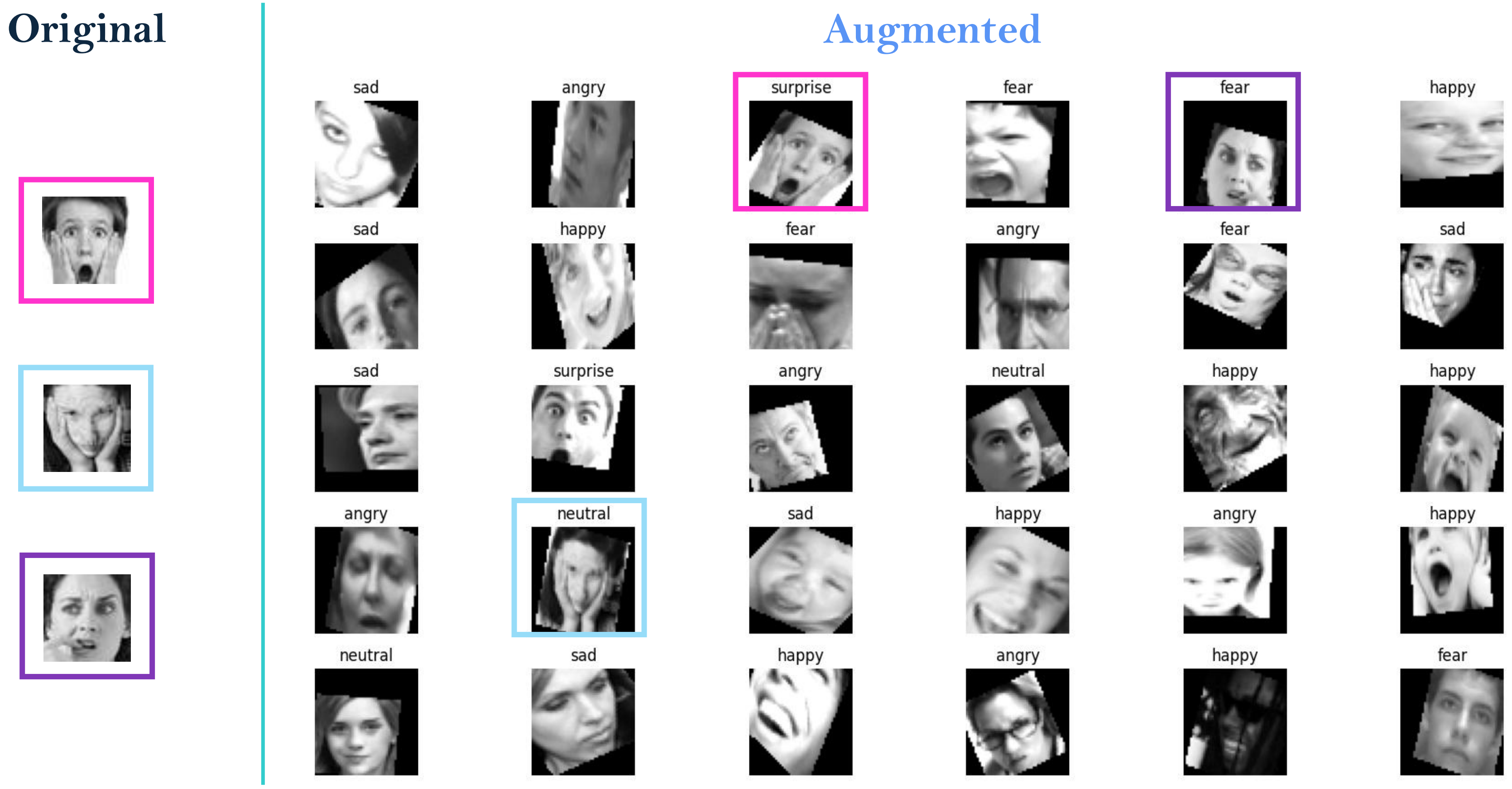
  Purpose: Increases dataset diversity using facial symmetry

  Why: Most emotions are symmetric (e.g., happiness, surprise)

- **Fill mode (*constant*)**

  Purpose: Handles empty pixels after transformations (e.g., rotation/shifts)

  Why *constant*: Simulates occlusions or out-of-frame faces realistically and avoids artificial patterns



Class weighting:

- **Problem:**
  - In FER2013, emotion "disgust" has ~ 400 samples, while "happy" has ~ 7000
  - Without weighting, the model prioritizes majority classes, leading to poor performance on rare emotions
- **Solution:**
  - Assign higher weights to underrepresented classes during training
  - Forces the model to "pay attention" to minority classes by amplifying their impact on the loss function (the loss for each sample is multiplied by its class weight)

Example:

If "disgust" has 400 samples and total samples = $28\,000$:

Weight = $\frac{28\,000}{7\times 400} = 10$: This means each "disgust" sample counts as 10 samples in the loss function

Class	Samples	Weight
"angry"	4953	0.81
"disgust"	547	7.31
"fear"	5121	0.78
"happy"	8989	0.44
"sad"	6077	0.66
"surprise"	4002	0.99
"neutral"	6198	0.65

3.2. Hybrid Architecture Design

ResNet50 Backbone:

- Pretrained on ImageNet to leverage strong low-level feature extractors
- Initial layers frozen during early training phases to retain general image features
- Later layers gradually unfrozen during fine-tuning to adapt to facial emotion nuances in grayscale FER2013 images

Key Architectural Additions:

1. Spatial-Channel Attention (SCA)

- Channel Attention: Learns which features (e.g., textures, edges) are most important per class
- Spatial Attention: Learns where to look on the face
- Suppresses irrelevant background noise and emphasizes subtle facial cues (focus on discriminative facial regions like eyes, mouth, and eyebrows)

