REAL VS FAKE IMAGES CLASSIFICATION THROUGH AUTOENCODING

Seth Dasuki & Mai Hashad

OBJECTIVE

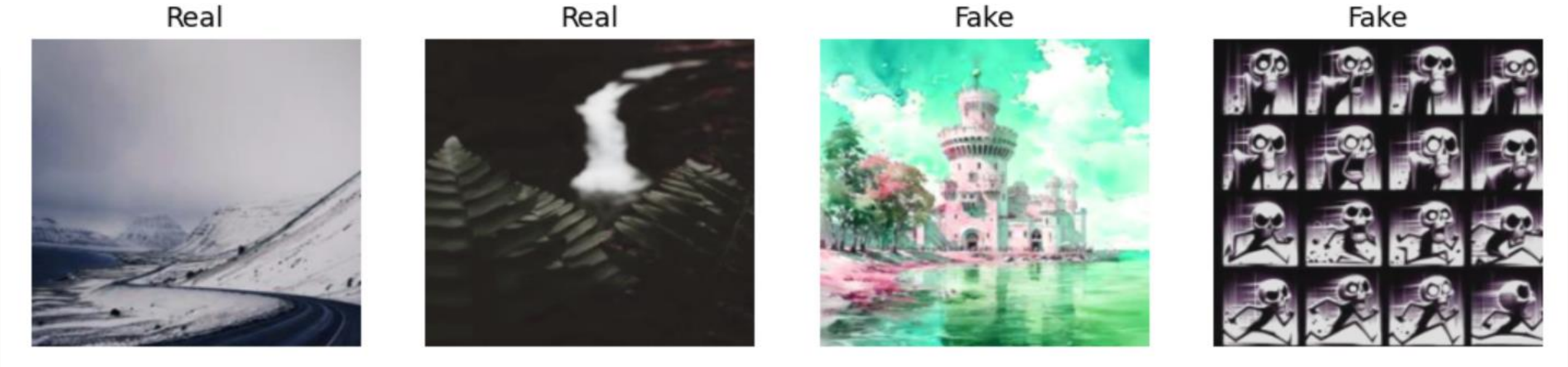
Use an Autoencoder to classify an image and determine whether it is a real image, or an AI generated image

DATASET INFORMATION

- The source dataset from Kaggle includes 30,000 samples from both real image sets and those generated by Generative AI tools from Midjourney DALL-E, and Stable Diffusion (60,000 total images, 52GB).
- Samples were pre-divided into a 80/20 Train/Test split.

- This project ran on 20% of the dataset in Kaggle ~ 12000 images (½ real, ½ fake) - 11 GB

PROJECT GOALS



Preprocess Images

Modify dataset images to appropriately scale and re-encode color palettes to ensure efficient processing.

Train for High Accuracy

Use techniques discussed in class to achieve a high, and ideally not overfitted, accuracy against a training and test data set.

Investigate Performance on greyscale images

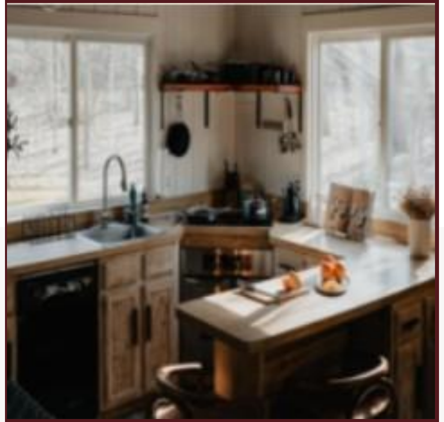
Experiment with generic CNNs, Transfer learning with popular CNN libraries, and GANs.

STANDARD OPERATIONS

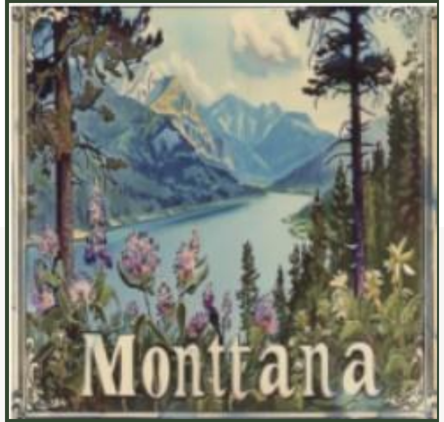
- Batch size 16
- 200 Epochs for model training
- Learning rate = 0.0003
- Started with 5% sample and moved to large sample size with initial success
- Randomization of samples seeded for repeatability
- T4 GPU

IMAGES

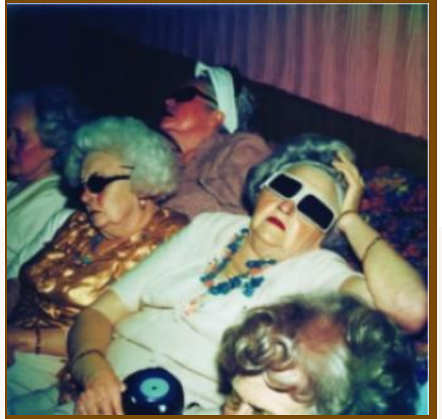
REAL



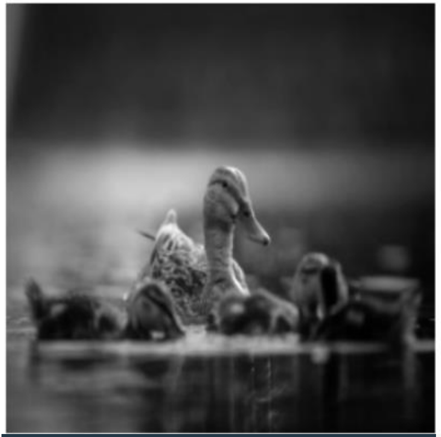
FAKE



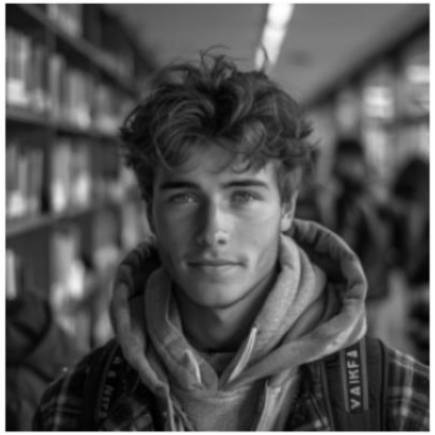
FAKE



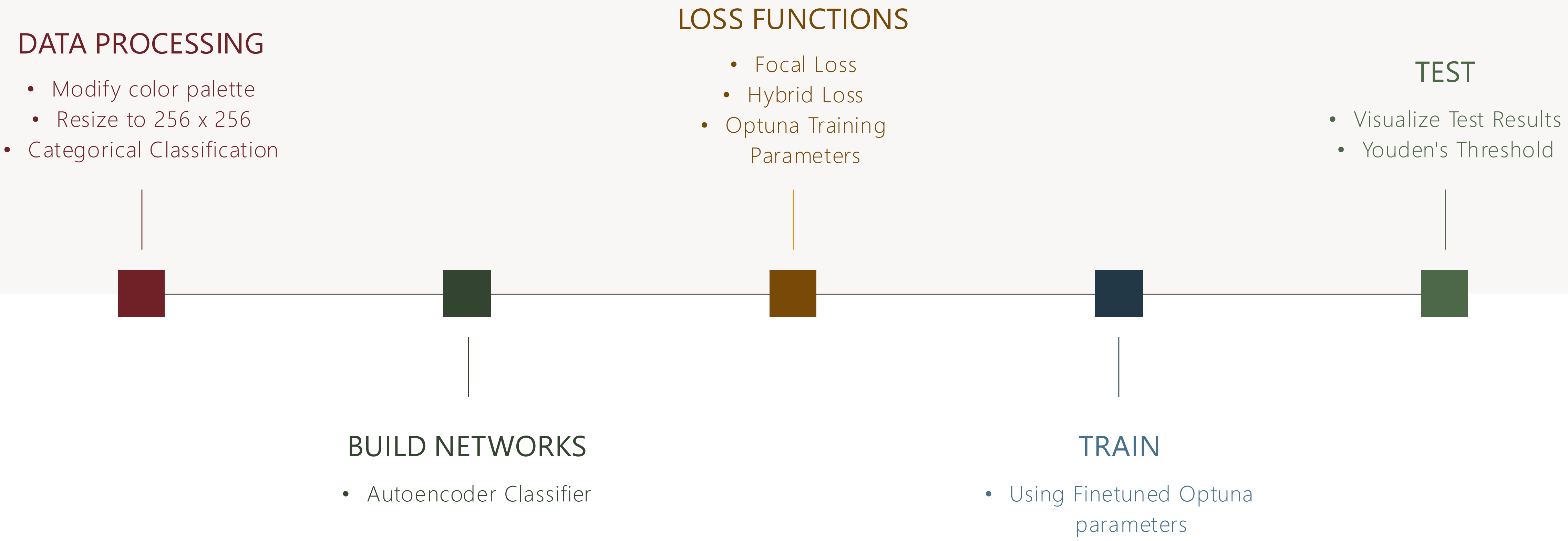
REAL



FAKE



PSEUDOCODE – NETWORKS (RGB & GRAYSCALE)



DATA PROCESSING

- Modify color palette
- Resize to 256 x 256
- Categorical Classification

BUILD NETWORKS

- Autoencoder Classifier

LOSS FUNCTIONS

- Focal Loss
- Hybrid Loss
- Optuna Training Parameters

TRAIN

- Using Finetuned Optuna parameters

TEST

- Visualize Test Results
- Youden's Threshold

RGB - GENERAL MODEL SUMMARY

Autoencoder

- Learn features of input images
- Series of convolutional then transpose layers
- Transfer Encoder
- Batch Normalization in Decoder

