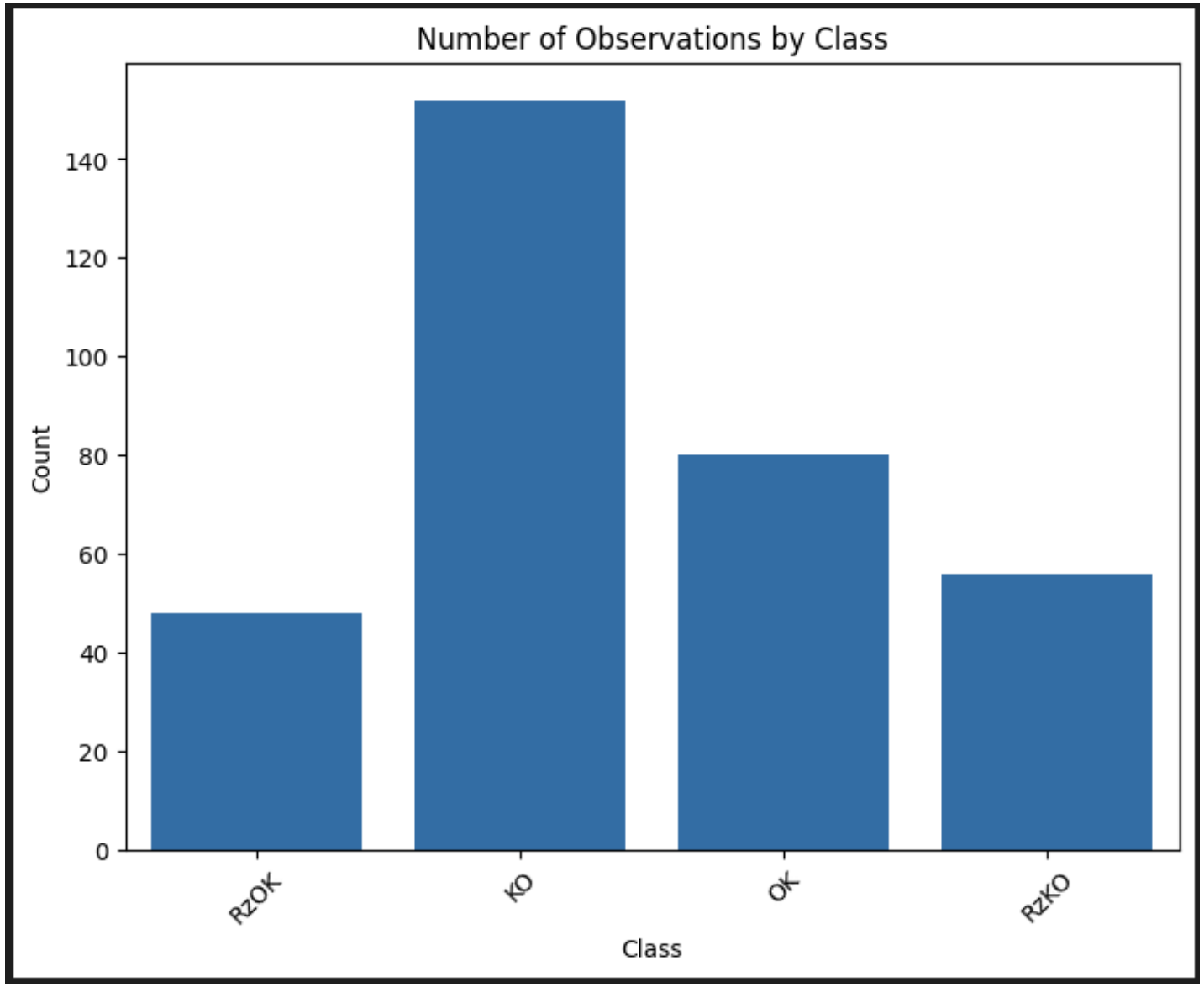
IMPLEMENTATION

November 2024

TRAIN DATASET

- 4 classes
  - KO: 140 samples
  - OK: 80 samples
  - RzKO: 55 samples
  - RzOK: 50 samples

DATASET


Number of Observations by Class
Count
0
20
40
60
80
100
120
140
RzOK
KO
OK
RzKO
Class

TEST DATASET

- 4 classes
  - KO: 16 samples
  - OK: 9 samples
  - RzKO: 14 samples
  - RzOK: 16 samples