2. Efficient Transformer Block

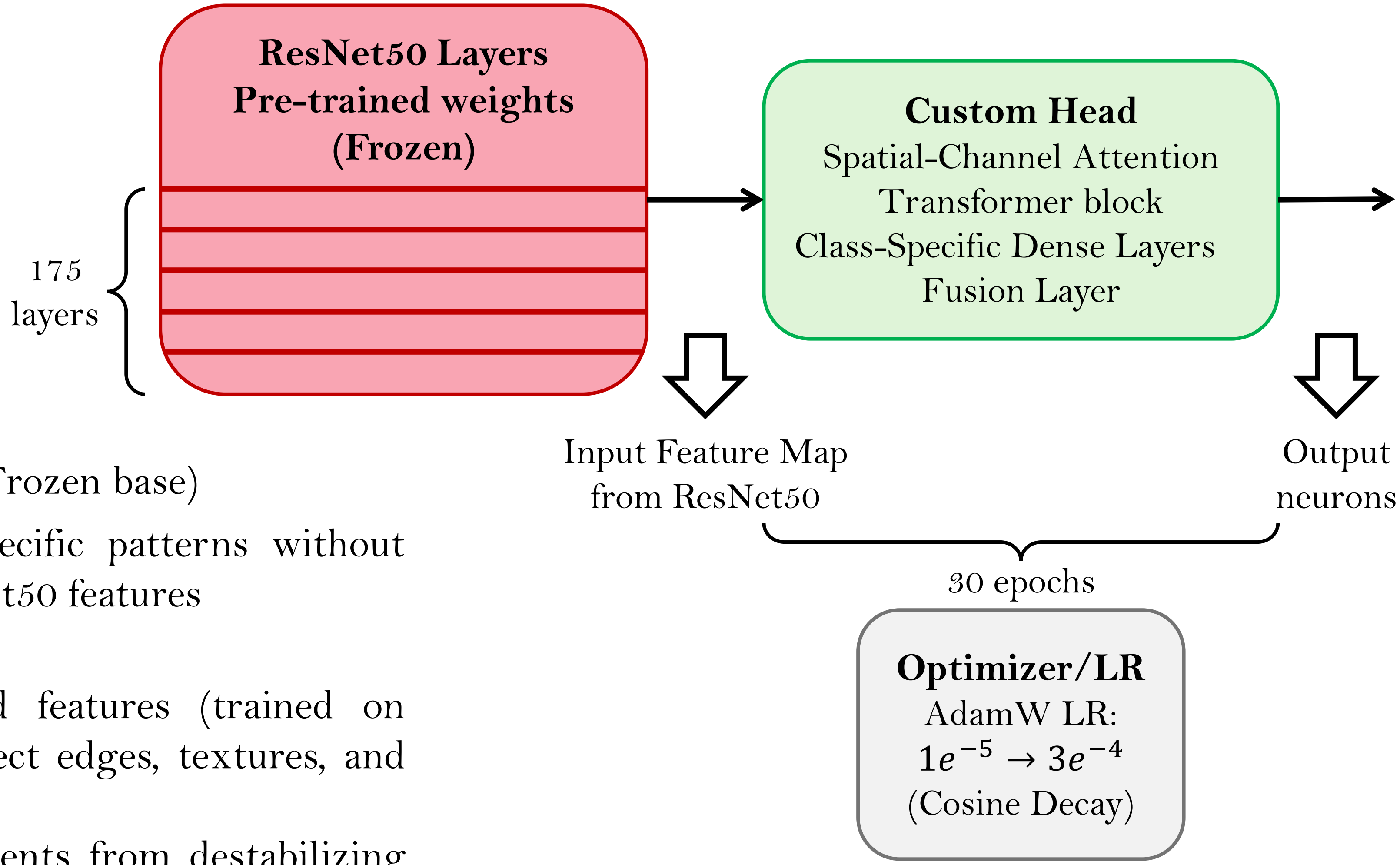
- Uses multi-head self-attention to detect relationships between distant facial landmarks
- Enhances understanding of global emotion context, especially useful in complex expressions like fear or surprise

3. Class-Specific Branches

- Combat class imbalance and improve performance on underrepresented or confused classes (e.g., angry, disgust, fear, sad)
- Each branch processes ResNet + Transformer output separately
- Encourages the model to learn unique features for challenging classes, boosting their recall and F1 scores

3.3. Progressive Training Strategy

FEATURE EXTRACTION (FROZEN BASE)



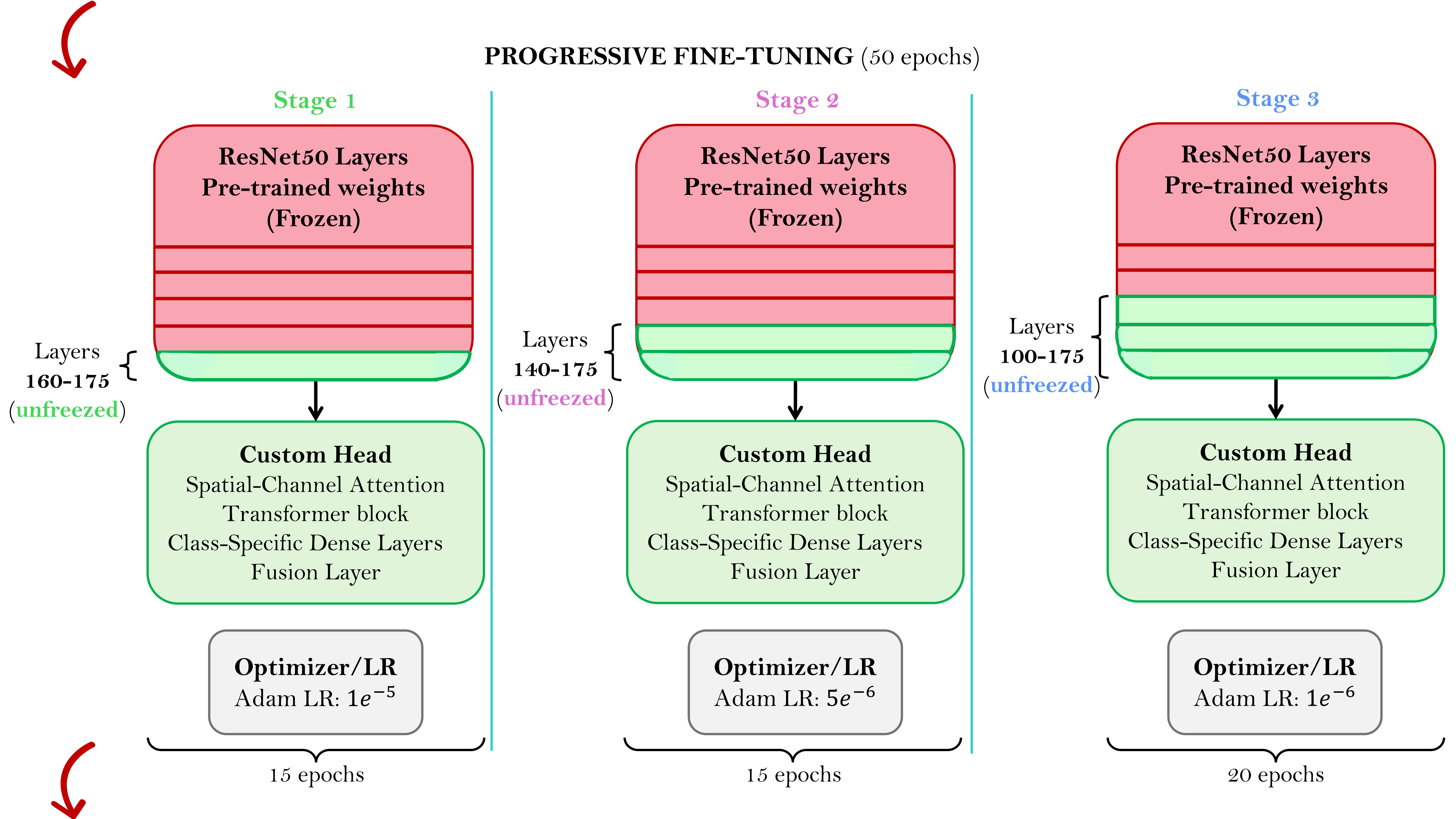
<u>Phase 1</u>: Feature extraction (Frozen base)

- **Objective:** Learn task-specific patterns without altering pre-trained ResNet50 features
- **Why?**
  - ResNet50's pre-trained features (trained on ImageNet) already detect edges, textures, and basic shapes
  - Prevents "noisy" gradients from destabilizing the base early in training