Classifier Head

- Predict the class of images using learned features
- Fully Connected Dense Layers
- Global Average Pooling & Dropout

Reconstruction

- Unsupervised task to reconstruct input images
- MSE LOSS
- Hybrid Loss – L1, LPIPS, SSIM

Classification

- Supervised task to classify images
- BCE Loss
- Smoothed BCE Loss
- Focal Loss

(1) CONVOLUTIONAL AE

Model Architecture

- Encoder: 4 Layers of convolutional 2-D layers with ReLu
- Decoder: 4 Layers of transpose 2-D layers with ReLu and Sigmoid
- Classifier: Flatten with 2 fully connected layers with ReLu and Sigmoid

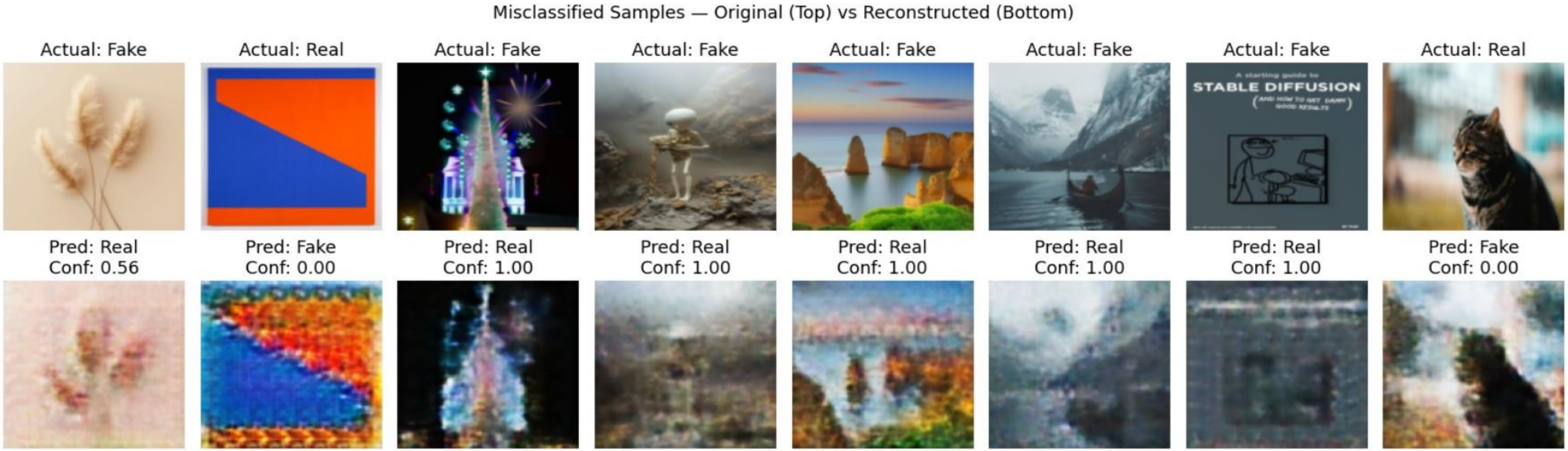
Loss Function

- Reconstruction Loss: MSE Loss
- Classification Loss: BCE Loss
- Total Loss = Reconstruction Loss + Classification Loss

Model Additions

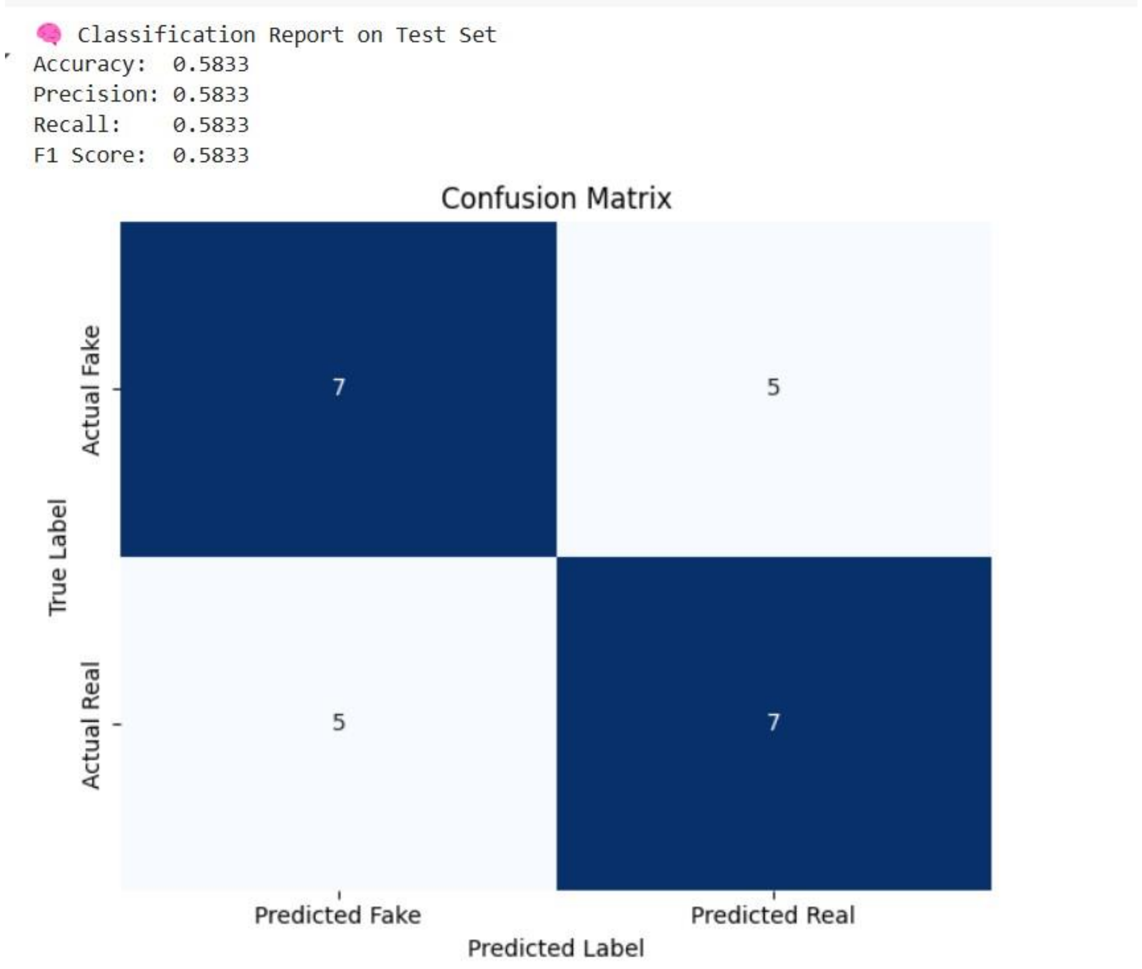
- Input Image size: 128 x 128
- Trained on 1% sample, 200 epochs

(1) CONVOLUTIONAL AE

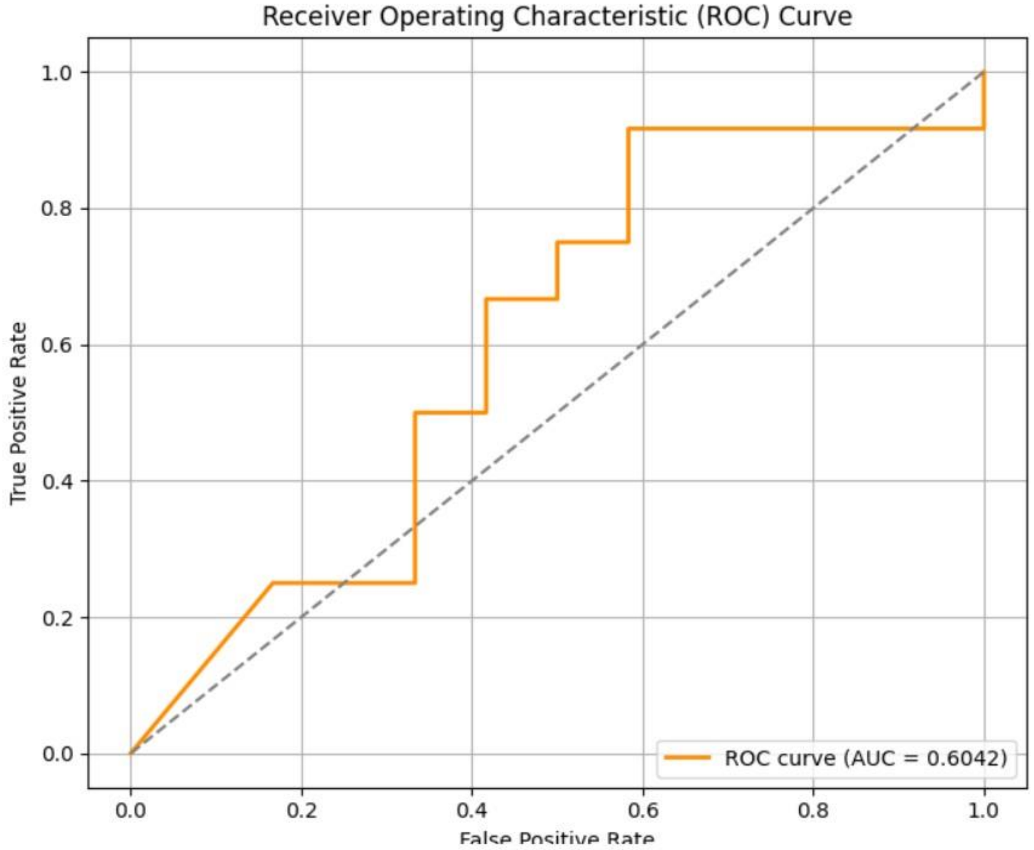


Misclassified Samples — Original (Top) vs Reconstructed (Bottom)

