

Facial Expression Recognition

Outline

- Introduction: What is our problem and why it's important (applications of our project)
- Dataset preparation
 - Dataset overview
 - Data enriching
 - Data preprocessing (greyscale, resize, normalization)
 - Dataset classes balancing
 - Dataset classes reducing
 - Train test split
- Model training
 - Architecture selection
 - Data augmentations
 - Fine-tuning of several layers for 6 emotions
 - Re-Training the whole model for 5 emotions
- Outcomes
- Future work
- Test-drive

Introduction

Facial expressions - A powerful means of conveying emotions.

Applications:

- 1. Healthcare:** helps in treating autistic patients or post-stroke aphasia, Mental Health Monitoring.
- 2. Customer Service:** analyze the satisfaction or frustration levels of customers during interactions

Dataset Overview

- The dataset consists of 7 emotions: **Anger, Fear, Happy, Neutral, Sad, Disgust, and Surprise.**
- Total images: 2921
- Image size: 244* 244

Dataset

kaggle dataset



+

Our own images

Data Enriching

- Different emotions
- Different angles
- Different light conditions



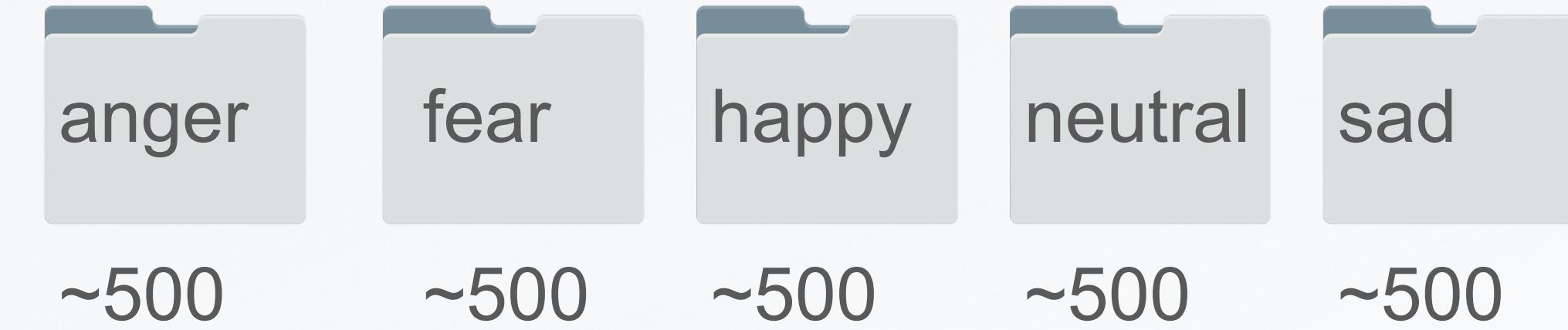
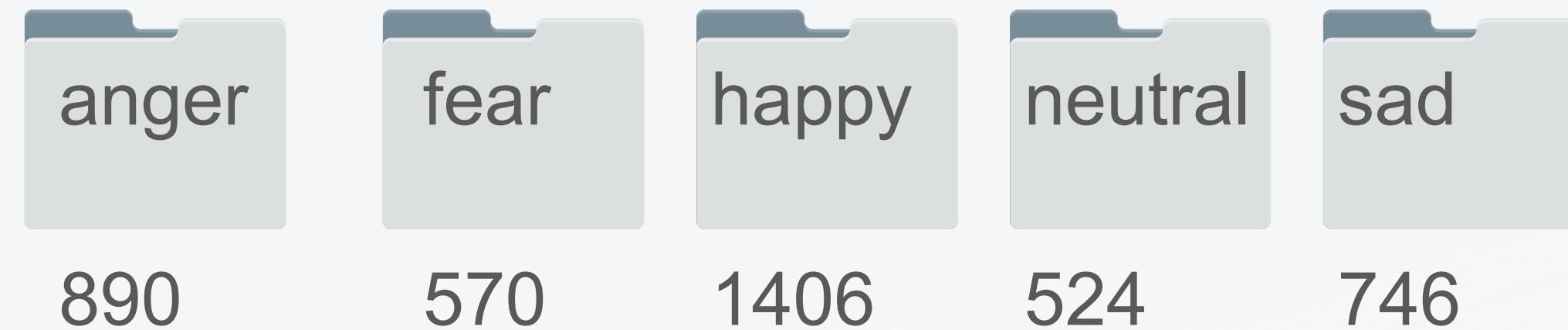
Data Preprocessing