Phase 2: Fine-tuning (Gradual layer unfreezing)

- **Objective:** Gradually adapt ResNet50 to emotion-specific features
- **Implementation:**
  - Layers unfreezing: Unfreeze ResNet50 layers in stages (deep → shallow)
    1. **Stage 1**: Layers 160-175 (final ResNet50 blocks, closest to head)
    2. **Stage 2**: Layers 140-160 (mid-level blocks)
    3. **Stage 3**: Layers 100-140 (earlier blocks)

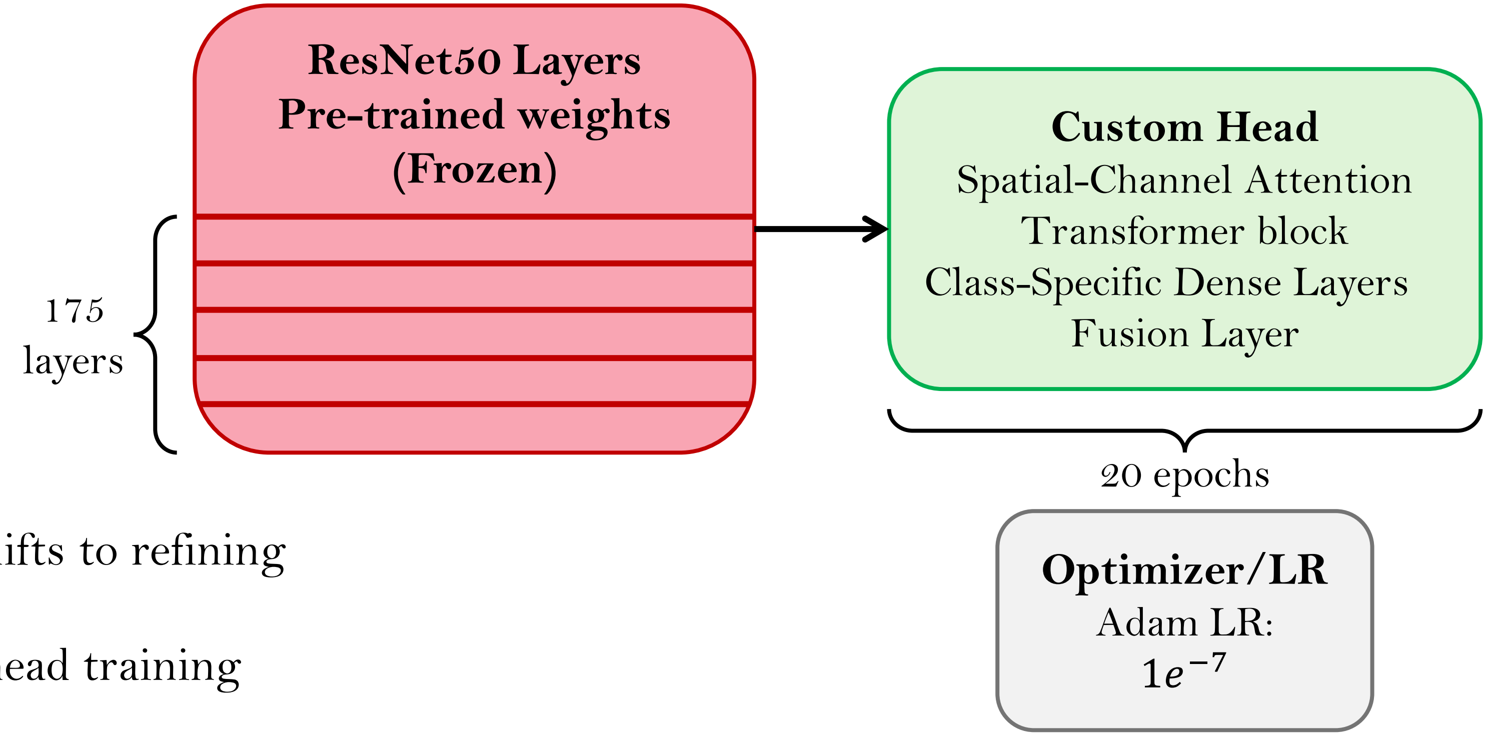
  - Why deep → shallow?
    1. Deeper layers encode high-level features (facial structures) critical for emotions
    2. Earlier layers detect generic patterns (edges) that need minimal adjustment


PROGRESSIVE FINE-TUNING (50 epochs)
Stage 1
ResNet50 Layers
Pre-trained weights
(Frozen)
Layers
160-175
(unfreezed)
Custom Head
Spatial-Channel Attention
Transformer block
Class-Specific Dense Layers
Fusion Layer
Optimizer/LR
Adam LR: $1e^{-5}$
15 epochs
Stage 2
ResNet50 Layers
Pre-trained weights
(Frozen)
Layers
140-175
(unfreezed)
Custom Head
Spatial-Channel Attention
Transformer block
Class-Specific Dense Layers
Fusion Layer
Optimizer/LR
Adam LR: $5e^{-6}$
15 epochs
Stage 3
ResNet50 Layers
Pre-trained weights
(Frozen)
Layers
100-175
(unfreezed)
Custom Head
Spatial-Channel Attention
Transformer block
Class-Specific Dense Layers
Fusion Layer
Optimizer/LR
Adam LR: $1e^{-6}$
20 epochs

<u>Phase 3</u>: Head optimization (Dense layers)

- **Objective:** Final polish of the custom architecture
- **Implementation:**
  - <u>Freeze ResNet50</u>: Base model weights locked again
  - Unfreeze head: Attention layers, dense layers, and class-specific branches
- **Why?**
  - After adapting ResNet50, focus shifts to refining decision boundaries
  - Prevents overfitting by isolating head training