(1) CONVOLUTIONAL AE (15 MIN)



```
🧠 Classification Report on Test Set
Accuracy:  0.5833
Precision: 0.5833
Recall:    0.5833
F1 Score:  0.5833
```



(2) TRANSFER AE W/ HYBRID LOSS

Model Architecture

- Encoder: **Resnet 18** transferred encoder model
- Layered Decoder
- Fully Connected Classifier: **Global Average Pooling**

Loss Function

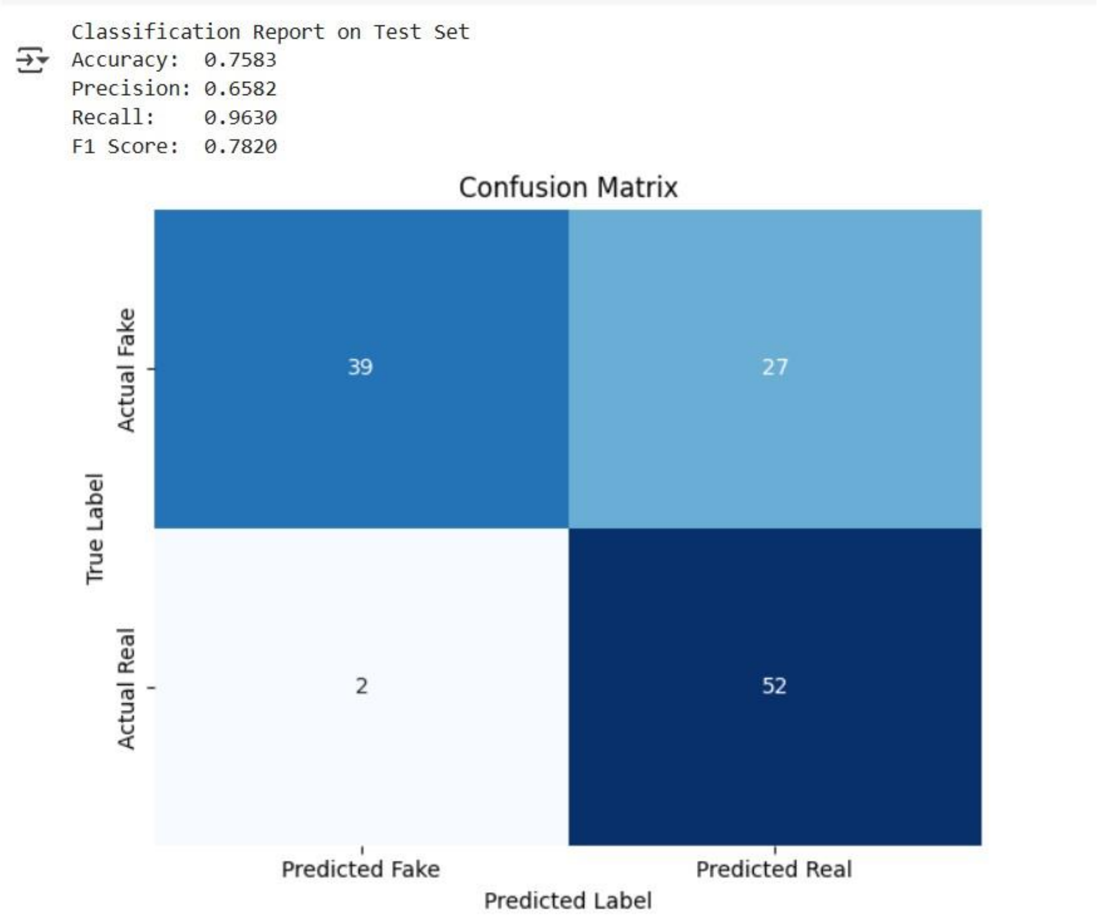
- Reconstruction Loss: **Hybrid Loss**
  - L1
  - SSIM
  - LPIPS
- Classification Loss: **Smoothed** BCE Loss
- Total Loss = recon loss * **r_weight** + class loss * **c_weight**

Model Additions

- Input Image size: **256 x 256**
- **Unfreeze** final 2 layers of transfer encoder during training
- **Freeze** classification loss on first 10% of epochs
- Trained on 5% sample, 200 epochs

(2) TRANSFER AE W/ HYBRID LOSS

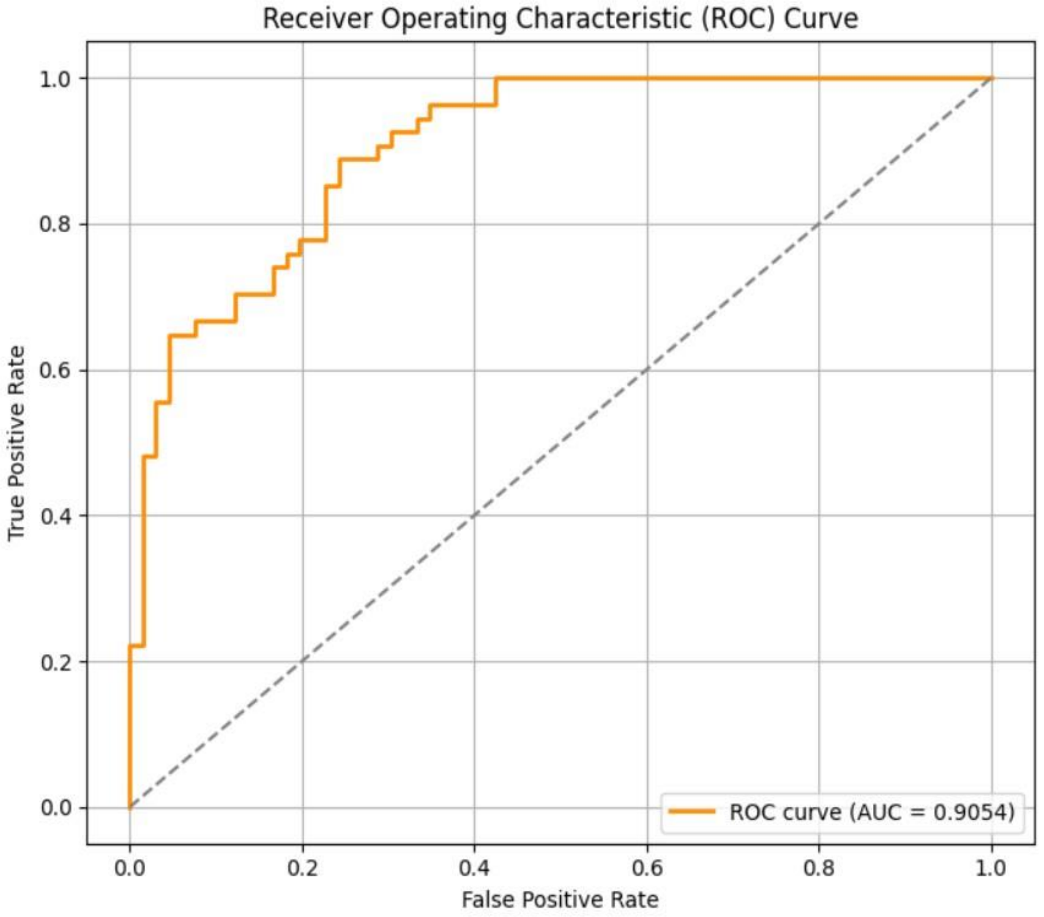
(2) TRANSFER AE W/ HYBRID LOSS (1.5 HOURS)