Grayscaleing
(1 channel)

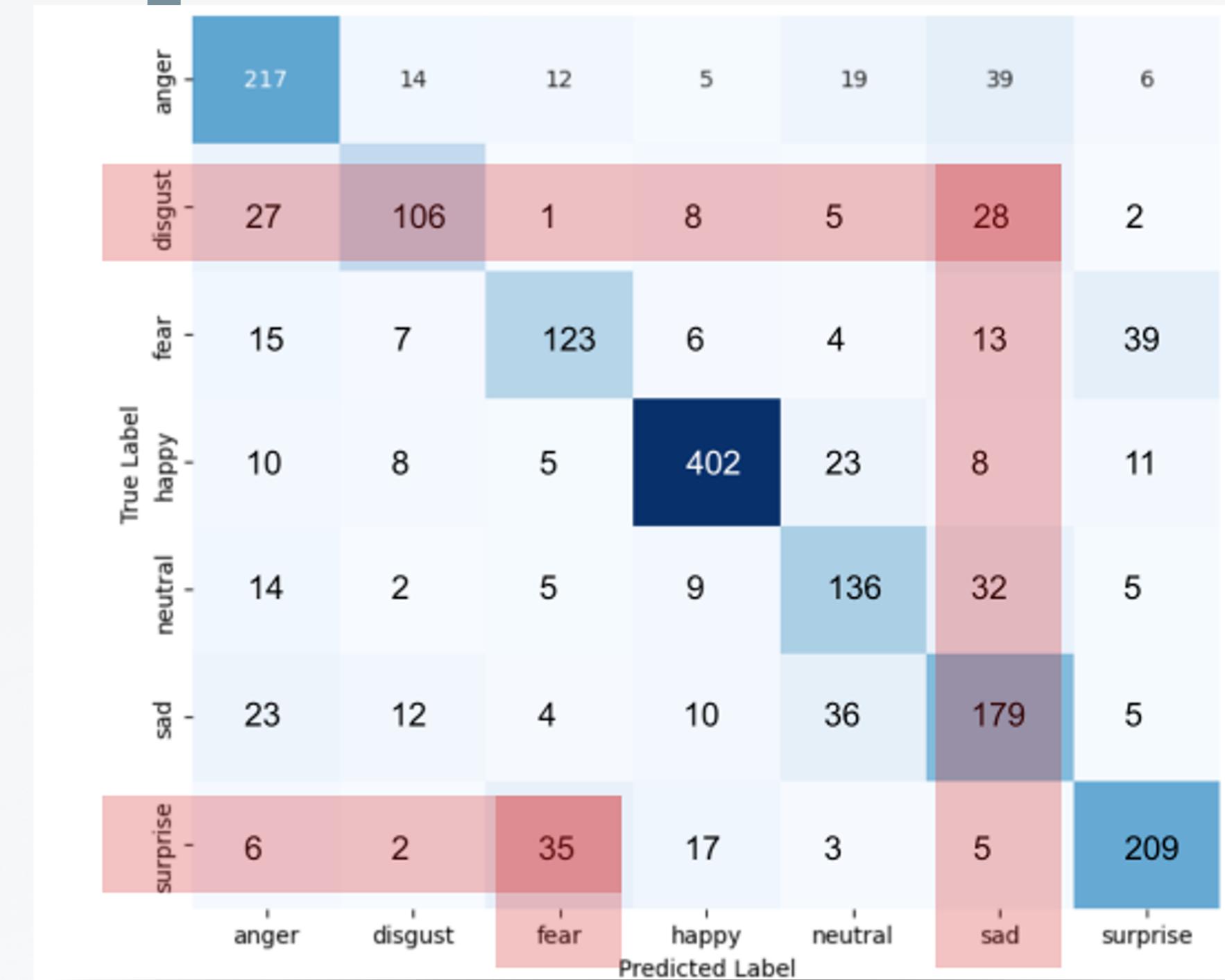