HEAD TUNING



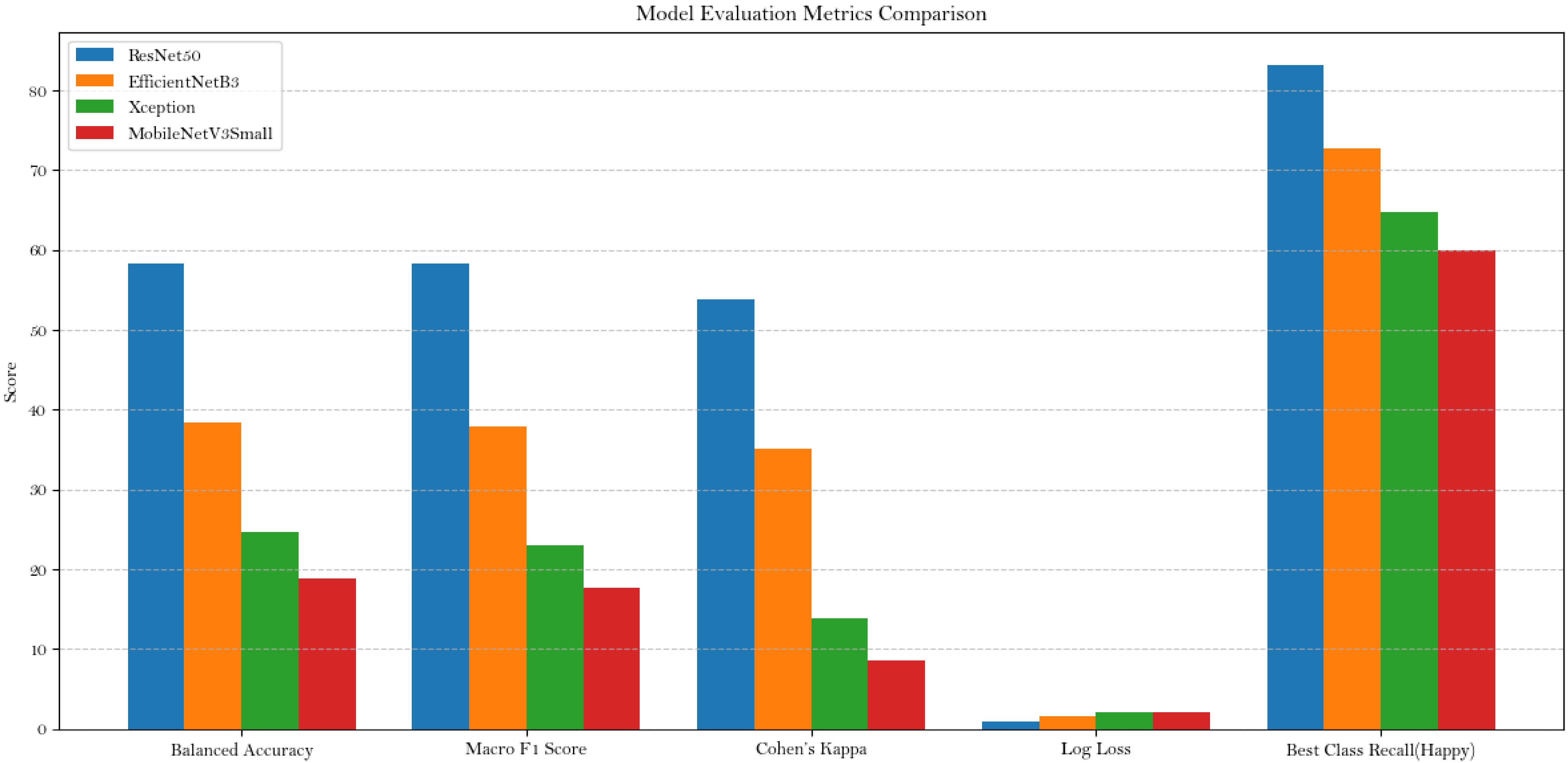
<u>Optimizer</u>: AdamW with cosine learning rate decay

- **Adam Optimizer:** Ensures robust parameter updates with effective regularization (preserving ResNet50's pre-trained knowledge)
- **Cosine Decay:** Enables smooth transitions between training phases

3.4. Comparative Analysis of 4 Models

Model	Strengths	Weaknesses	Highlights
ResNet50	• Best balanced accuracy (58.30%), Top macro F1 (58.29%) • Excellent per-class recall (e.g., Happy 83.2%, Surprise 77.5%)	• High computational cost (77.4M params) • Slower inference	High-accuracy applications in production and scientific research
EfficientNetB3	• Good recall for Sad (73%) and Surprise (65%) • Moderate Cohen's Kappa (42.81%)	• Fails on Fear (12%) and Disgust (28%) • Performance unstable without tuning	Resource-aware deployments that can afford tuning and moderate performance
Xception	• Moderate recall for Happy (64.83%) • Lightweight with 74.6M parameters	• Low balanced accuracy (24.6%) • Very poor on Surprise (2.49%) and Sad (5.68%)	Only with class balancing or additional training data
MobileNetV3Small	• Compact model • Decent recall on Happy (59.92%) • Very low parameter count (~2.5M)	• Extremely low recall on Disgust (6.31%) and Fear (2.05%) • Lowest macro F1 (17.69%)	Edge devices, mobile apps, rapid prototyping where speed > accuracy

IV. Results & Analysis



Why ResNet50?

1. Best Overall Performance

ResNet50 consistently outperformed all other tested models across all key evaluation metrics:

- **Highest balanced accuracy** (58.87%)—indicating better handling of class imbalance.
- **Strong macro F1 score** (57.37%)—demonstrating reliable predictions across all emotion classes.
- **Lowest log loss** (1.056)—showing well-calibrated confidence in predictions.
- **Highest per-class recall** (83.09% for *Happy*)—capturing even subtle expressions better than others.

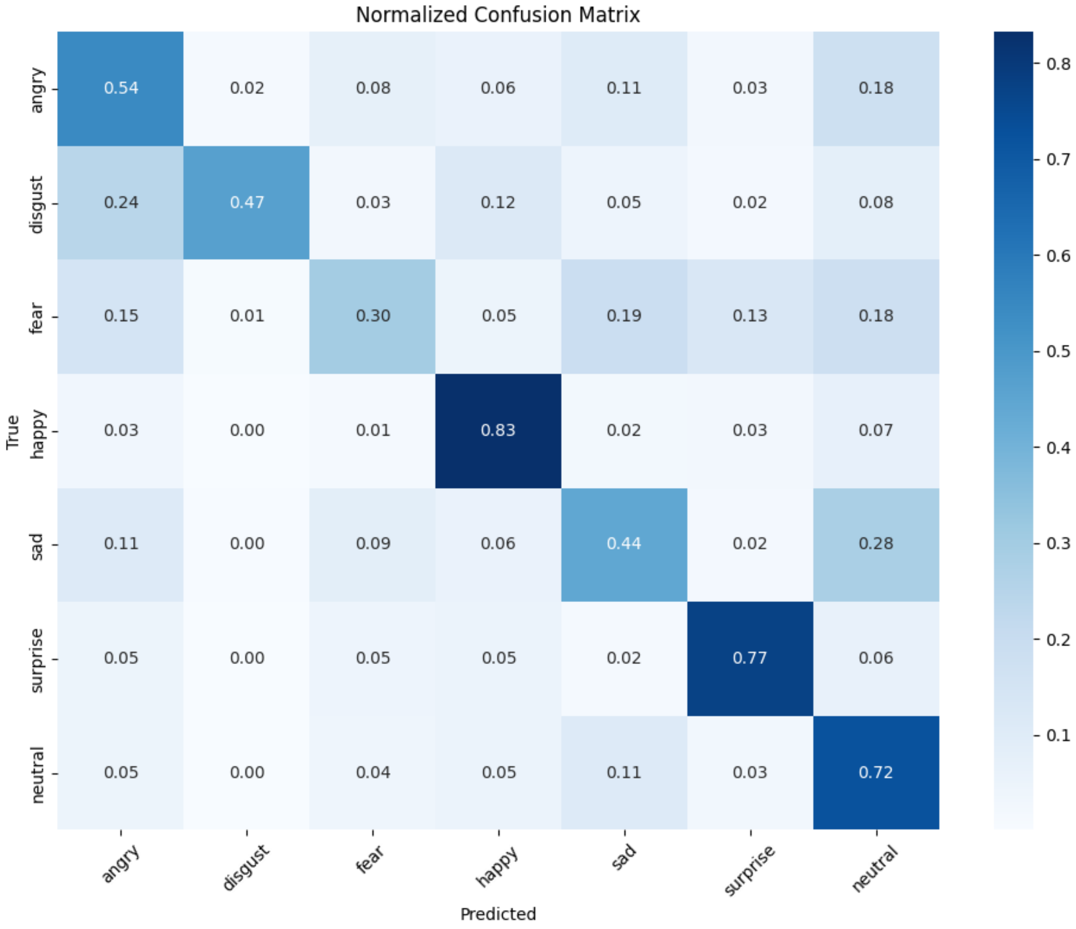
2. Powerful Yet Proven Architecture

ResNet50 uses residual connections to allow training of deep networks without vanishing gradients. This enables it to:

- Learn complex patterns in facial expressions.
- Generalize better to unseen faces.
- Avoid performance collapse that occurs in very deep or very shallow models.