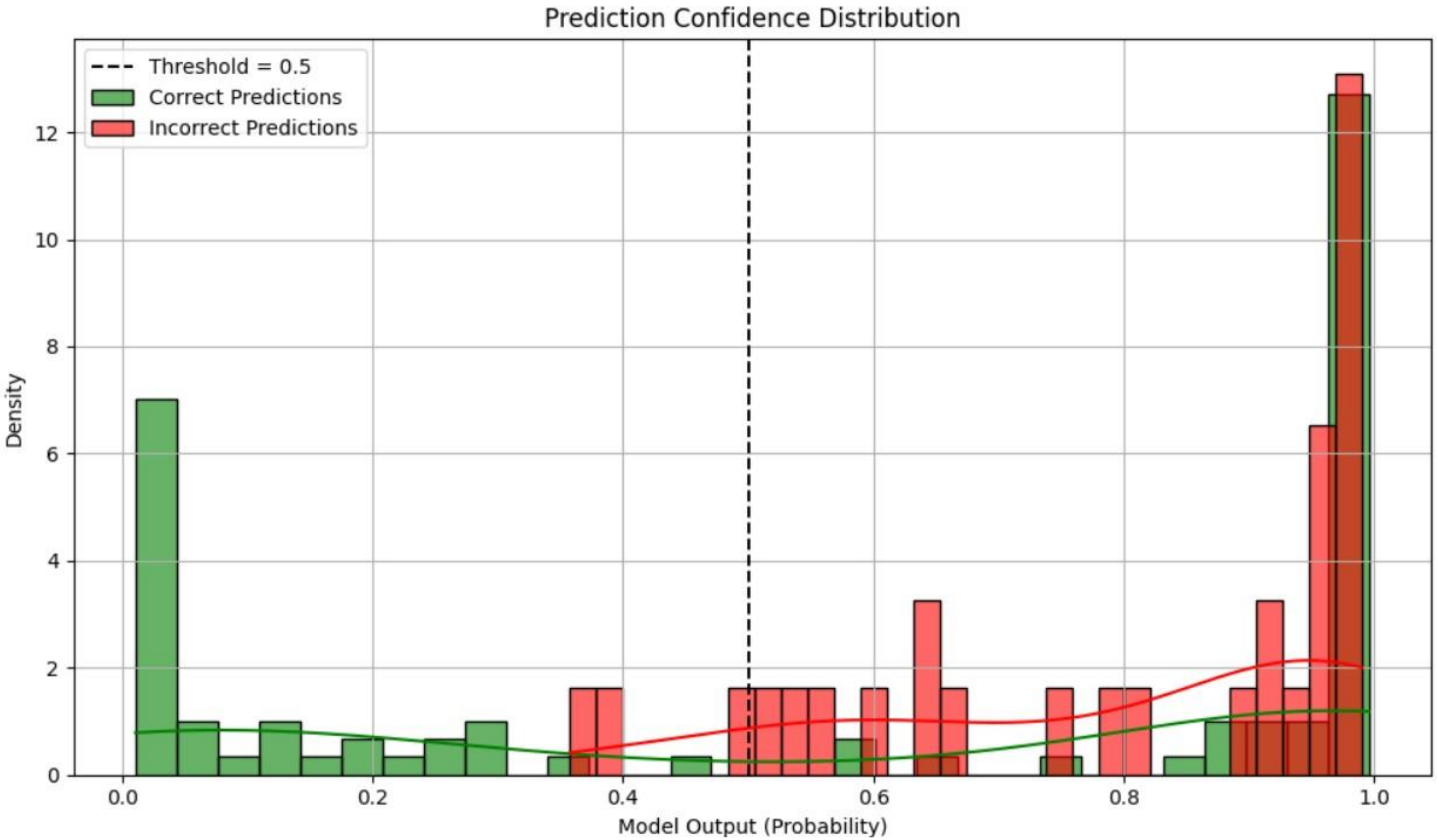
```
Classification Report on Test Set
Accuracy:  0.7583
Precision: 0.6582
Recall:    0.9630
F1 Score:  0.7820
```

Confusion Matrix

	Predicted Fake	Predicted Real
Actual Fake	39	27
Actual Real	2	52



(2) TRANSFER AE W/ HYBRID LOSS



(3) FOCAL LOSS W/ OPTUNA

Model Architecture

- Transferred Encoder: Unfrozen layers on training
- Layered Decoder: **Batch Normalization**
- Fully Connected Classifier: **Dropout Layers** & Global Average Pooling

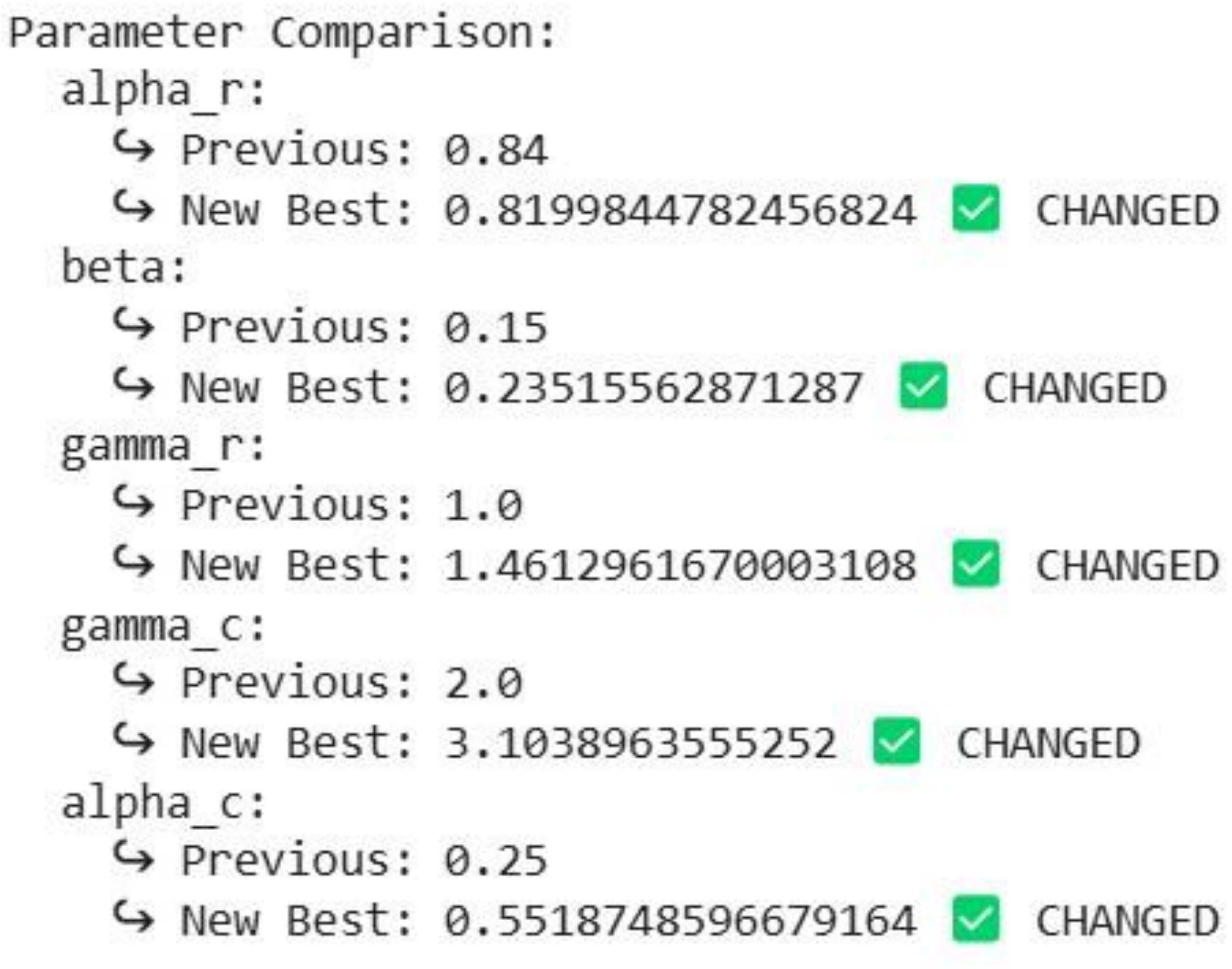
Loss Function

- Reconstruction Loss: Hybrid Loss
- Classification Loss: **Focal** Loss
- Total Loss = recon loss * r_weight + class loss * c_weight (**Progressive weighting** and freeze epochs)

Model Additions

- Unfreeze final **3** layers of transfer encoder during training
- **Optuna Optimization**
- Trained on 5% sample, 200 epochs