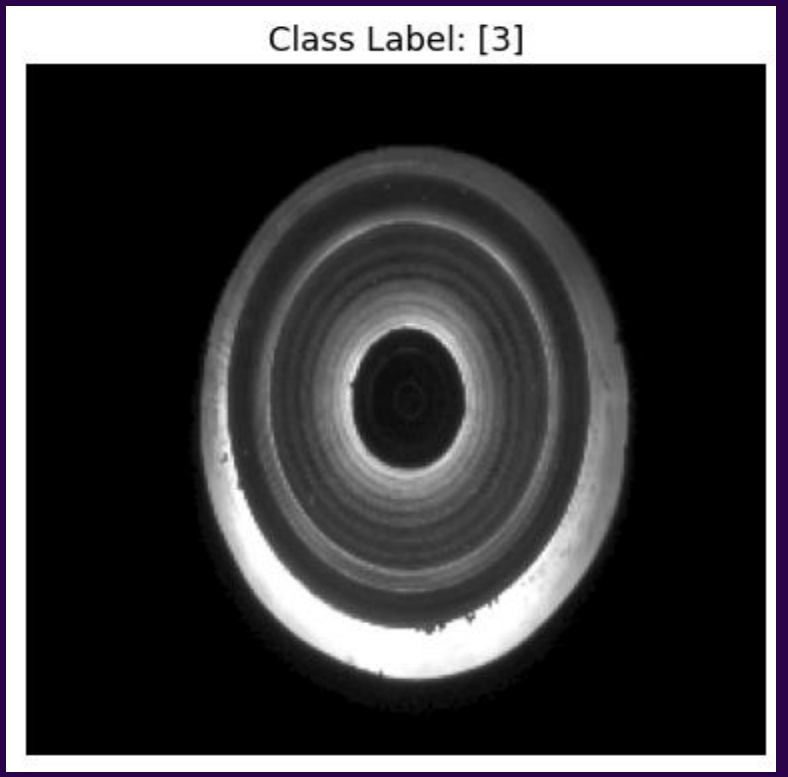
APPROACHS

1. Gamma transform
2. Gamma for augmentation
3. Only Normal sample
4. crop

GAMMA TRANSFORM

GAMMA TRANSFORM

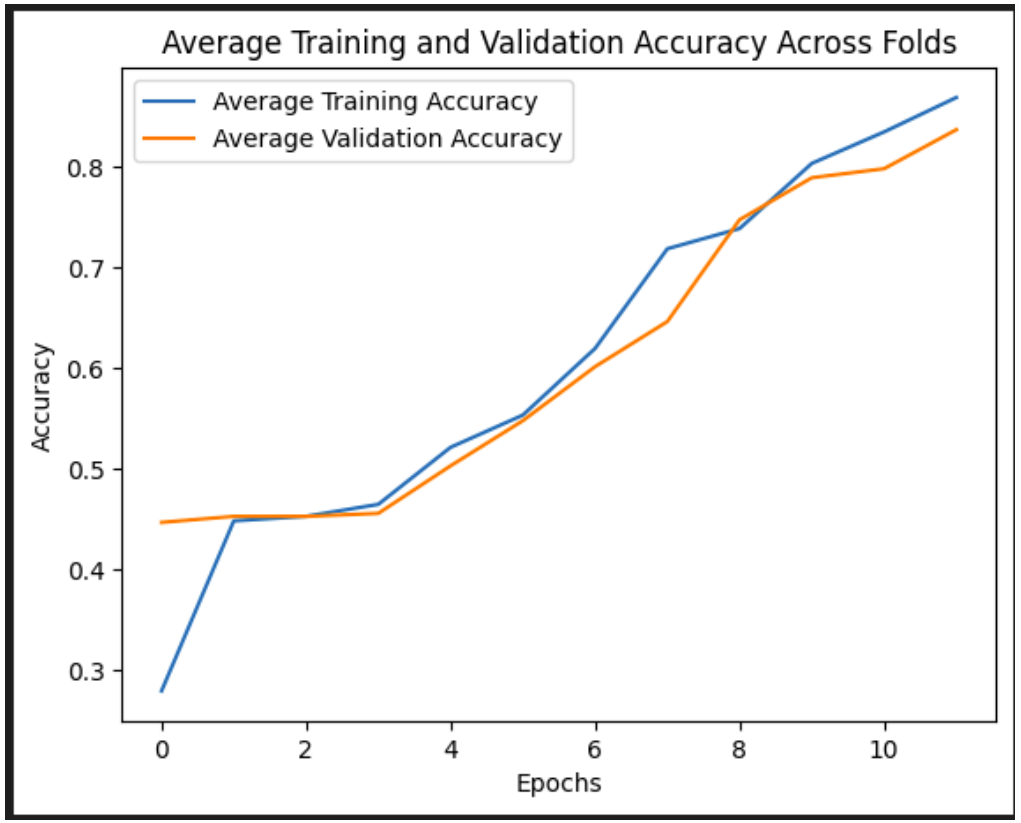
- Apply gamma on all dataset
- Do not use use original data

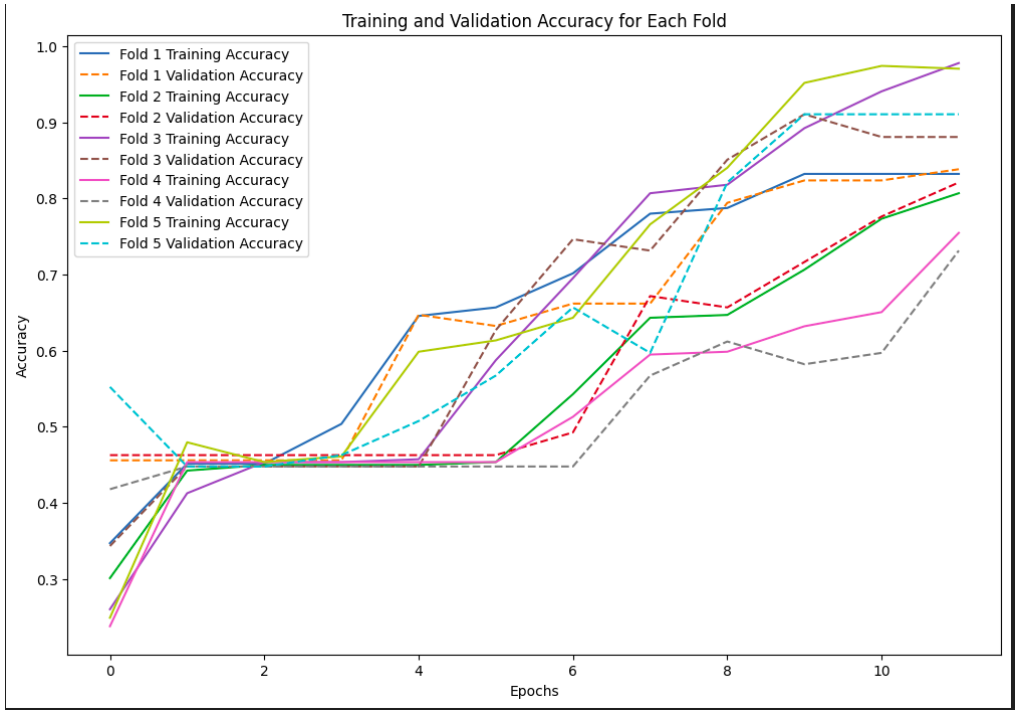


CROSS VALIDATION

- Use StratifiedKFold
- 5 FOLDS
- Save weights of best cross-validation
- ResNet50 classifier
- Loss function: sparse categorical crossentropy
- Optimizer: SGD
- Learning rate: Cosine Decay with Warm-Up
  - weight_decay=5e-4,
  - momentum=0.9
- Train for 12 epochs
- gamma_value = 1.5

RESULT FOR CROSS VALIDATION


Average Training and Validation Accuracy Across Folds
Average Training Accuracy
Average Validation Accuracy
Accuracy
Epochs
0.3
0.4
0.5
0.6
0.7
0.8
0
2
4
6
8
10

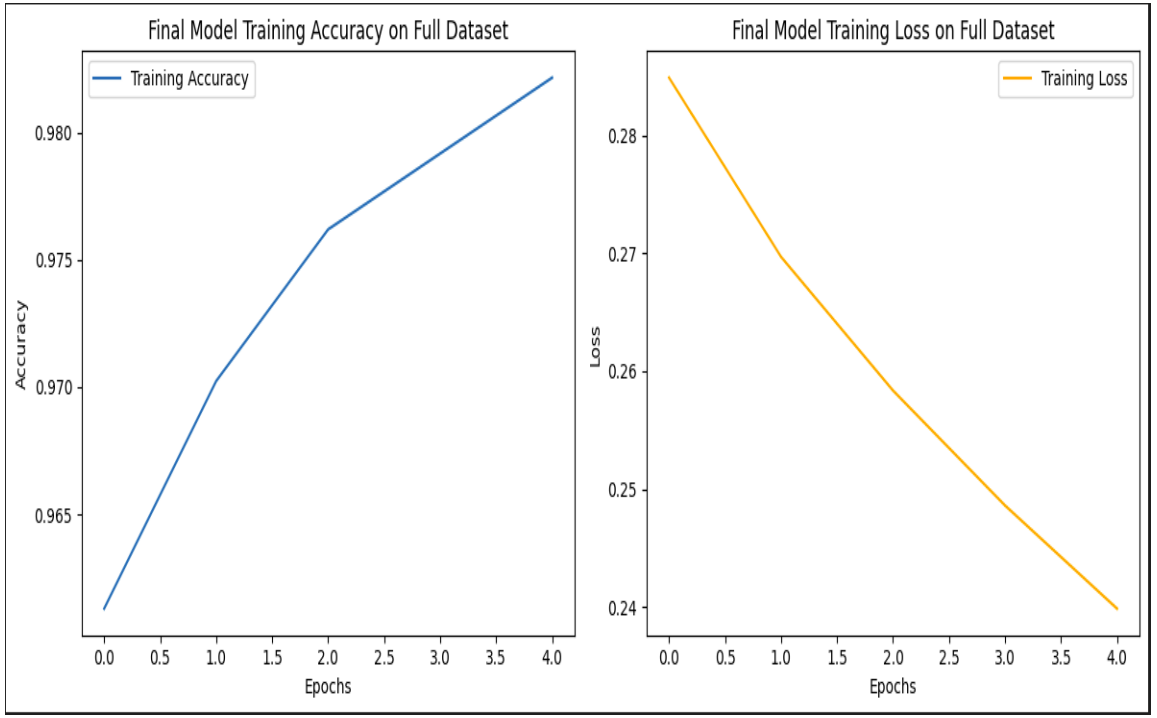

Training and Validation Accuracy for Each Fold
Fold 1 Training Accuracy
Fold 1 Validation Accuracy
Fold 2 Training Accuracy
Fold 2 Validation Accuracy
Fold 3 Training Accuracy
Fold 3 Validation Accuracy
Fold 4 Training Accuracy
Fold 4 Validation Accuracy
Fold 5 Training Accuracy
Fold 5 Validation Accuracy
Accuracy
Epochs
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
0
2
4
6
8
10