3. Ideal for Transfer Learning

- Pretrained on ImageNet, ResNet50 effectively transfers low-level features to FER2013, making it perfect for small, real-world emotion datasets.



4. Balance Between Accuracy and Efficiency

While heavier than MobileNet, ResNet50 is much smaller than newer transformer-based models and runs efficiently on standard GPUs, making it suitable for real-time applications like social robotics and assistive technologies.

V. Key Challenges & Solutions

1. Class Imbalance & Confidence Issues

Challenge: The model struggled with underrepresented classes like *Disgust* (111 samples) and *Fear* (1,024), while overpredicting dominant ones like *Happy* (1,774). Predictions were often overconfident and poorly calibrated.

Solution:

- Apply class-aware augmentations and synthetic oversampling (GANs).
- Incorporate heavy class weighting (e.g., Disgust ×15, Fear ×8).
- Use temperature scaling and MC Dropout to improve prediction confidence and uncertainty handling.

2. Input Mismatch & Feature Loss

Challenge: FER2013 grayscale images (48×48) were mismatched with ResNet50's 224×224 RGB input, leading to degraded feature quality with naive channel repetition.

Solution:

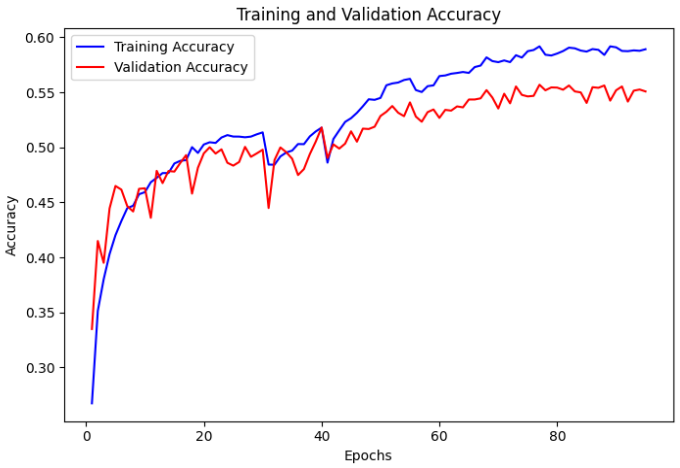
- Introduce a learnable grayscale-to-RGB adapter using CNN layers.
- Employ multi-scale image fusion to preserve both local and global expression cues.

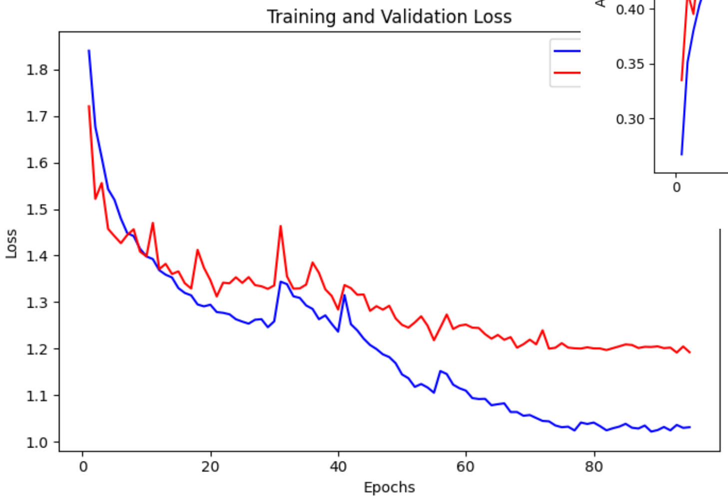
3. Overfitting & Optimization

Challenge: With ~77 million parameters, ResNet50 risked overfitting the small FER2013 dataset.

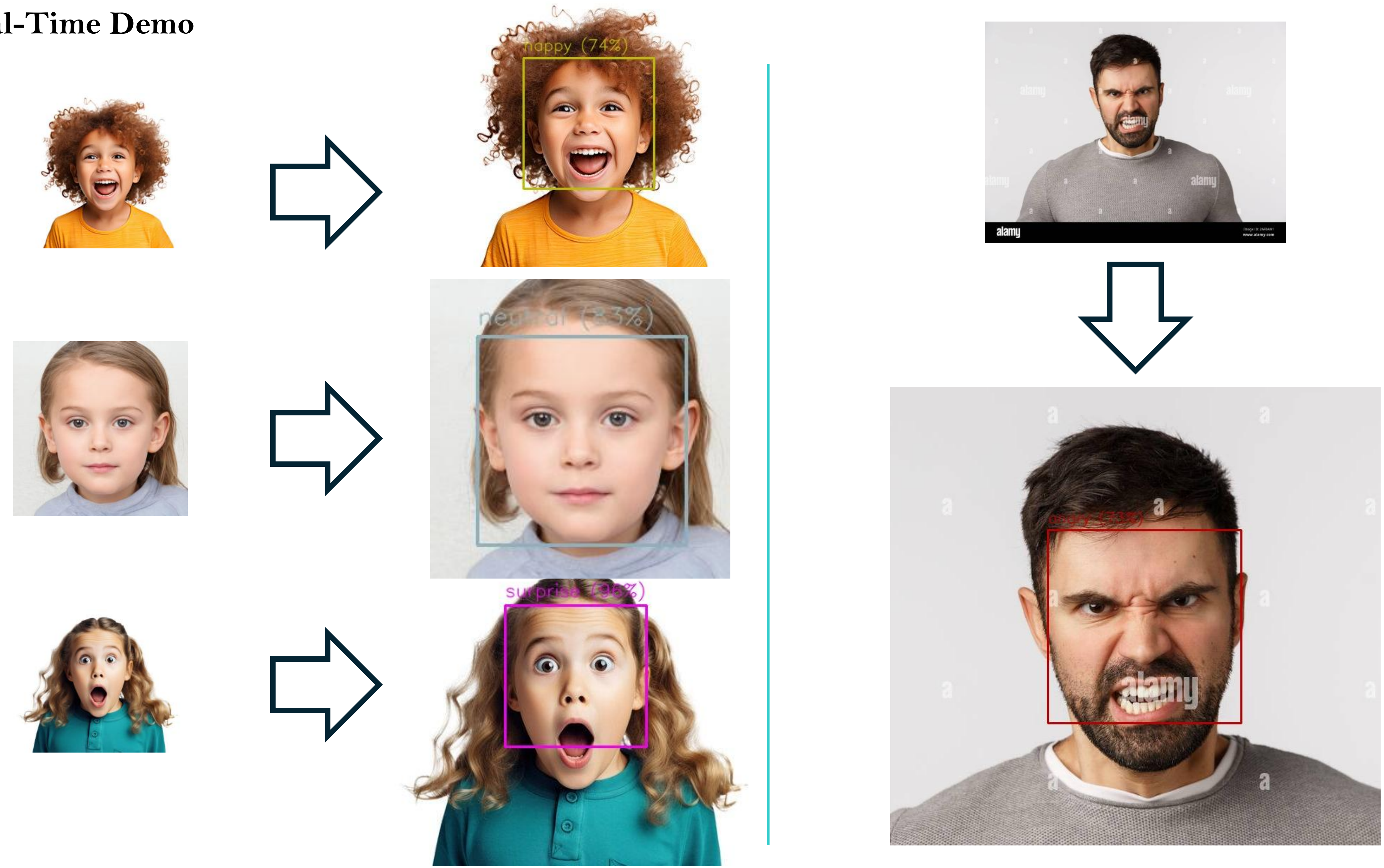
Solution:

- Frozen early layers and fine-tuned only higher layers.
- Add strong regularization (dropout, L2 penalty).
- Use cosine learning rate decay with warmup for stable training dynamics.



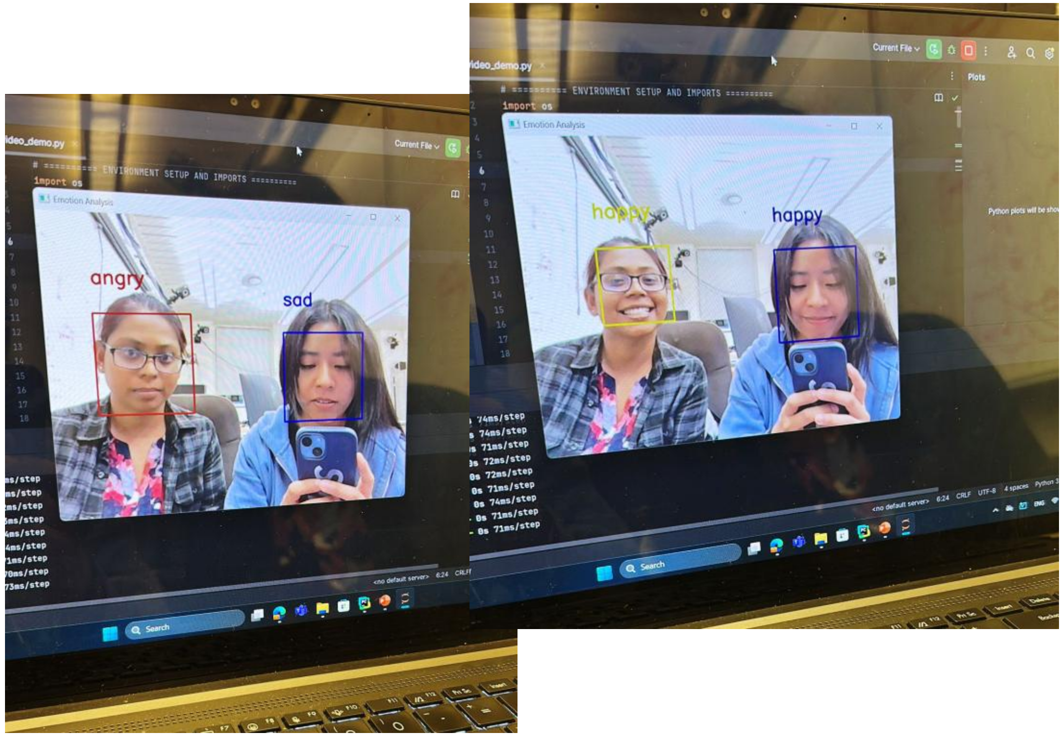


VI. Real-Time Demo



VII. Summary

- Developed a ResNet50-based architecture tailored for FER2013 emotion recognition
- Incorporated attention modules and class-specific output branches
- Tackled key challenges: class imbalance, grayscale input, and small dataset size
- Used a progressive training strategy (freeze →fine-tune → optimize) to reduce overfitting
- Model was robust, scalable, and well-calibrated



Take Home:

- ResNet50 proved to be a high-performing and adaptable backbone
- Ideal for real-time emotion recognition in assistive robotics, mental health monitoring, and affective computing

VIII. Future Work

8.1. Improve Feature Discrimination

Challenge: Misclassification of similar emotions (e.g., "fear" vs. "sad“, “angry” vs. “disgust”, “sad” vs. “angry”)

Innovations:

- **Cross-Domain Attention**: Add 3D attention blocks to capture spatial-temporal features
- **Multi-Modal Fusion**: Combine facial, vocal, and textual cues for holistic emotion analysis

8.2. Expand Dataset Diversity

Current Limitation: Biases in FER2013 (limited demographics/lighting conditions)

Steps:

- Use AffectNet or RAF-DB for richer variation
- Audit model performance across age, gender, and ethnicity subgroups

8.3. Explore Cutting-Edge Architectures

Research Directions:

- **Vision Transformers (ViTs)**
- **Self-Supervised Learning**: Pre-train on unlabeled video data to improve feature learning

8.4. Real-World Applications

Use Cases:

- **Mental Health Monitoring**: Integrate with apps to track emotional well-being
- **Human-Computer Interaction**: Enable emotion-aware chatbots or VR systems

Deployment: Develop a user-friendly API for easy integration

IX. References

- **ResNet50 Architecture**

K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778. DOI: 10.1109/CVPR.2016.90.

- **Data Augmentation**

F. Chollet et al., "Keras: Deep Learning for Humans," GitHub Repository, 2015. [Online]. Available: https://keras.io/api/preprocessing/image/. [Accessed: 10-Oct-2023].

- **Spatial-Channel Attention**

J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation Networks," in IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 7132-7141. DOI: 10.1109/CVPR.2018.00745.

- **Transformer Blocks**

A. Vaswani et al., "Attention Is All You Need," in Advances in Neural Information Processing Systems (NeurIPS), 2017, pp. 5998-6008. arXiv: 1706.03762.

- **Class Weighting**

F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825-2830, 2011. [Online]. Available: https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html.

- **FER2013 Dataset**

I. J. Goodfellow et al., "Challenges in Representation Learning: A Report on Three Machine Learning Contests," Neural Networks, vol. 64, pp. 59-63, 2015. DOI: 10.1016/j.neunet.2014.09.005.

- **Mixed Precision Training**

P. Micikevicius et al., "Mixed Precision Training," in International Conference on Learning Representations (ICLR), 2018. arXiv: 1710.03740.

- **AdamW Optimizer**

I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," in International Conference on Learning Representations (ICLR), 2019. arXiv: 1711.05101.