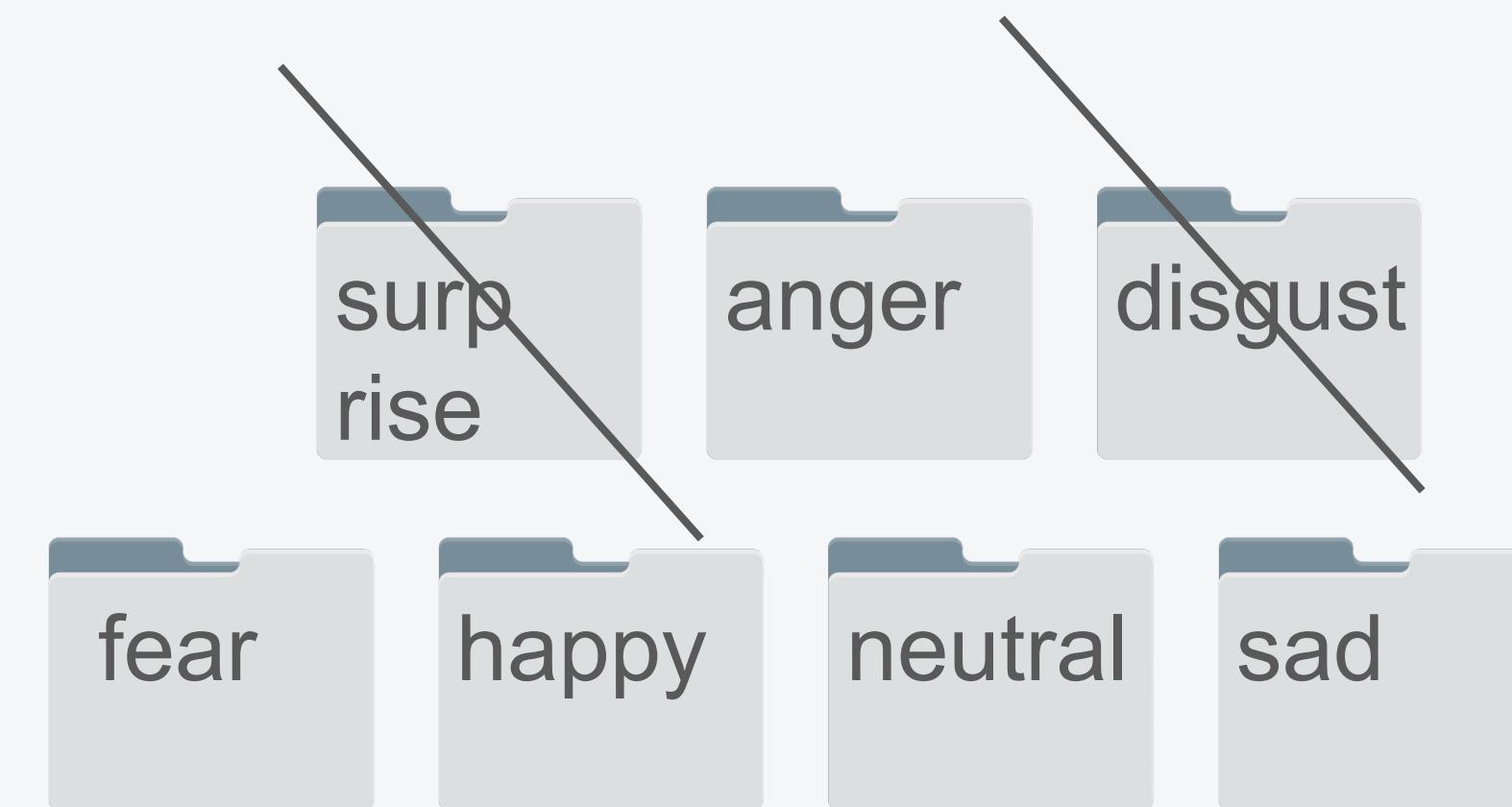
Resizing to 224x224