MAIN MODEL

- Load best weights from cross-validation
- Use whole dataset for train
- Loss function: sparse categorical crossentropy
- Optimizer: SGD
- Learning rate: fixe(0.01)
- Train for 5 epochs

TRAINING FINAL MODEL

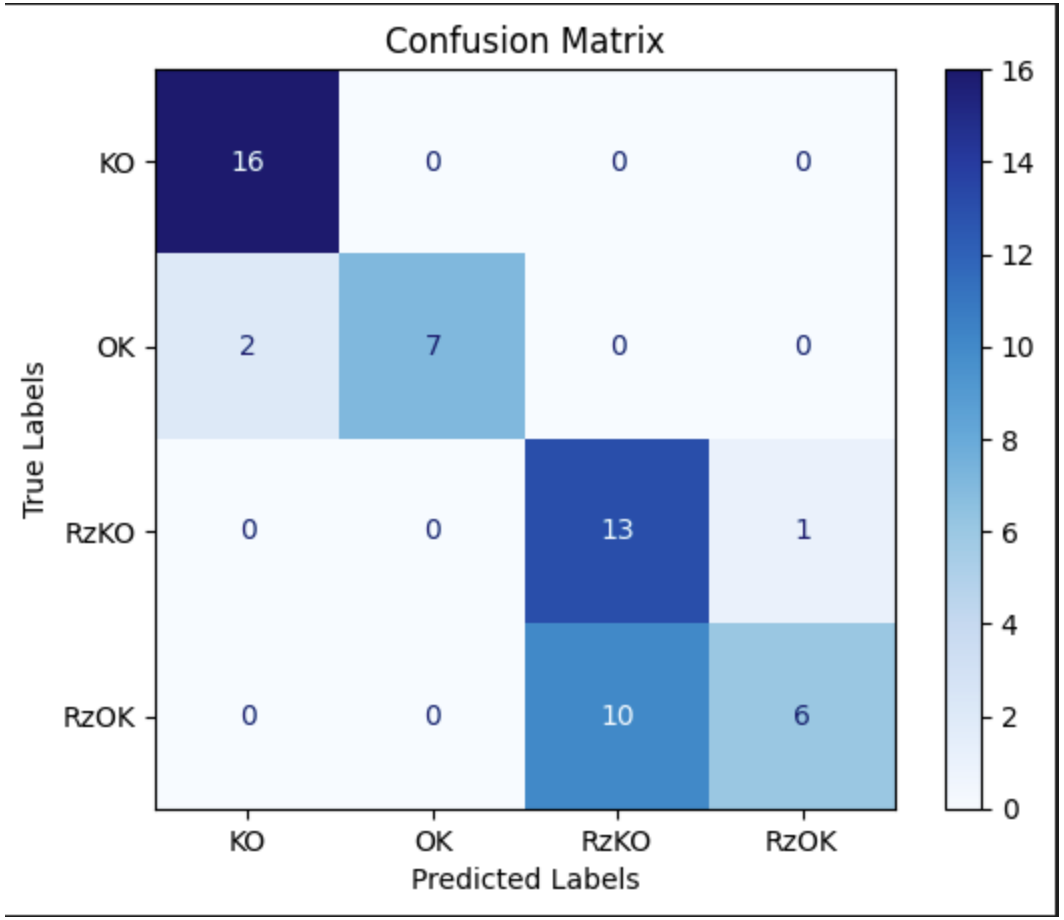
```
Epoch 1/5
11/11 ━━━━━━━━━━━━━━━━━━━━ 38s 3s/step - accuracy: 0.9594 - loss: 0.2748
Epoch 2/5
11/11 ━━━━━━━━━━━━━━━━━━━━ 10s 206ms/step - accuracy: 0.9704 - loss: 0.2598
Epoch 3/5
11/11 ━━━━━━━━━━━━━━━━━━━━ 3s 208ms/step - accuracy: 0.9756 - loss: 0.2481
Epoch 4/5
11/11 ━━━━━━━━━━━━━━━━━━━━ 3s 209ms/step - accuracy: 0.9778 - loss: 0.2383
Epoch 5/5
11/11 ━━━━━━━━━━━━━━━━━━━━ 3s 209ms/step - accuracy: 0.9788 - loss: 0.2298
```



RESULT FOR FINAL MODEL

```
Found 55 files belonging to 4 classes.
Class names: ['KO', 'OK', 'RzKO', 'RzOK']
2/2 ━━━━━━━━━━━━━━━━━━━━ 12s 6s/step - accuracy: 0.7903 - loss: 0.6748

Test Loss: 0.6993468999862671
Test Accuracy: 0.7636363506317139
1/1 ━━━━━━━━━━━━━━━━━━━━ 2s 2s/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 2s 2s/step
```