- **Model Architecture Inspiration**

A. Mollahosseini, B. Hasani, and M. H. Mahoor, "AffectNet: A Database for Facial Expression, Valence, and Arousal Computing in the Wild," IEEE Transactions on Affective Computing, vol. 10, no. 1, pp. 18-31, 2019. DOI: 10.1109/TAFFC.2017.2740923.

- **Cosine Learning Rate Decay**

I. Loshchilov and F. Hutter, "SGDR: Stochastic Gradient Descent with Warm Restarts," in International Conference on Learning Representations (ICLR), 2017. arXiv: 1608.03983.

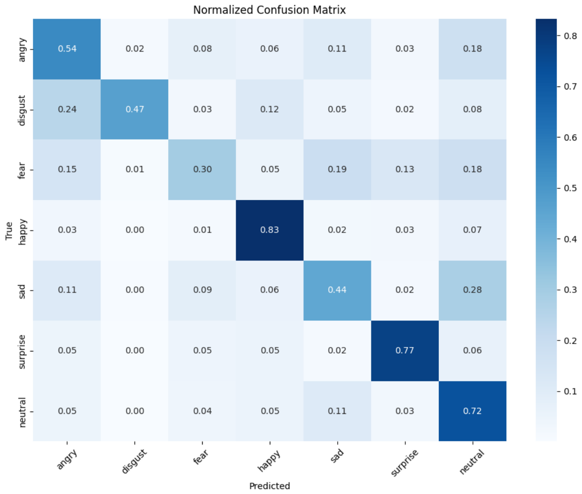
- **Real-Time Deployment**

H. Yang et al., "Efficient Facial Emotion Recognition Using Hierarchical Neural Networks," 2017. arXiv: 1710.07557v1.

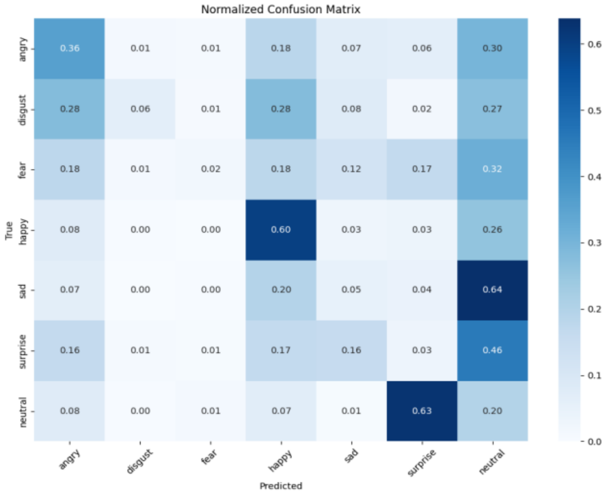
Thank you!

Appendix I

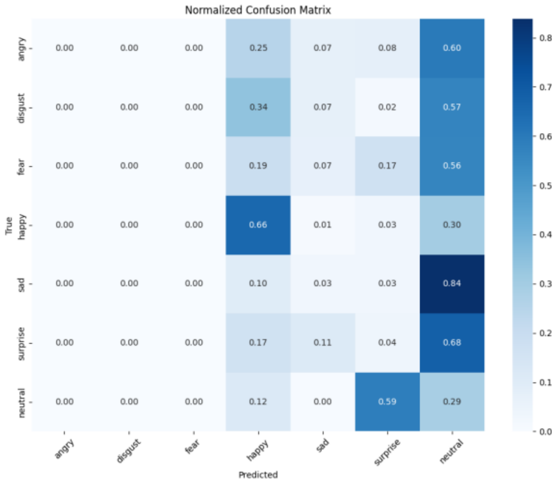
(Confusion Matrix)