Normalization

Classes Balancing



Classes Reducing



Train Test Split

80% training

20% validation

Architecture selection

ResNet-50

Trade-off between computational
load and accuracy

Data Augmentation

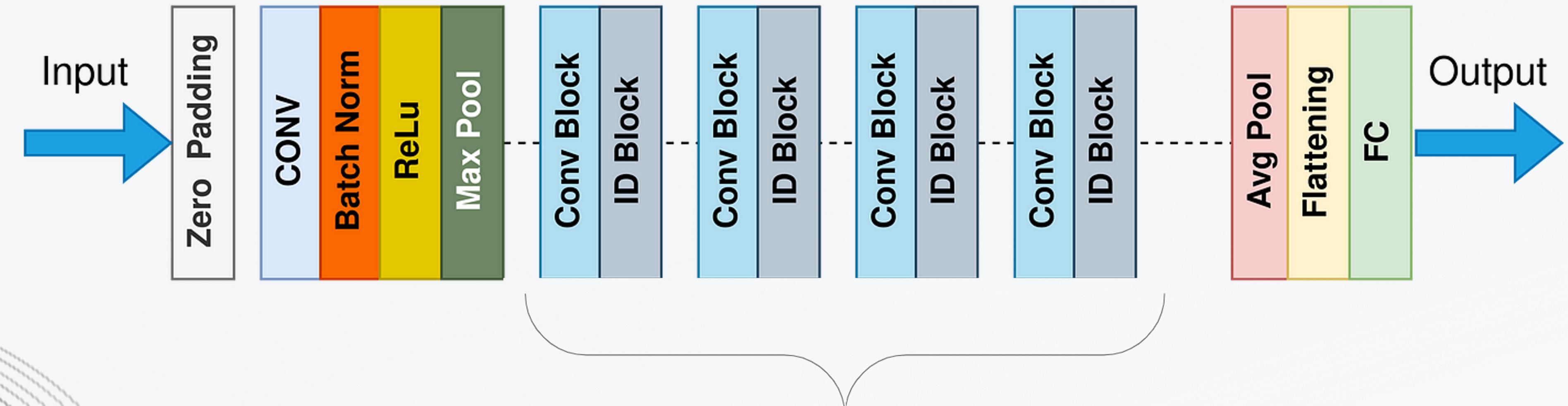
Horizontal Flip

Resized Crop

Color Jitter

Rotation

Fine-tuning of several layers



Bottlenecks for gradual unfreezing

Fine-tuning of several layers

Preliminary:

- Modification of FL classifier for 6 classes (then 5 classes)
- Feature extraction and FL training on them

Layers:

- FC + Bottleneck 4
- FC + Bottleneck 4, 3
- FC + Bottleneck 4, 3, 2
- FC + Bottleneck 4, 3, 2, 1

Hyperparameters:

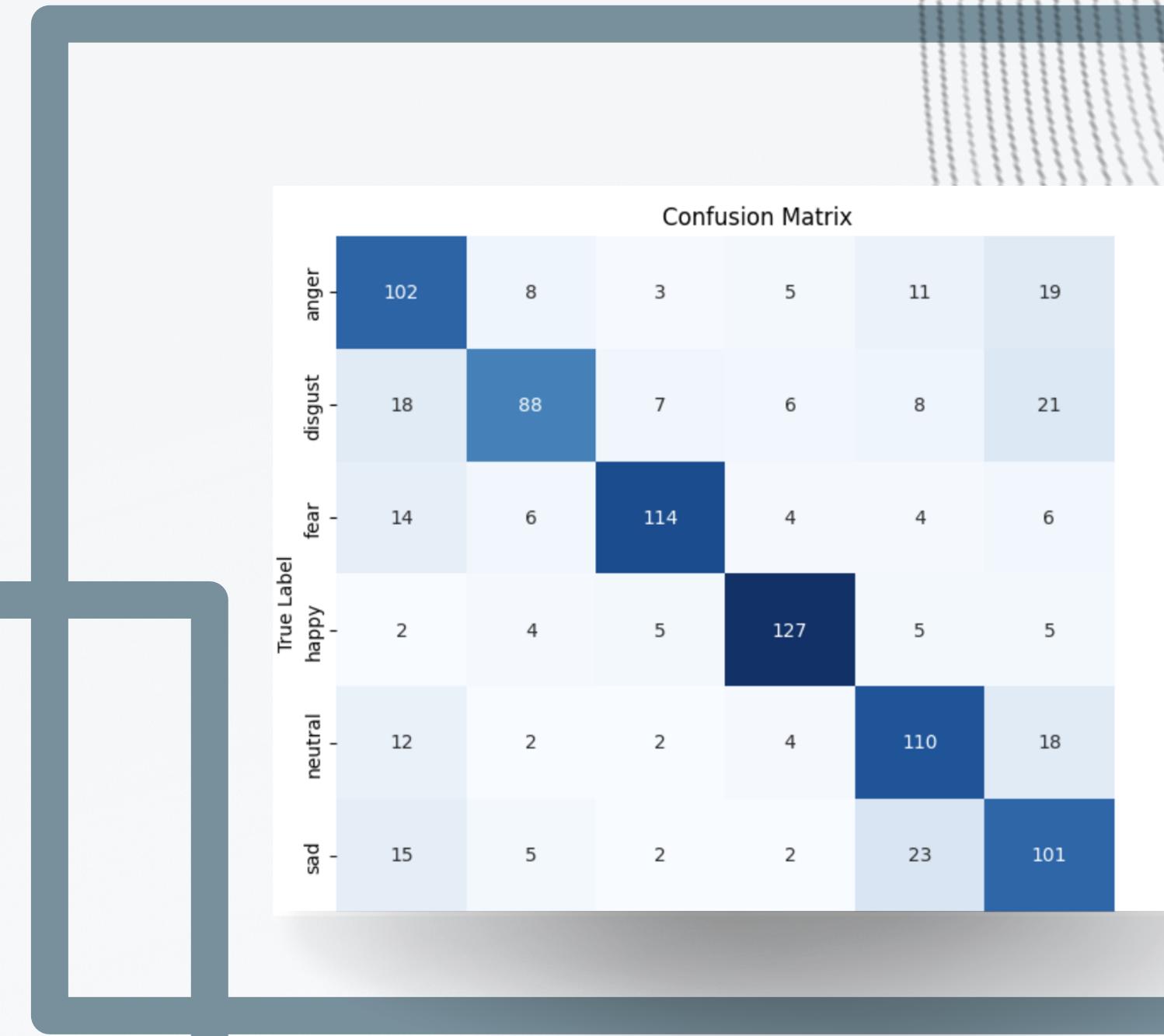
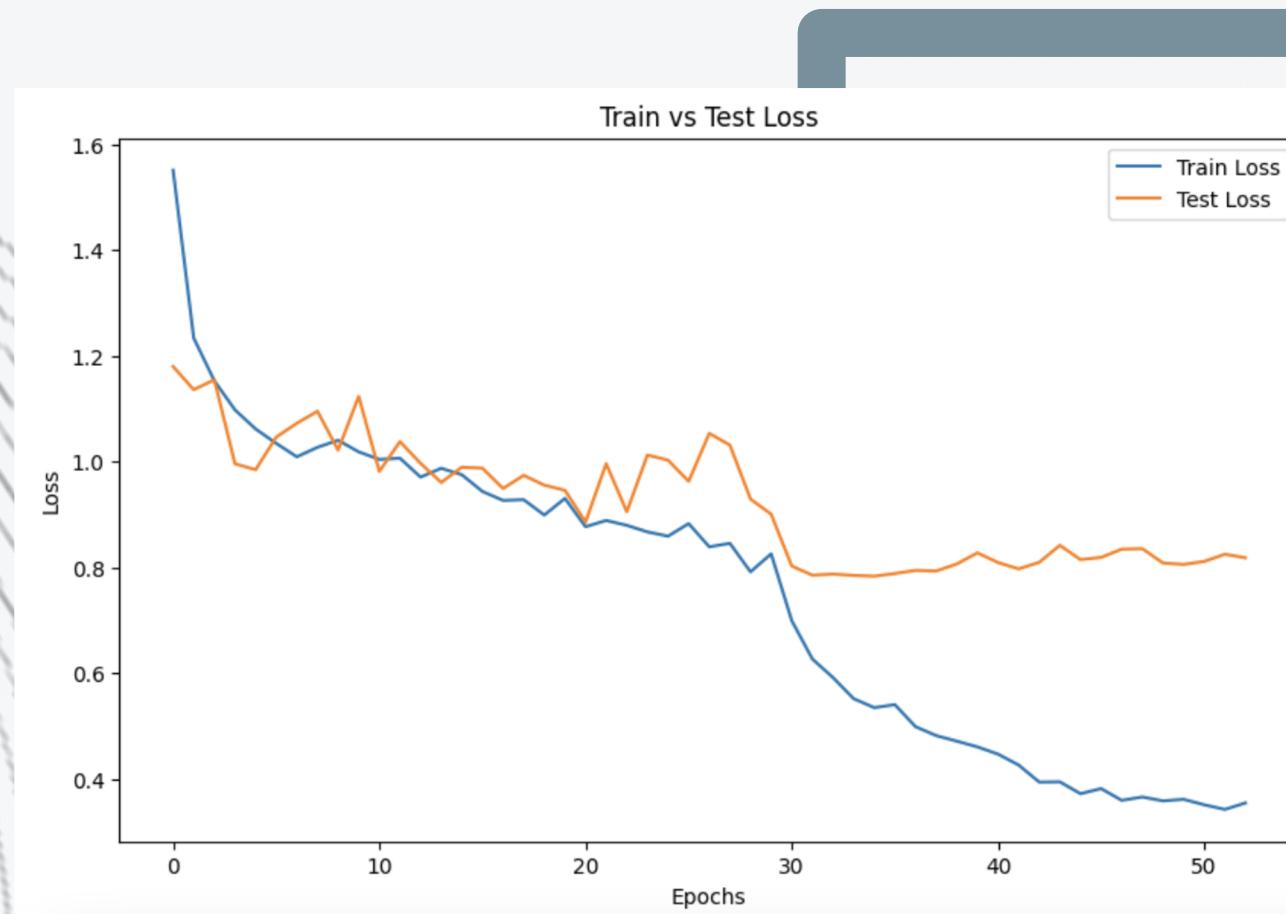
- Loss Function = CrossEntropyLoss
- LR = $1e-4$ for FC classifier, $1e-3$ for layers
- batch size = 8 for training, 64 for testing
- # of epochs = 100
- Early stopping patience 18
- Scheduler type = ReduceLROnPlateau
- Scheduler patience = 8
- (L2 regularization) weight_decay = $1e-1$ for FC classifier, $1e-2$ for layers

Fine-tuning of several layers, performance

FC classifier + block 4, 3

6 classes: Accuracy = 75.9%

5 classes: Accuracy = 82.7%



Real Time implementation

- Utilized the Opencv library for detecting facial expressions.
- Processed the frames with the same test transformations
- Passed the preprocessed face image to model
- Drew boxes around faces with predicted emotion on top



Future Scope and Conclusion

Future scope:

- Enhancing the Model Accuracy
- Inclusion of more emotions into the model
- Adapt to Diverse Demographics
- Integration in New Domains

Conclusion:

- Successfully developed with validation accuracy of 83.93% on the dataset
- Implemented Realtime emotion recognition
- Found that unfreezing of 2 bottlenecks brings pretty same accuracy as a whole network re-training

Test Drive

Let's try together!