GAMMA FOR AUGMENTATION

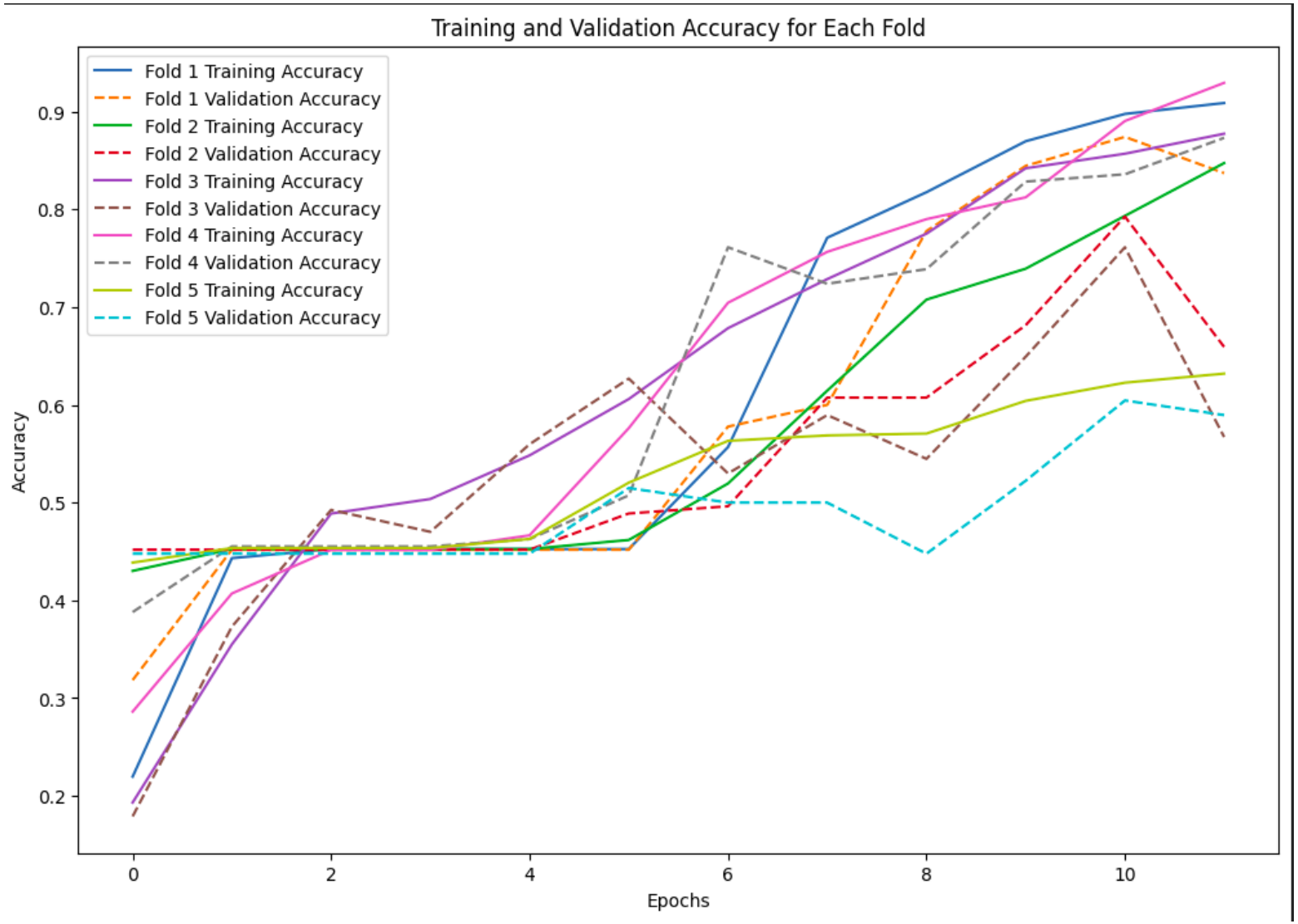
GAMMA FOR AUGMENTATION

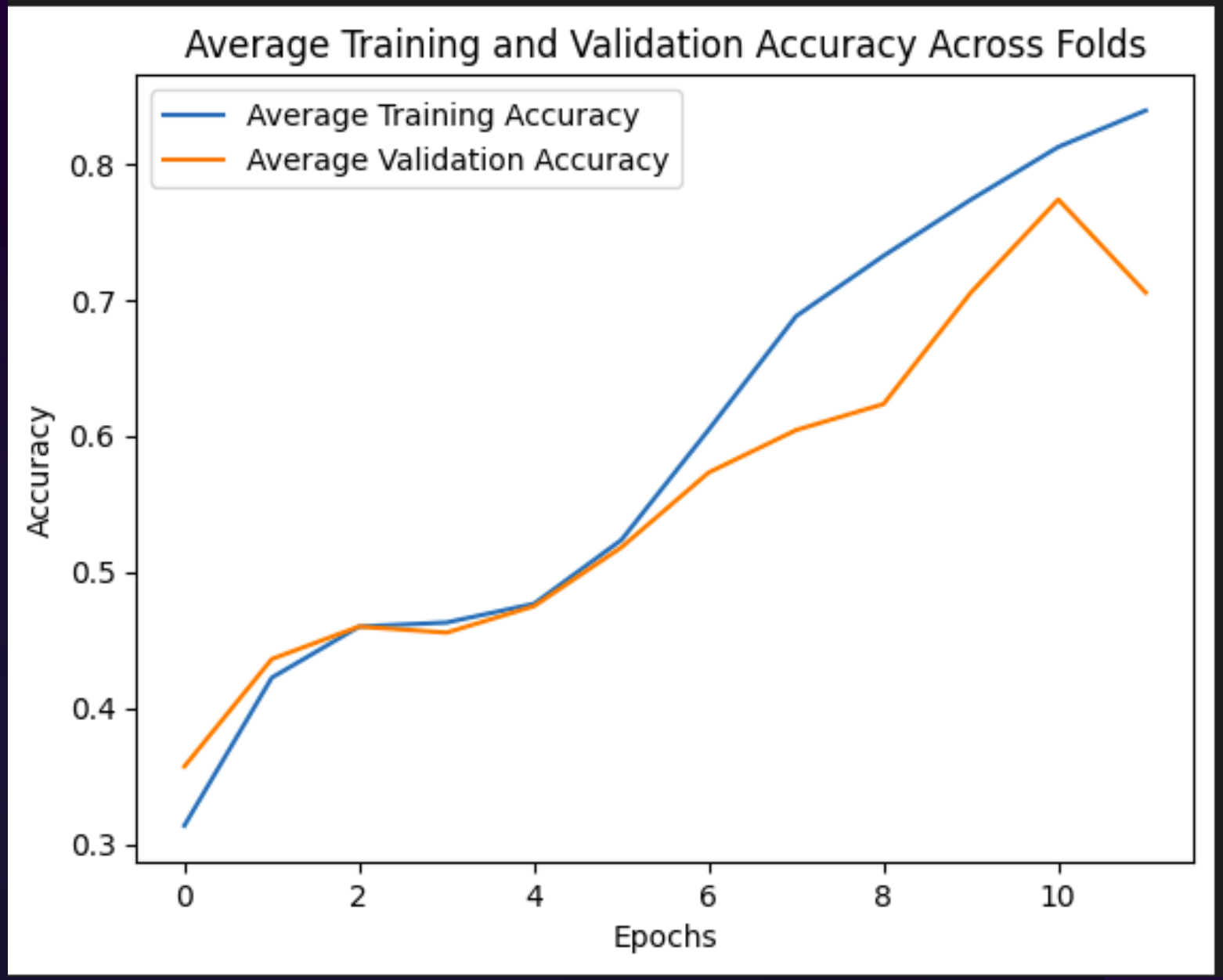
- Apply gamma on all dataset
- add new images to dataset
- Dataset contain normal images + gamma version
- gamma_value = 1.5

CROSS VALIDATION

- Use StratifiedKFold
- 5 FOLDS
- Save weights of best cross-validation
- ResNet50 classifier
- Loss function: sparse categorical crossentropy
- Optimizer: SGD
- Learning rate: Cosine Decay with Warm-Up
  - weight_decay=5e-4,
  - momentum=0.9
- Train for 12 epochs

RESULT FOR CROSS VALIDATION

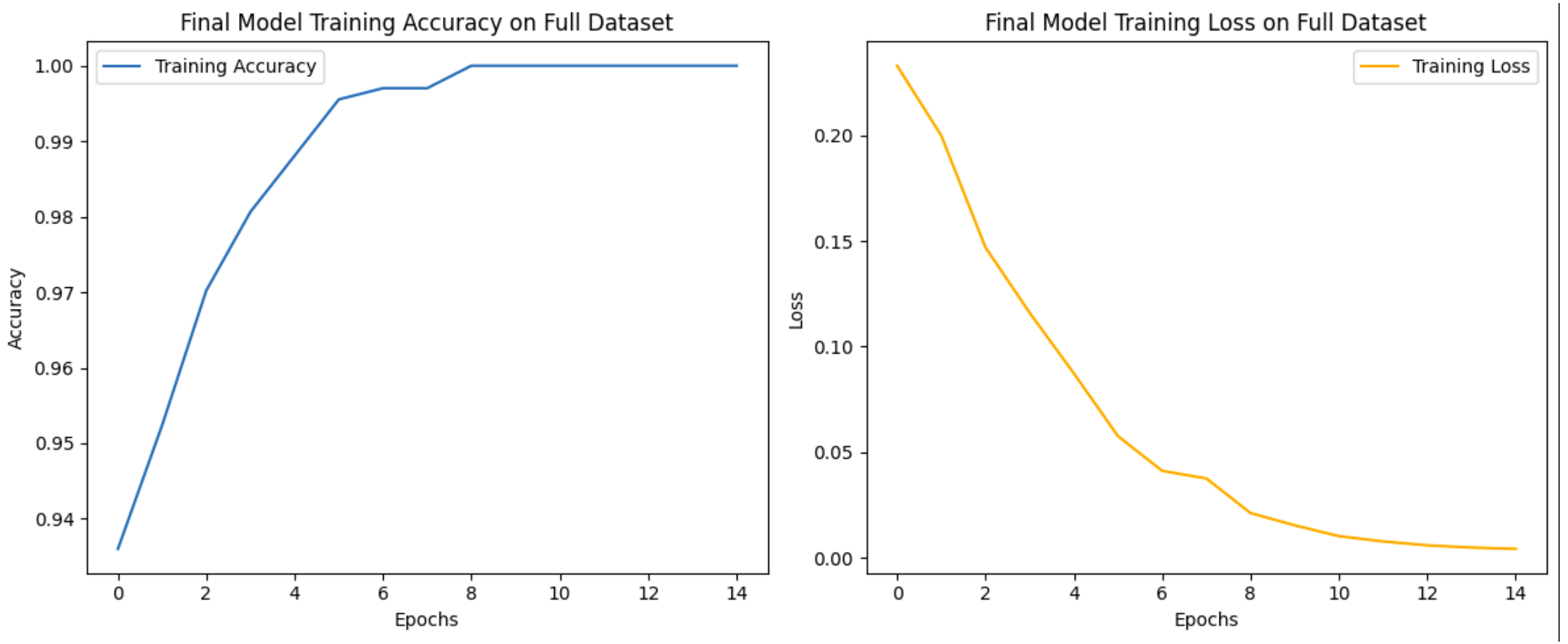

Training and Validation Accuracy for Each Fold
Fold 1 Training Accuracy
Fold 1 Validation Accuracy
Fold 2 Training Accuracy
Fold 2 Validation Accuracy
Fold 3 Training Accuracy
Fold 3 Validation Accuracy
Fold 4 Training Accuracy
Fold 4 Validation Accuracy
Fold 5 Training Accuracy
Fold 5 Validation Accuracy
Accuracy
Epochs
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
0
2
4
6
8
10