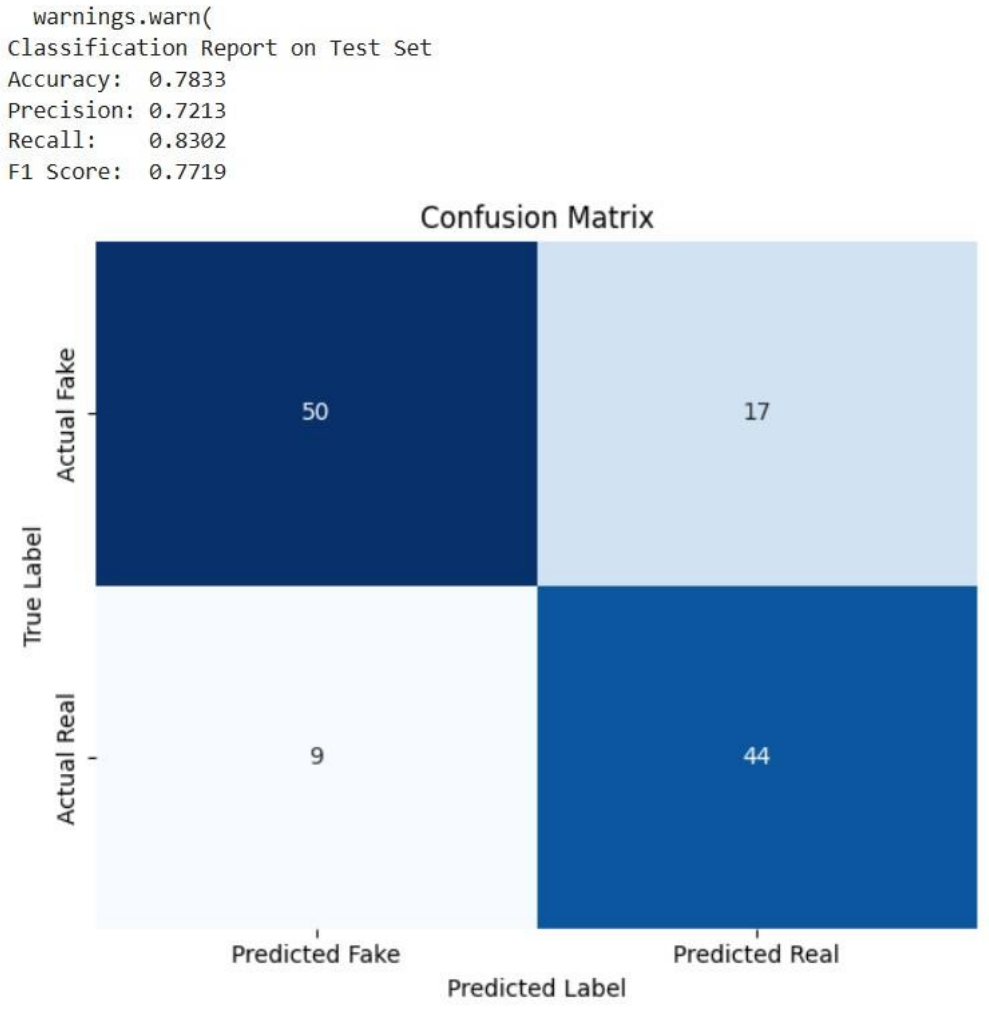
(3) FOCAL LOSS W/ OPTUNA



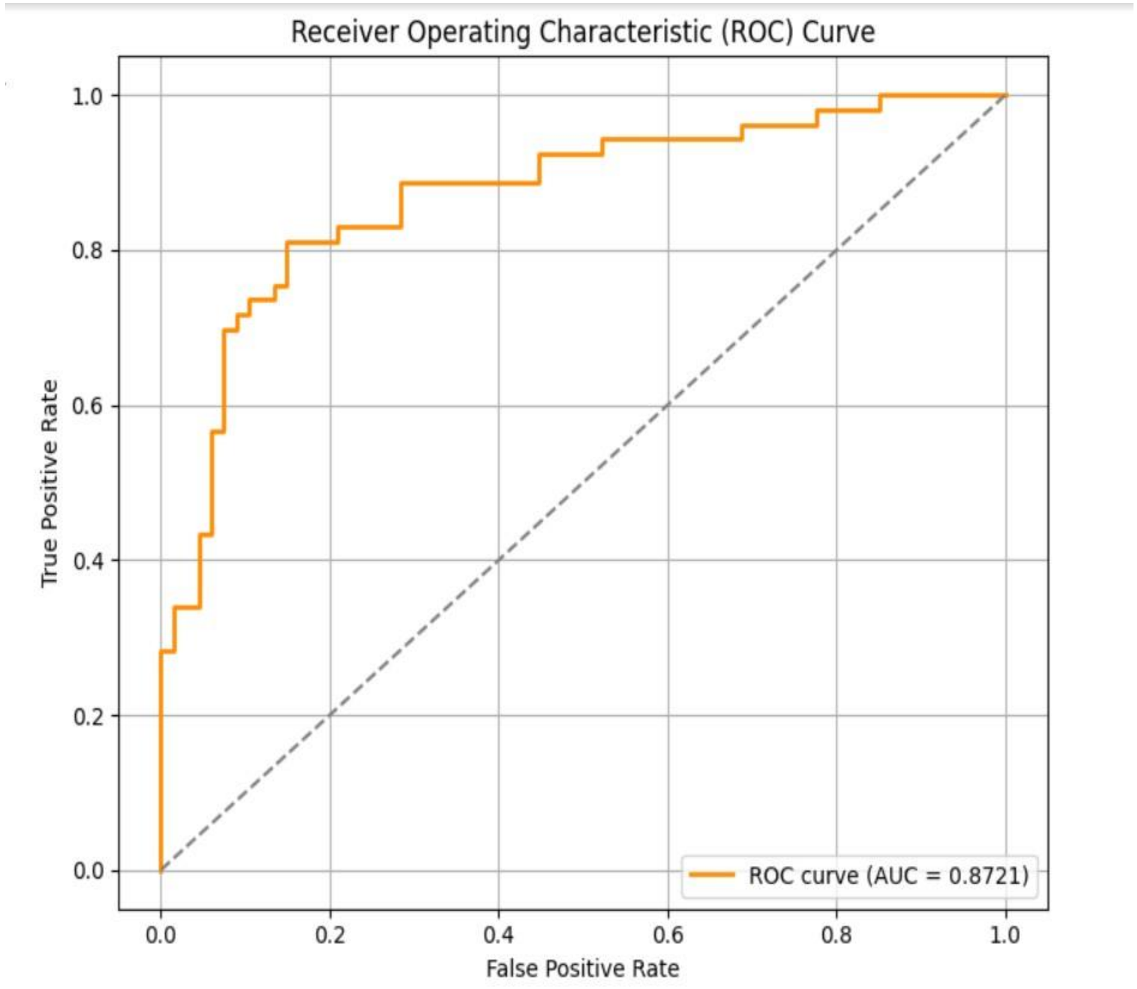
```
Parameter Comparison:
  alpha_r:
    ↪ Previous: 0.84
    ↪ New Best: 0.8199844782456824 ✅ CHANGED
  beta:
    ↪ Previous: 0.15
    ↪ New Best: 0.23515562871287 ✅ CHANGED
  gamma_r:
    ↪ Previous: 1.0
    ↪ New Best: 1.4612961670003108 ✅ CHANGED
  gamma_c:
    ↪ Previous: 2.0
    ↪ New Best: 3.1038963555252 ✅ CHANGED
  alpha_c:
    ↪ Previous: 0.25
    ↪ New Best: 0.5518748596679164 ✅ CHANGED
```

* From 5% Sample, 10 epochs, 10 trials

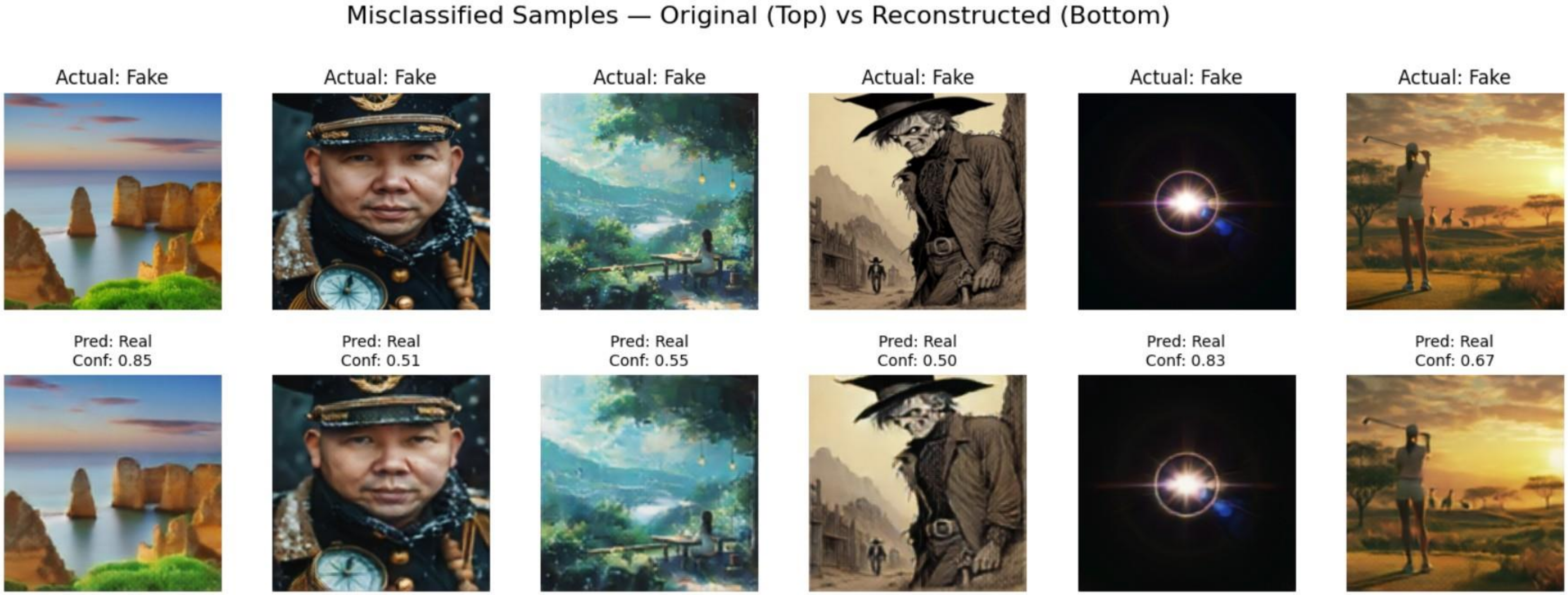
(3) FOCAL LOSS W/ OPTUNA (3 HOURS)