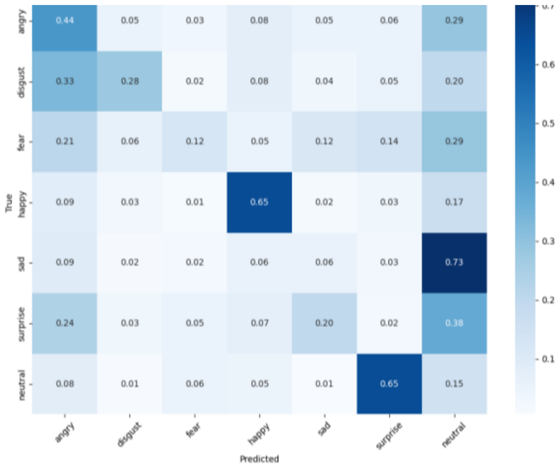
ResNet50



MobileNetV3Small

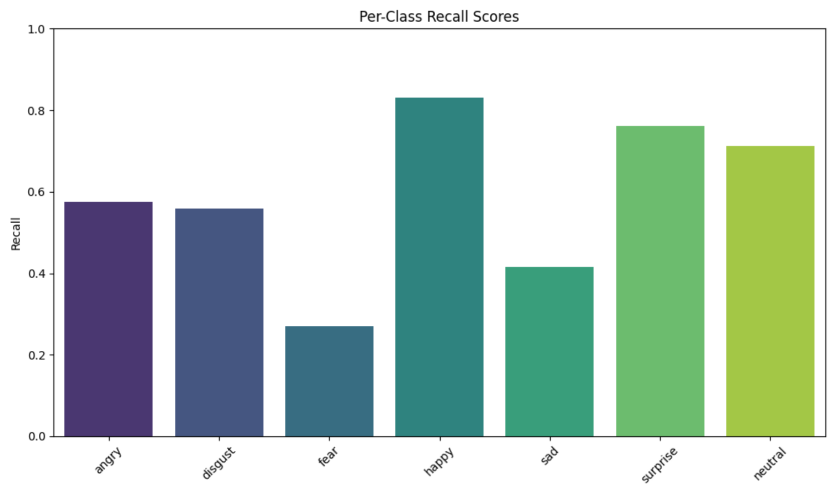


EfficientNetB3

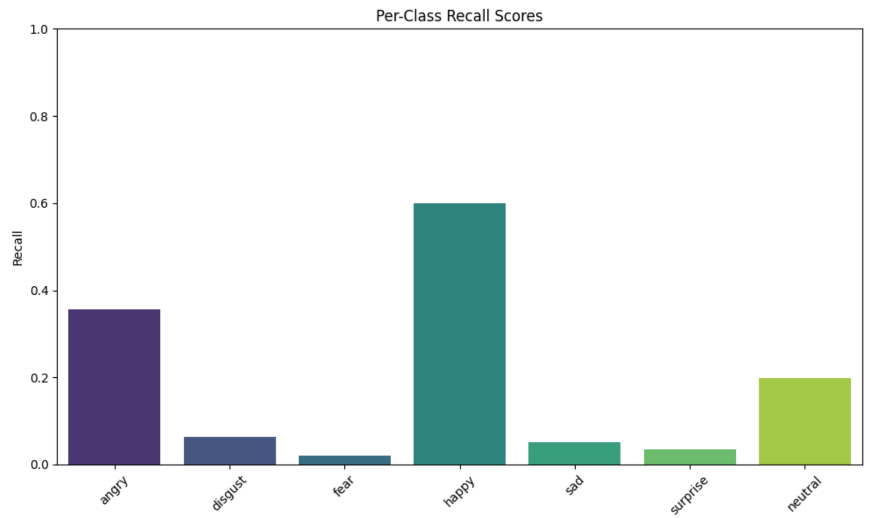


Xception

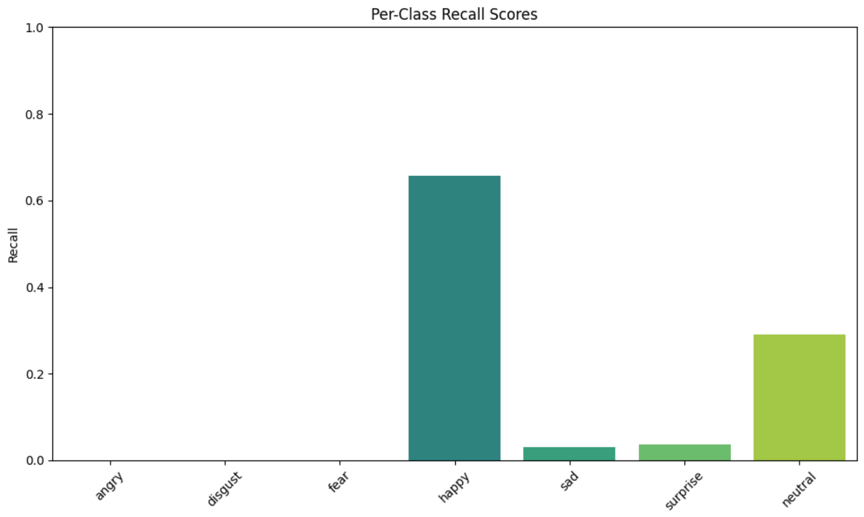
Appendix I
(Recall Distribution)



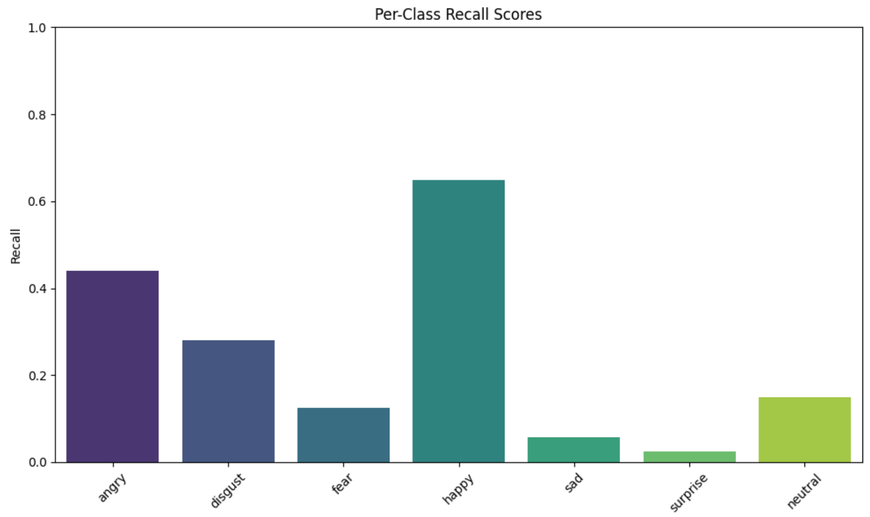
ResNet50



MobileNetV3Small



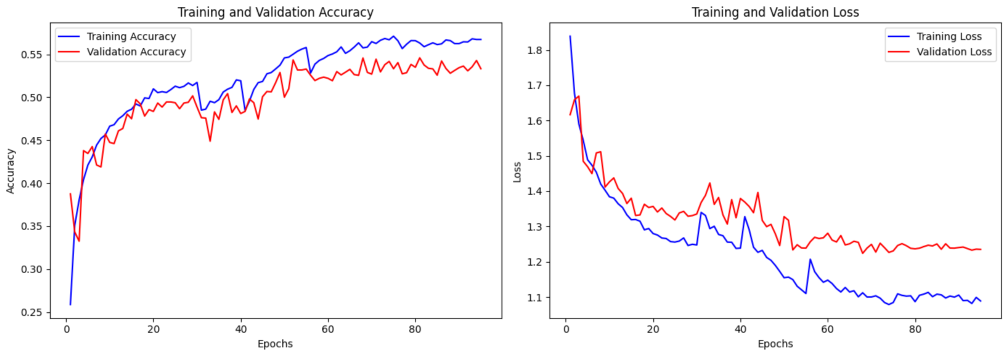
EfficientNetB3



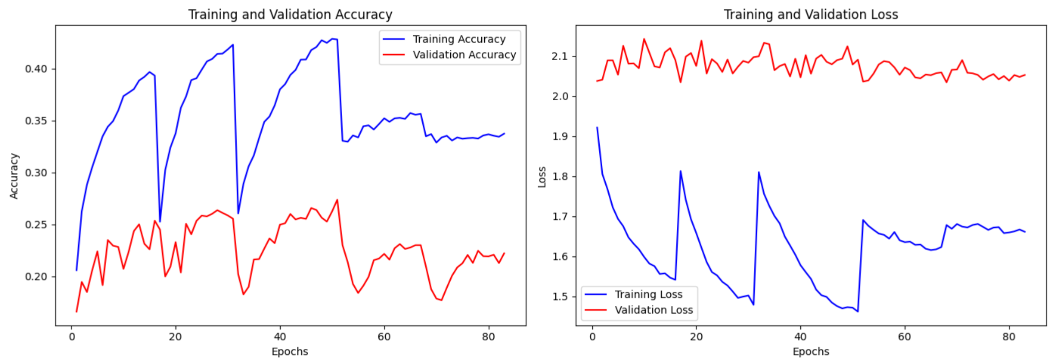
Xception

Appendix I

(Training History)

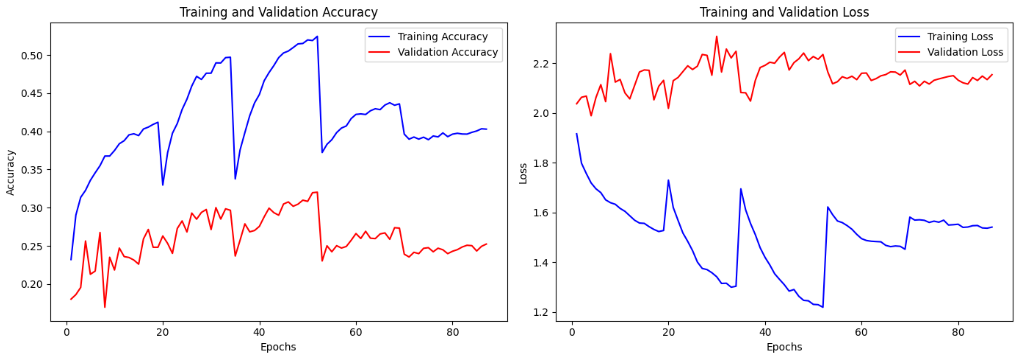


ResNet50



MobileNetV3Small

EfficientNetB3



Xception