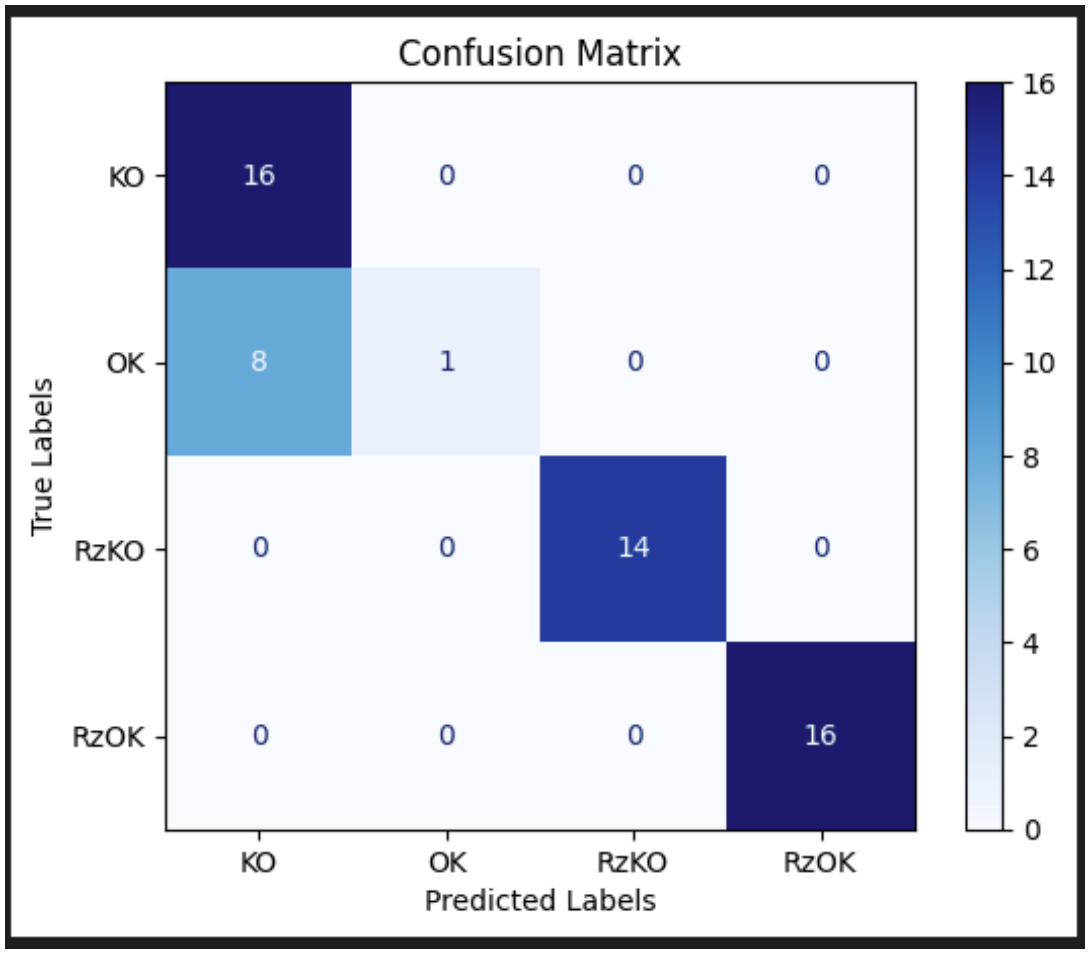

Average Training and Validation Accuracy Across Folds
Average Training Accuracy
Average Validation Accuracy
Accuracy
0.3
0.4
0.5
0.6
0.7
0.8
0
2
4
6
8
10
Epochs

MAIN MODEL

- Load best weights from cross-validation
- Use whole dataset for train
- Loss function: sparse categorical crossentropy
- Optimizer: SGD
- Learning rate: Cosine Decay with Warm-Up
  - weight_decay=5e-4,
  - momentum=0.9
- Train for 15 epochs

TRAINING FINAL MODEL


Final Model Training Accuracy on Full Dataset
Training Accuracy
Accuracy
Epochs
1.00
0.99
0.98
0.97
0.96
0.95
0.94
0
2
4
6
8
10
12
14
Final Model Training Loss on Full Dataset
Training Loss
Loss
Epochs
0.20
0.15
0.10
0.05
0.00




```
Found 55 files belonging to 4 classes.
Class names: ['KO', 'OK', 'RzKO', 'RzOK']
2/2 ━━━━━━━━━━━━━━━━━━━━ 3s 733ms/step - accuracy: 0.8509 - loss: 1.1111

Test Loss: 1.1123638153076172
Test Accuracy: 0.8545454144477844
1/1 ━━━━━━━━━━━━━━━━━━━━ 2s 2s/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 1s 1s/step
```