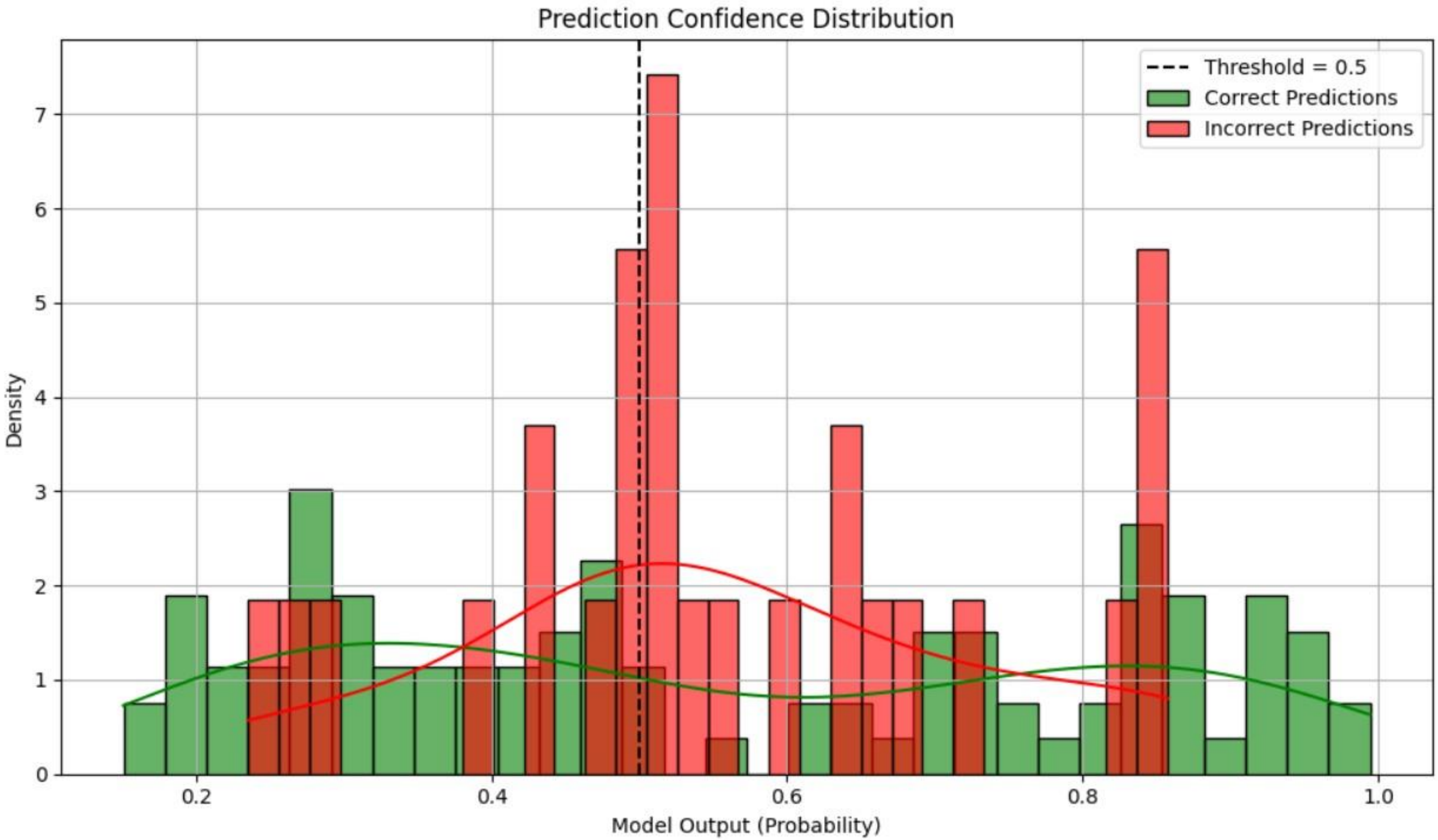
```
  warnings.warn(
Classification Report on Test Set
Accuracy:  0.7833
Precision: 0.7213
Recall:    0.8302
F1 Score:  0.7719
```



(3) FOCAL LOSS W/ OPTUNA



(3) HYBRID/FOCAL LOSS W/ OPTUNA



(4) HYBRID LOSS W/ OPTUNA & YOUDEN

Model Architecture

- Transferred Encoder: Unfrozen layers on training
- Layered Decoder: Batch Normalization
- Fully Connected Classifier: Dropout Layers & Global Average Pooling

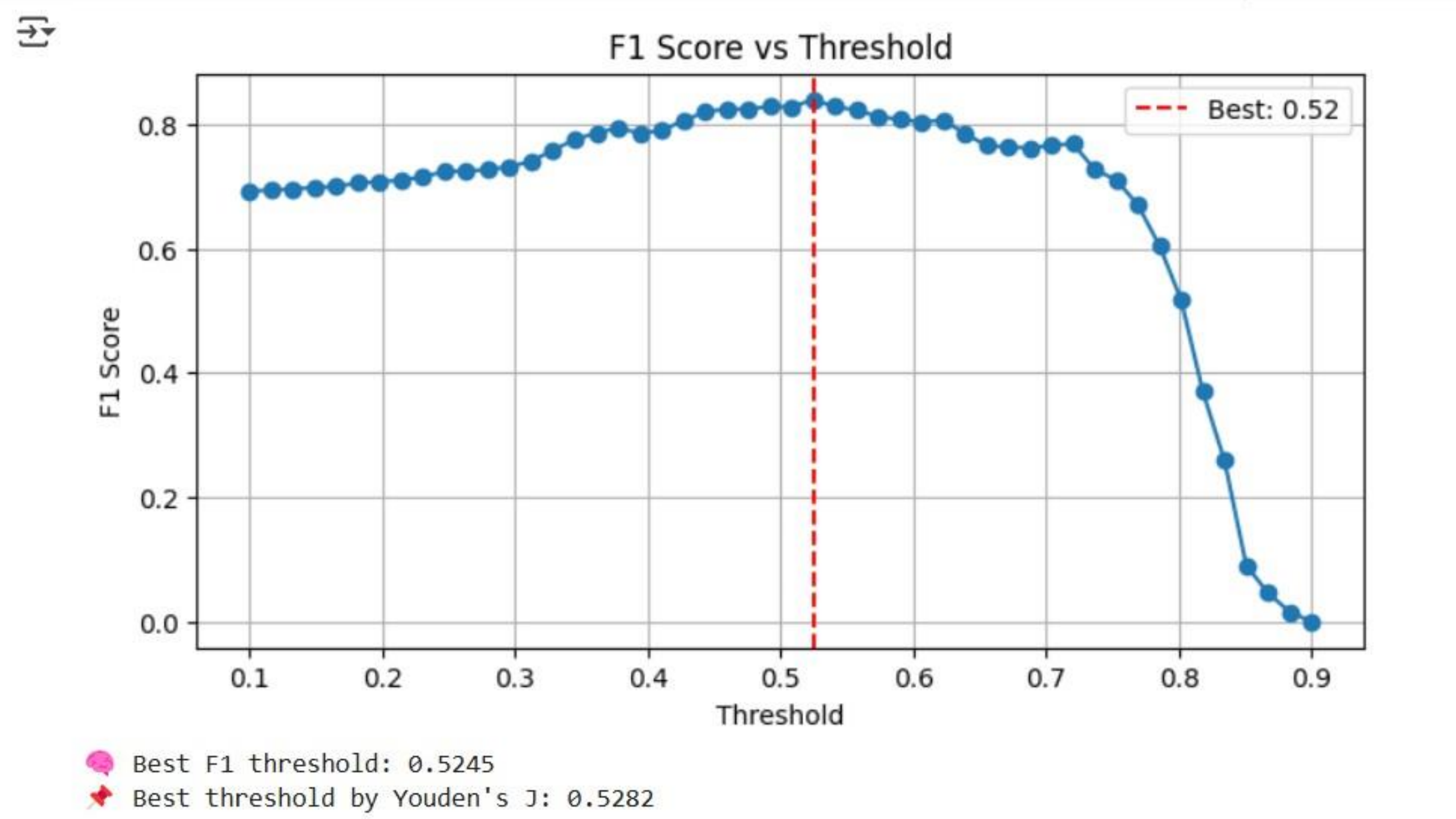
Loss Function

- Reconstruction Loss: Hybrid Loss
- Classification Loss: Focal Loss
- Total Loss = recon loss * r_weight + class loss * c_weight (Progressive weighting and freeze epochs)

Model Additions

- Unfreeze final 3 layers of transfer encoder during training
- Youden Threshold
- Trained on 10%, 50% and 100% sample, 100 epochs
- Save Checkpoints

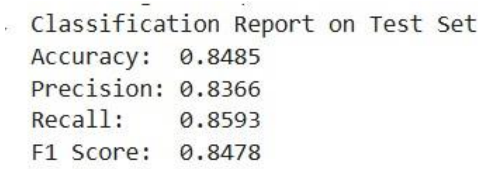
(4) HYBRID LOSS W/ OPTUNA & YOUDEN



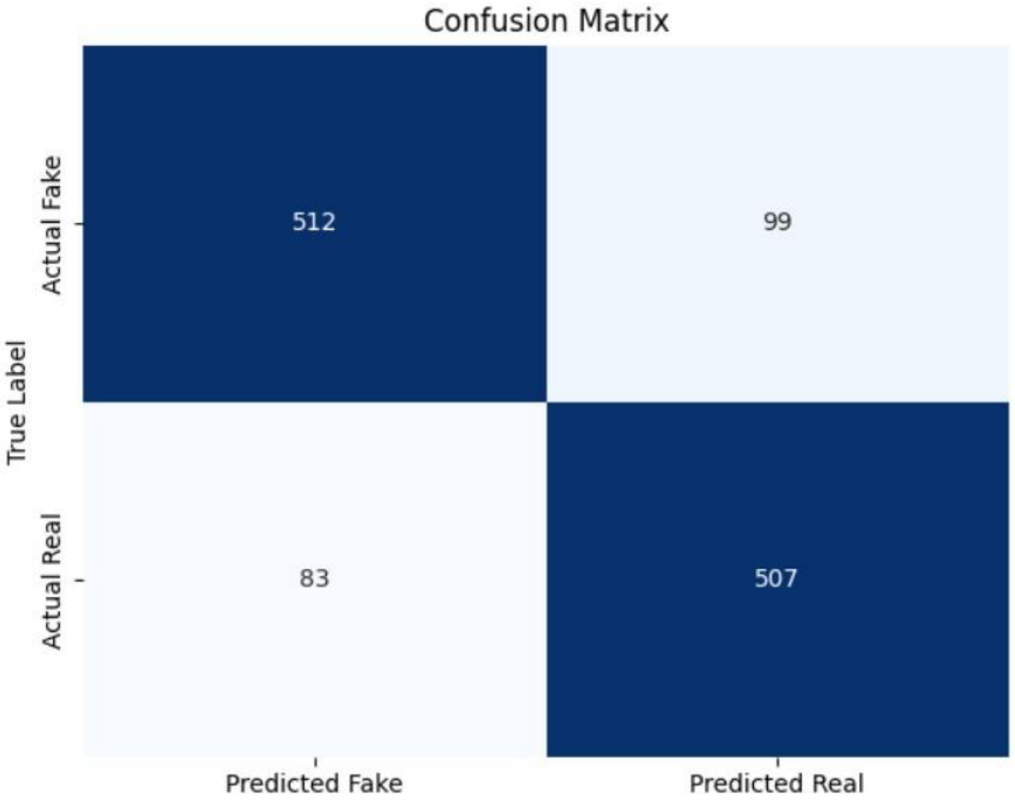
```
🧠 Best F1 threshold: 0.5245
📌 Best threshold by Youden's J: 0.5282
```