ONLY NORMAL SAMPLE

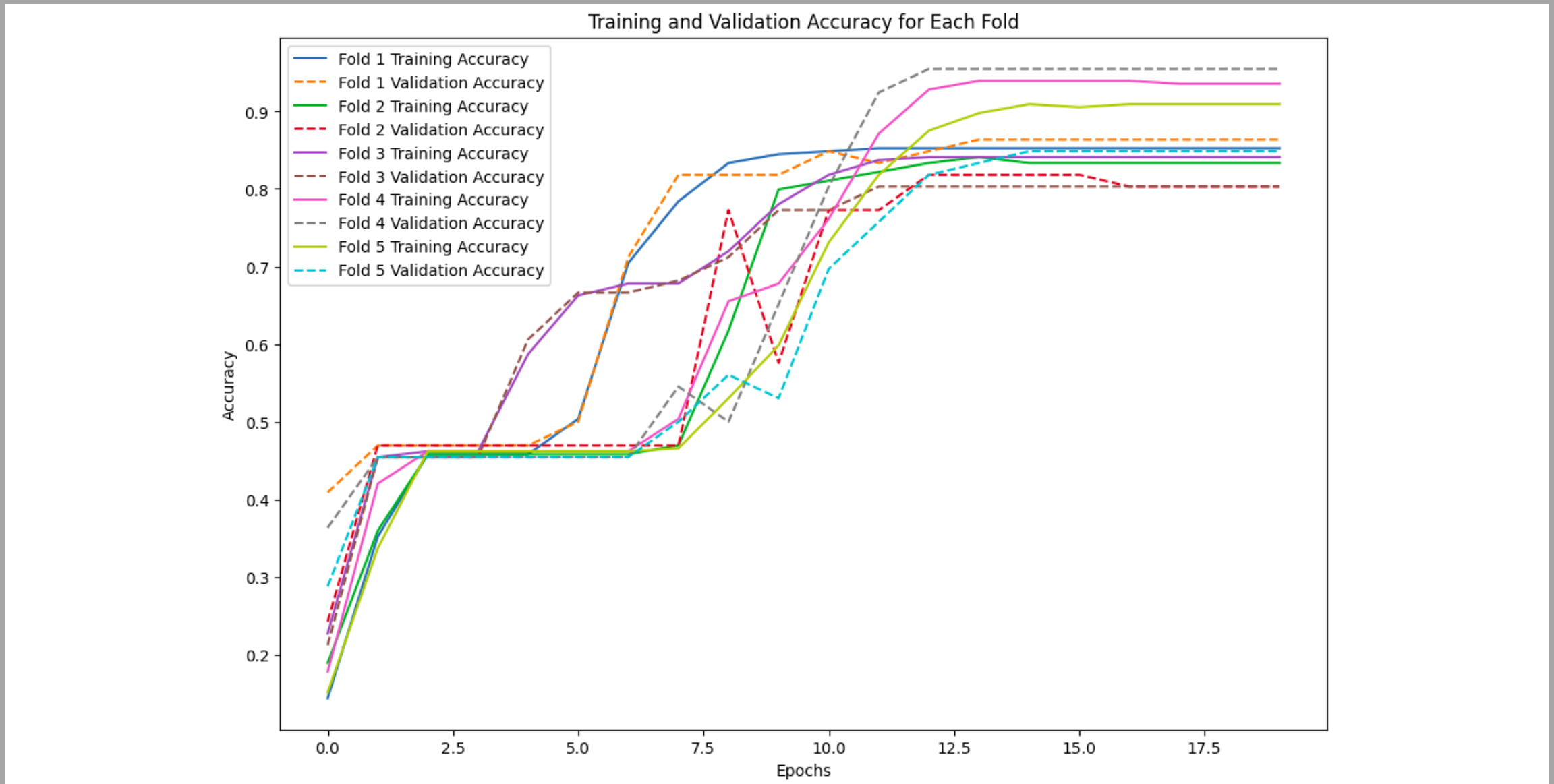
NORMAL SAMPEL

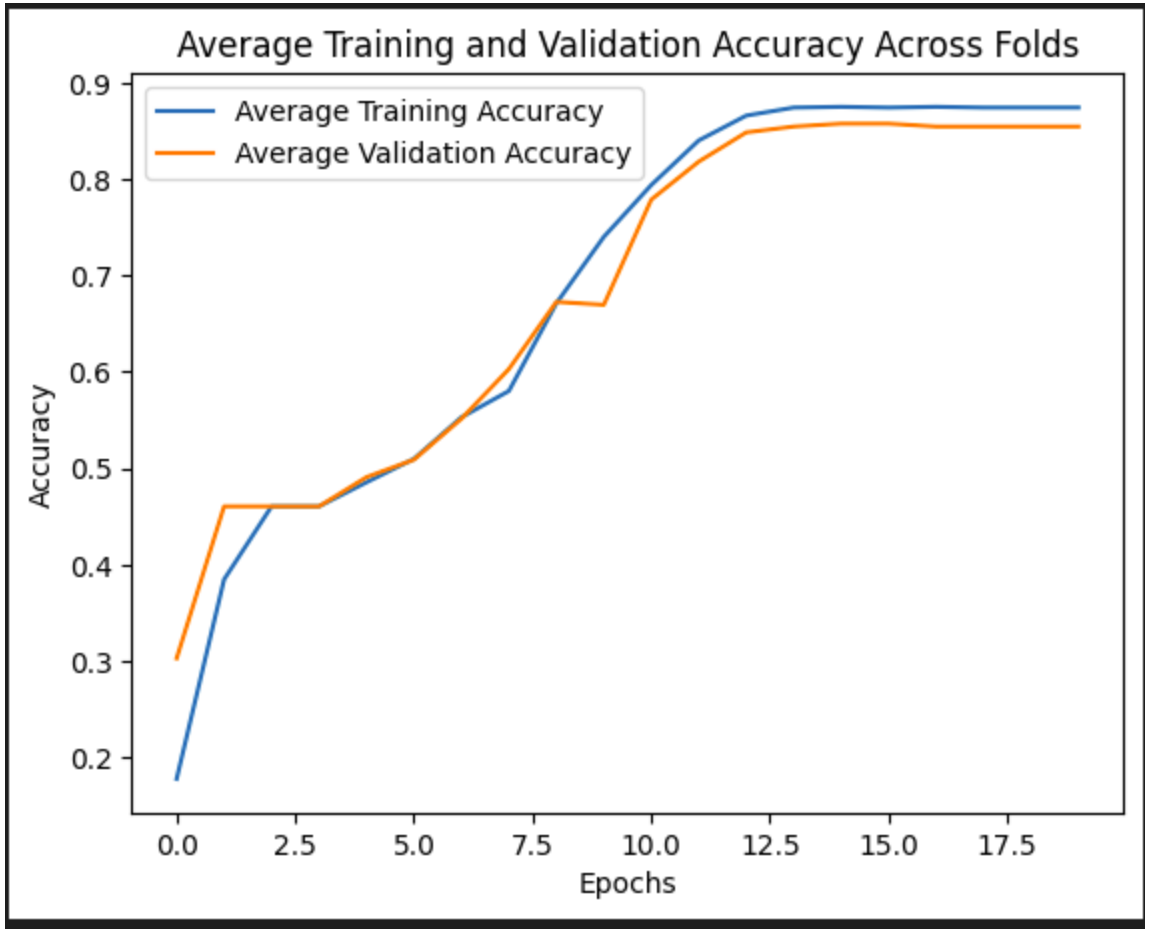
- In this model, I use only normal data without applying any modifications to the samples or increasing the dataset through augmentation.

CROSS VALIDATION

- Use StratifiedKFold
- 5 FOLDS
- Save weights of best cross-validation
- ResNet50 classifier
- Loss function: sparse categorical crossentropy
- Optimizer: SGD
- Learning rate: Cosine Decay with Warm-Up
  - weight_decay=5e-4,
  - momentum=0.9
- Train for 20 epochs

RESULT FOR CROSS VALIDATION

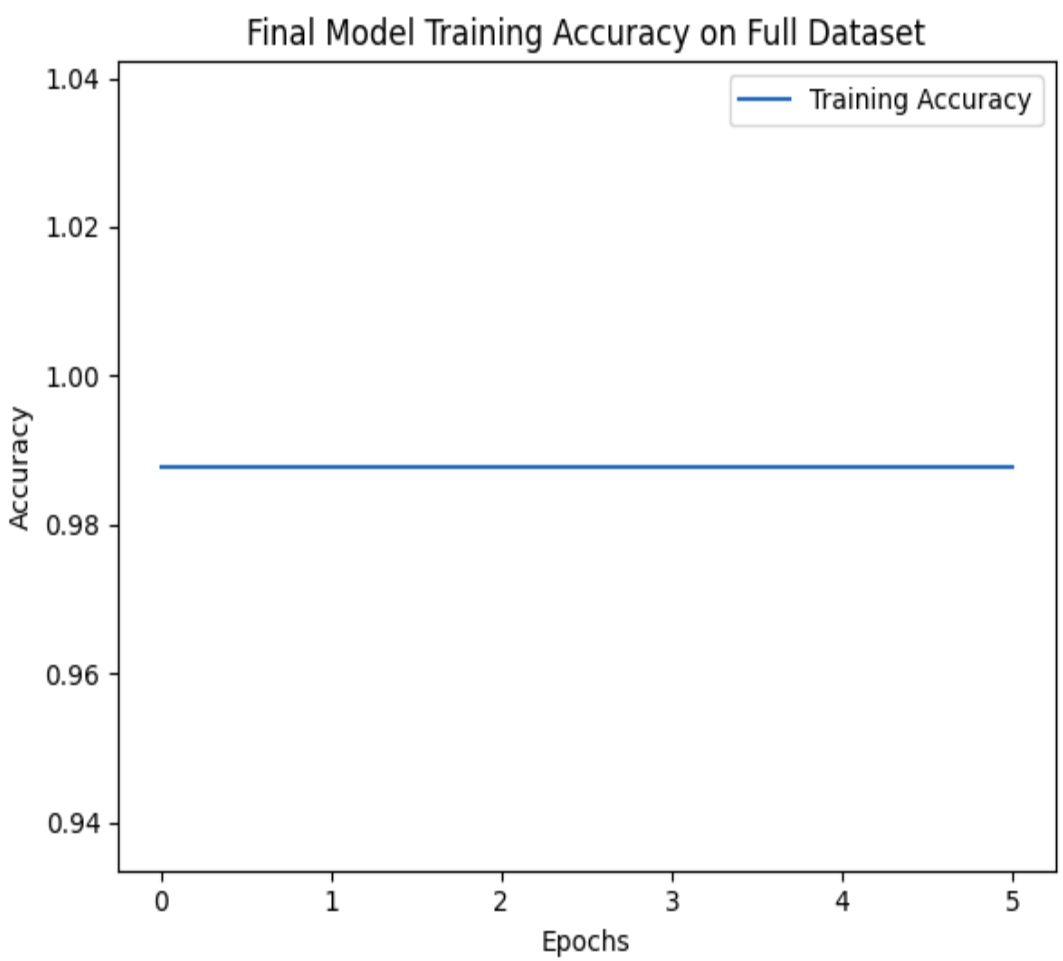

Training and Validation Accuracy for Each Fold
Fold 1 Training Accuracy
Fold 1 Validation Accuracy
Fold 2 Training Accuracy
Fold 2 Validation Accuracy
Fold 3 Training Accuracy
Fold 3 Validation Accuracy
Fold 4 Training Accuracy
Fold 4 Validation Accuracy
Fold 5 Training Accuracy
Fold 5 Validation Accuracy
Accuracy
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Epochs

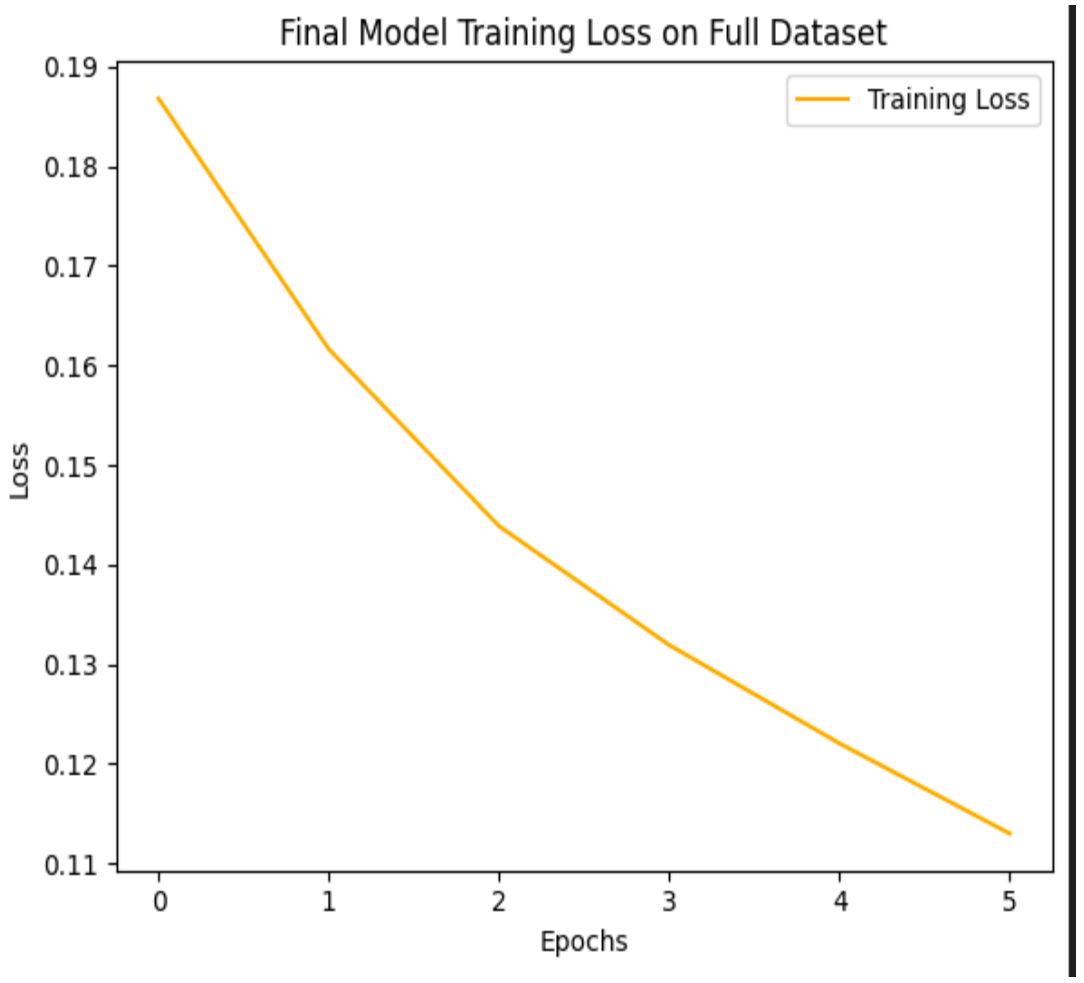

Average Training and Validation Accuracy Across Folds
Average Training Accuracy
Average Validation Accuracy
Accuracy
Epochs
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

MAIN MODEL

- Load best weights from cross-validation
- Use whole dataset for train
- Loss function: sparse categorical crossentropy
- Optimizer: SGD
- Learning rate: Fixed (0.01)

TRAINING FINAL MODEL


Final Model Training Accuracy on Full Dataset
Training Accuracy
Accuracy
Epochs
1.04
1.02
1.00
0.98
0.96
0.94
0
1
2
3
4
5