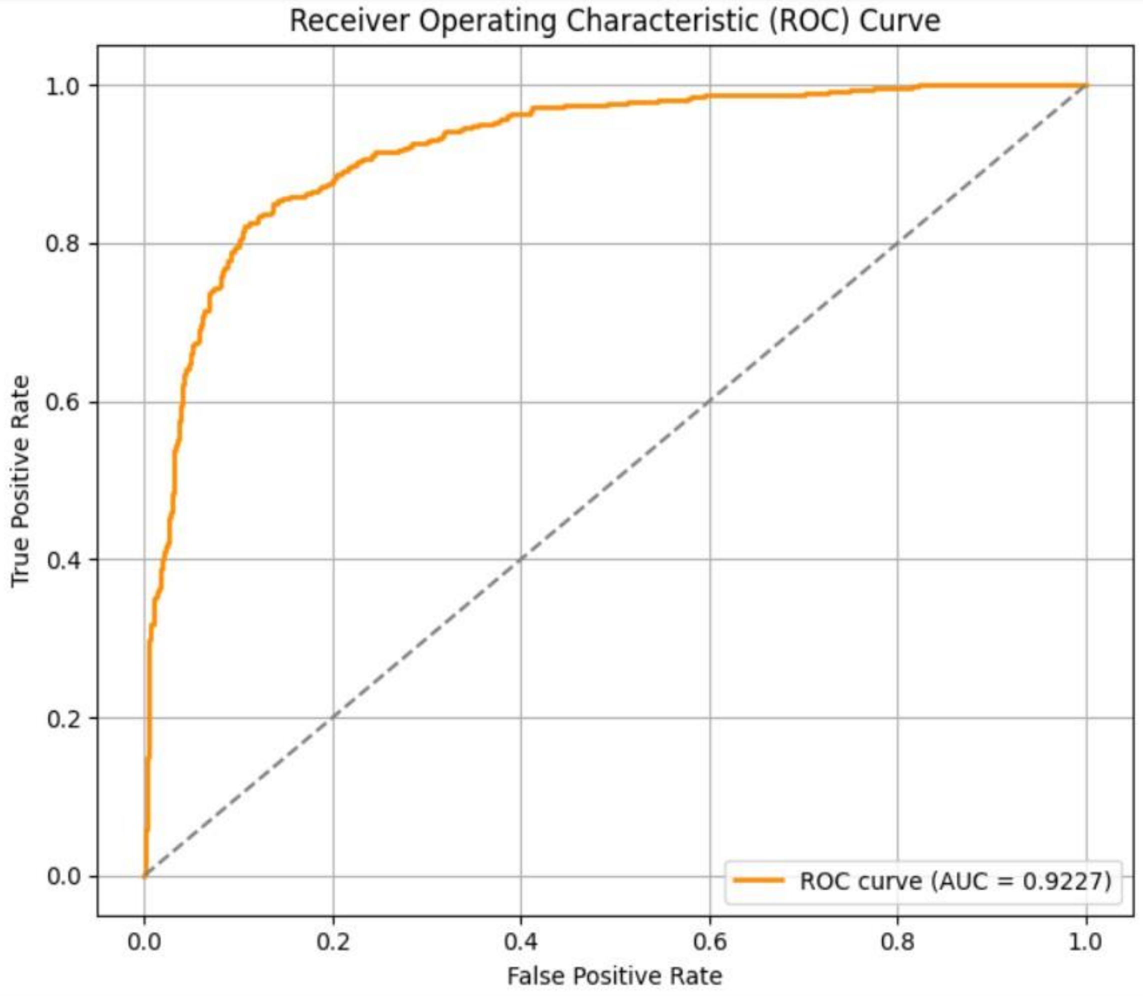
* From 10% Sample

(4) HYBRID LOSS W/ OPTUNA & YOUDEN (12 HOURS)

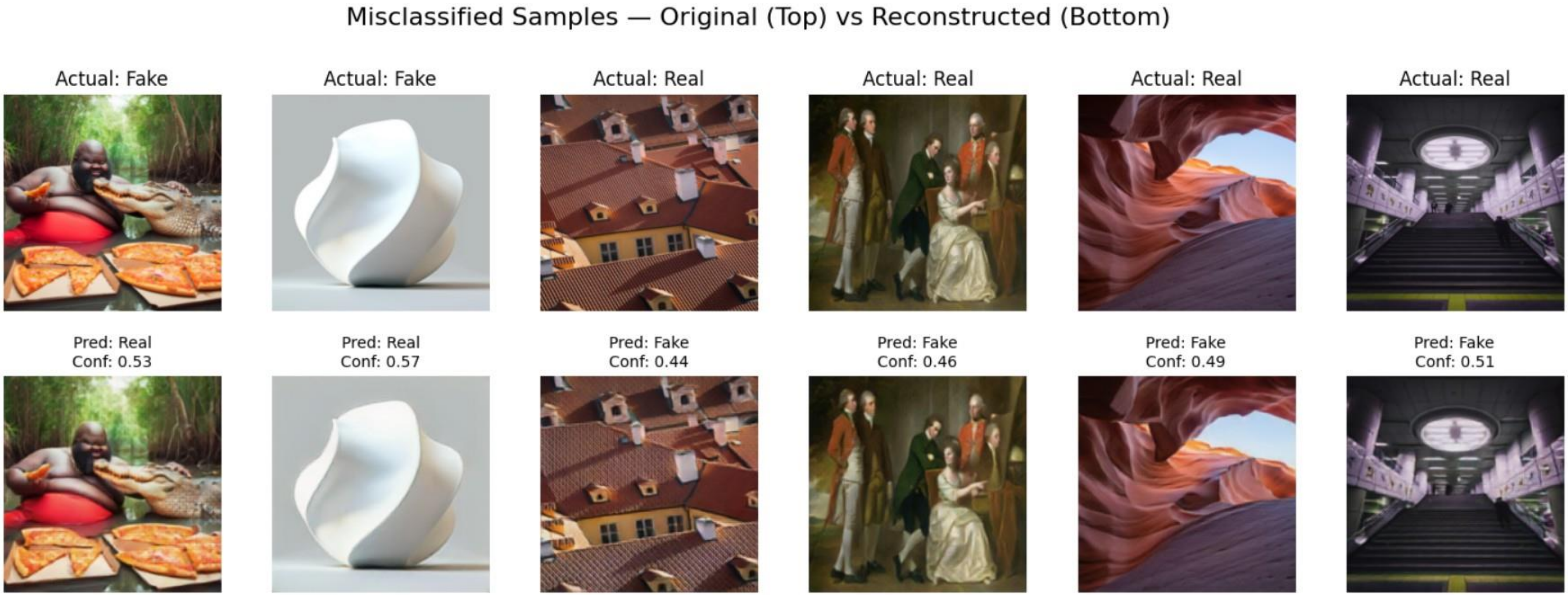


```
Classification Report on Test Set
Accuracy:  0.8485
Precision: 0.8366
Recall:    0.8593
F1 Score:  0.8478
```