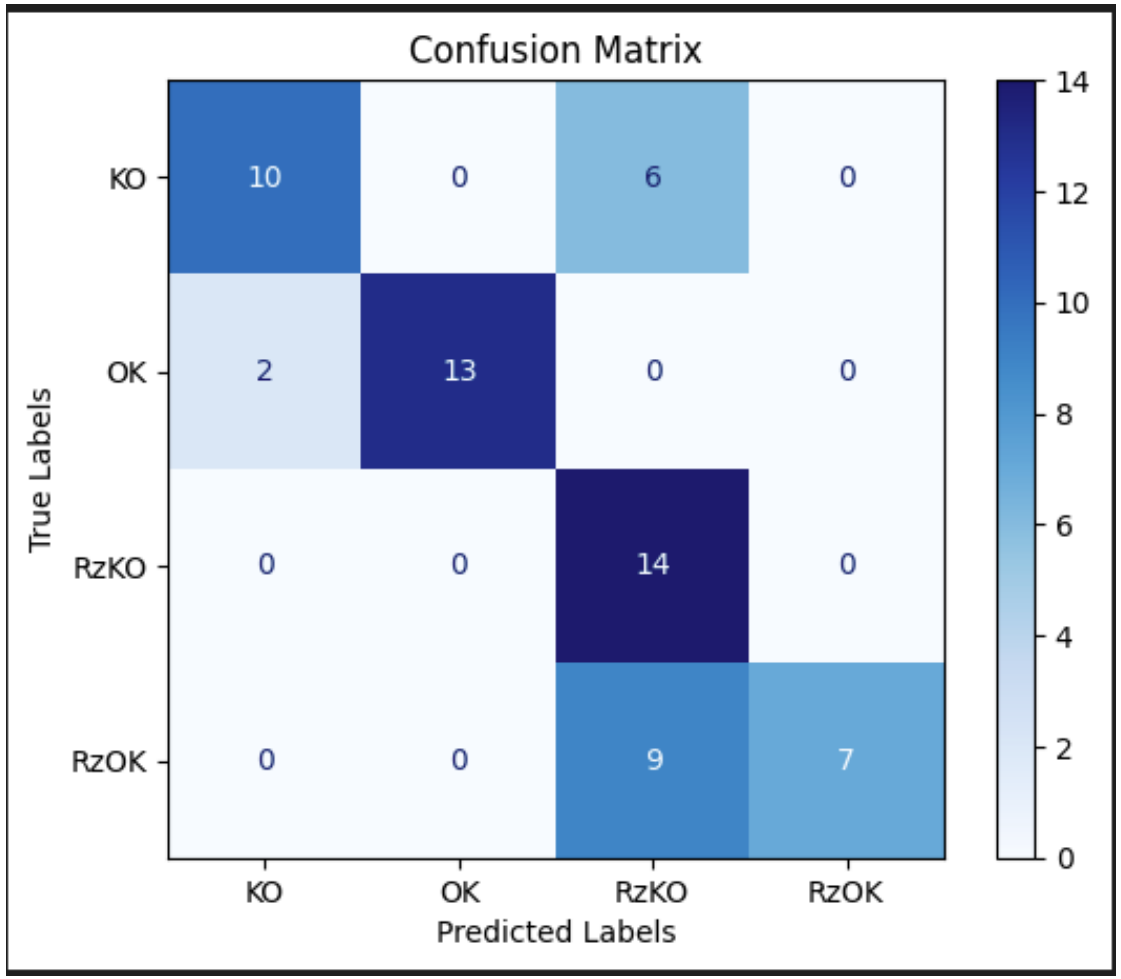

Final Model Training Loss on Full Dataset
Training Loss
Loss
Epochs
0.19
0.18
0.17
0.16
0.15
0.14
0.13
0.12
0.11
0
1
2
3
4
5

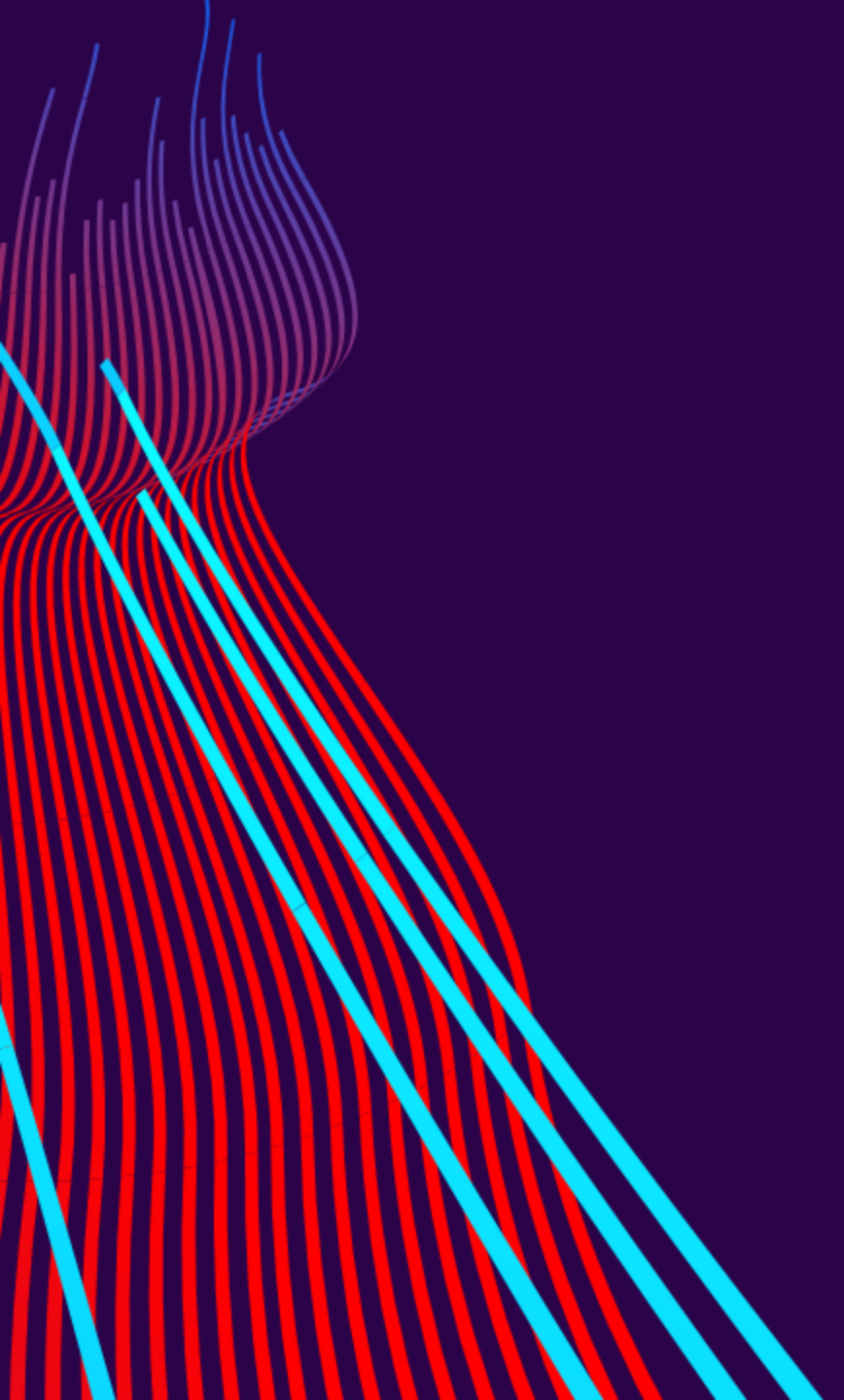
RESULT FOR FINAL MODEL

```
Found 61 files belonging to 4 classes.
Class names: ['KO', 'OK', 'RzKO', 'RzOK']
2/2 ━━━━━━━━━━━━━━━━━━━━ 3s 689ms/step - accuracy: 0.7205 - loss: 0.5924

Test Loss: 0.5987216234207153
Test Accuracy: 0.7213115096092224
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 65ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 34ms/step

<Figure size 800x800 with 0 Axes>
```

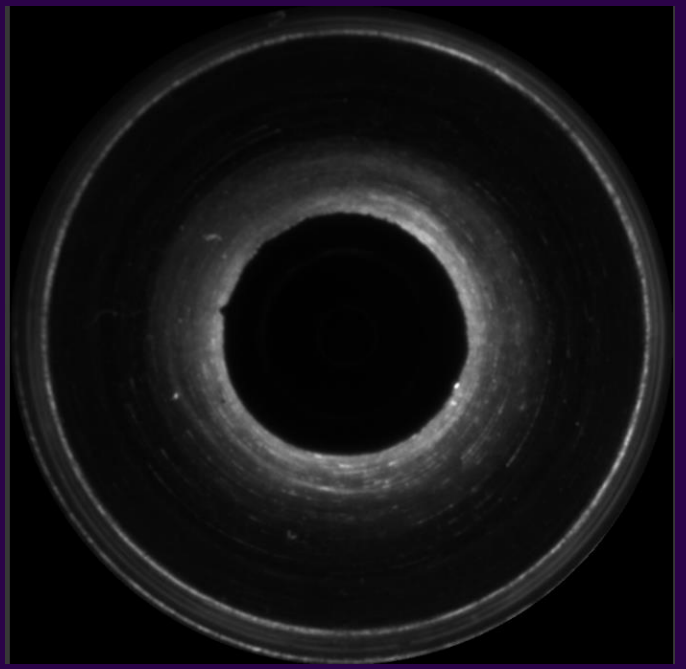




CROP

CROP

- In this model, I crop central part of all images in our dataset to only train model on the part which we expect anomaly
- Hese you can see result after apply crop function on images in dataset



- The model performs very poorly, so I have not included the results here.