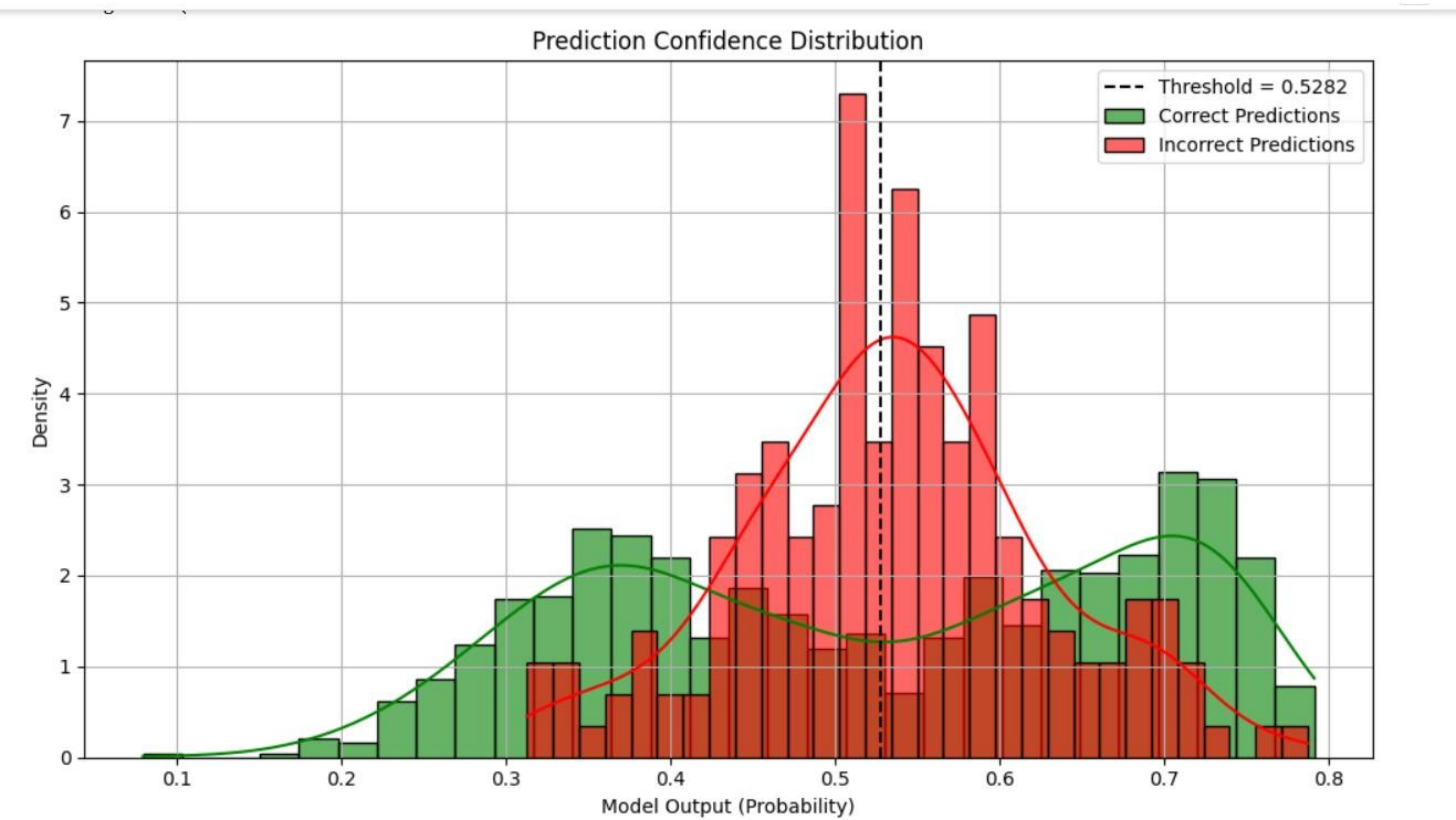


(4) HYBRID LOSS W/ OPTUNA & YOUDEN

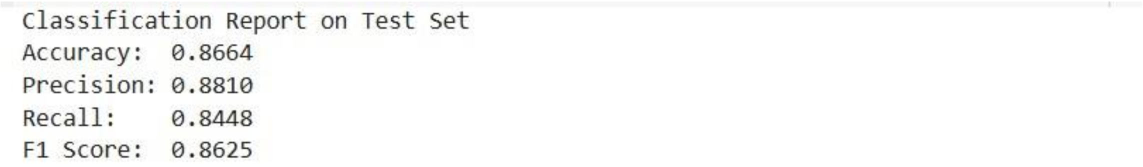


Misclassified Samples — Original (Top) vs Reconstructed (Bottom)

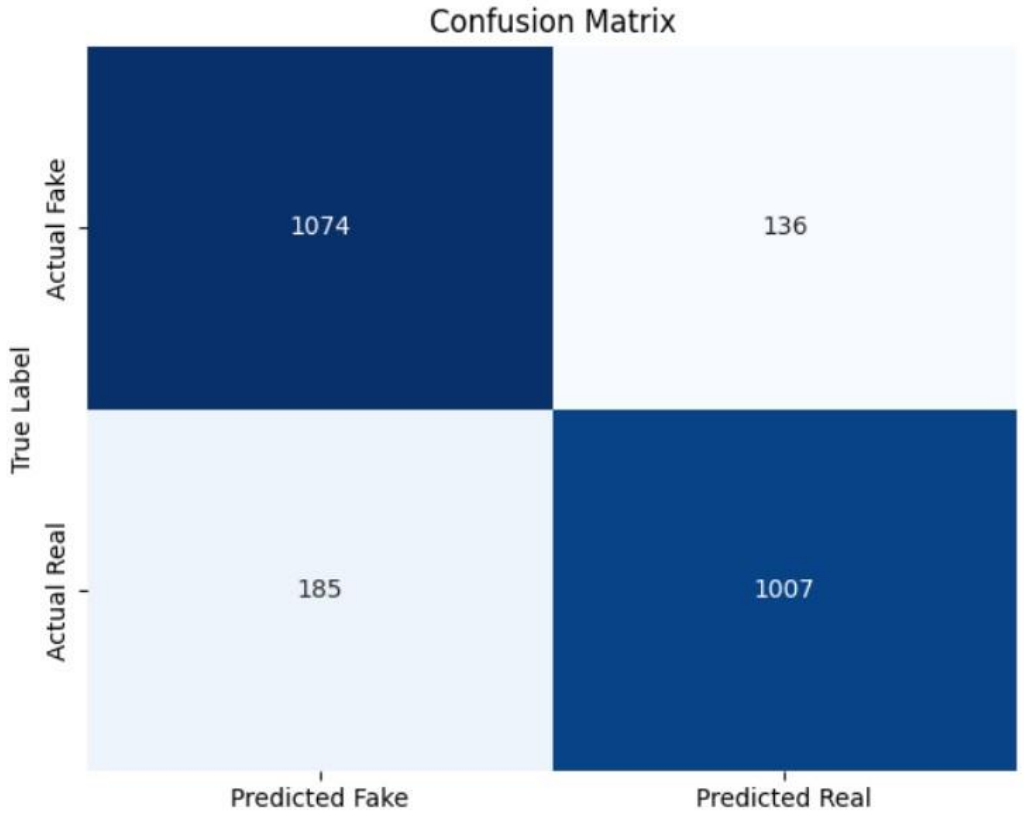
(4) HYBRID LOSS W/ OPTUNA & YOUDEN

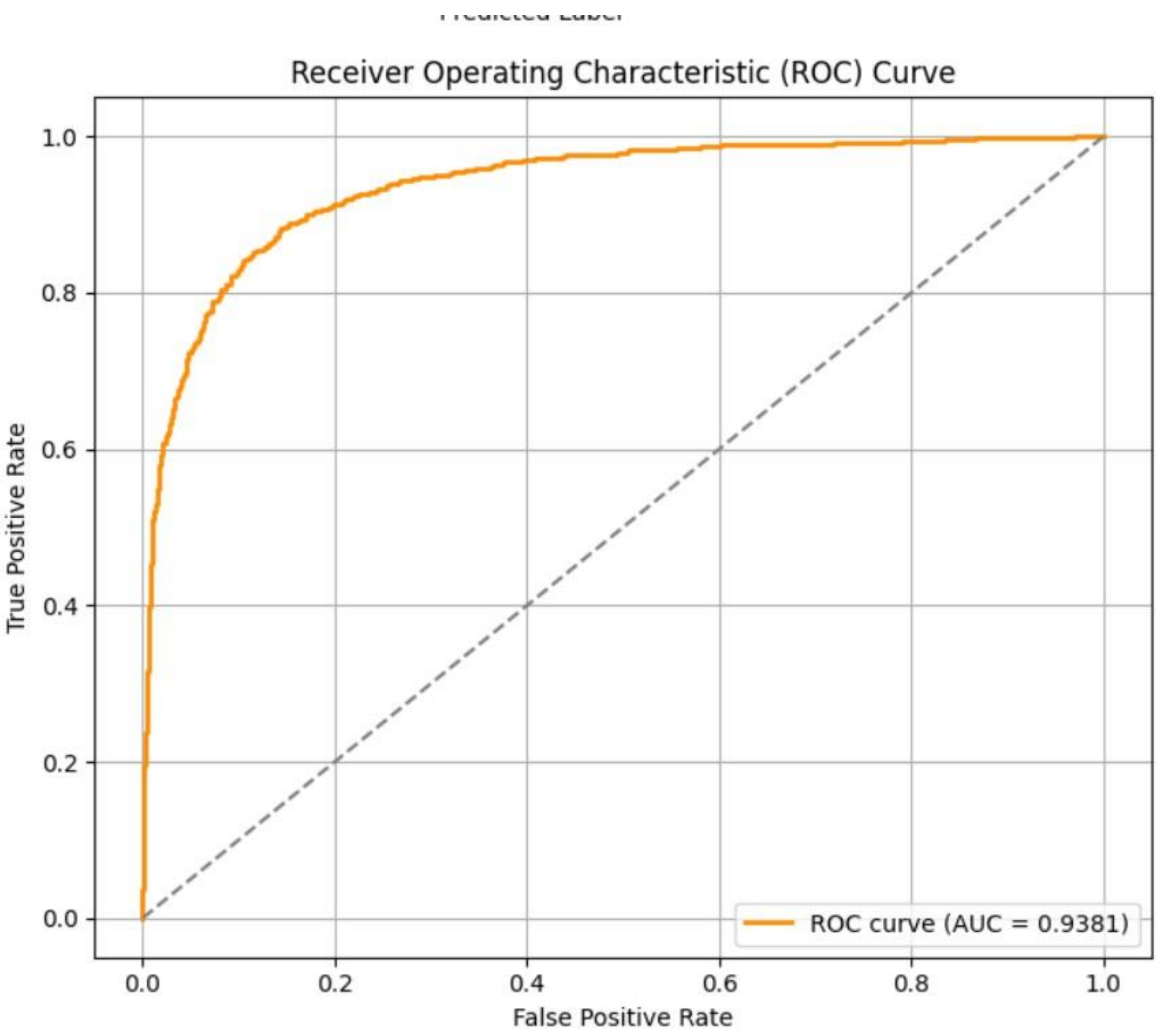


(5) BEST MODEL FULL TRAINING (24 HOURS)

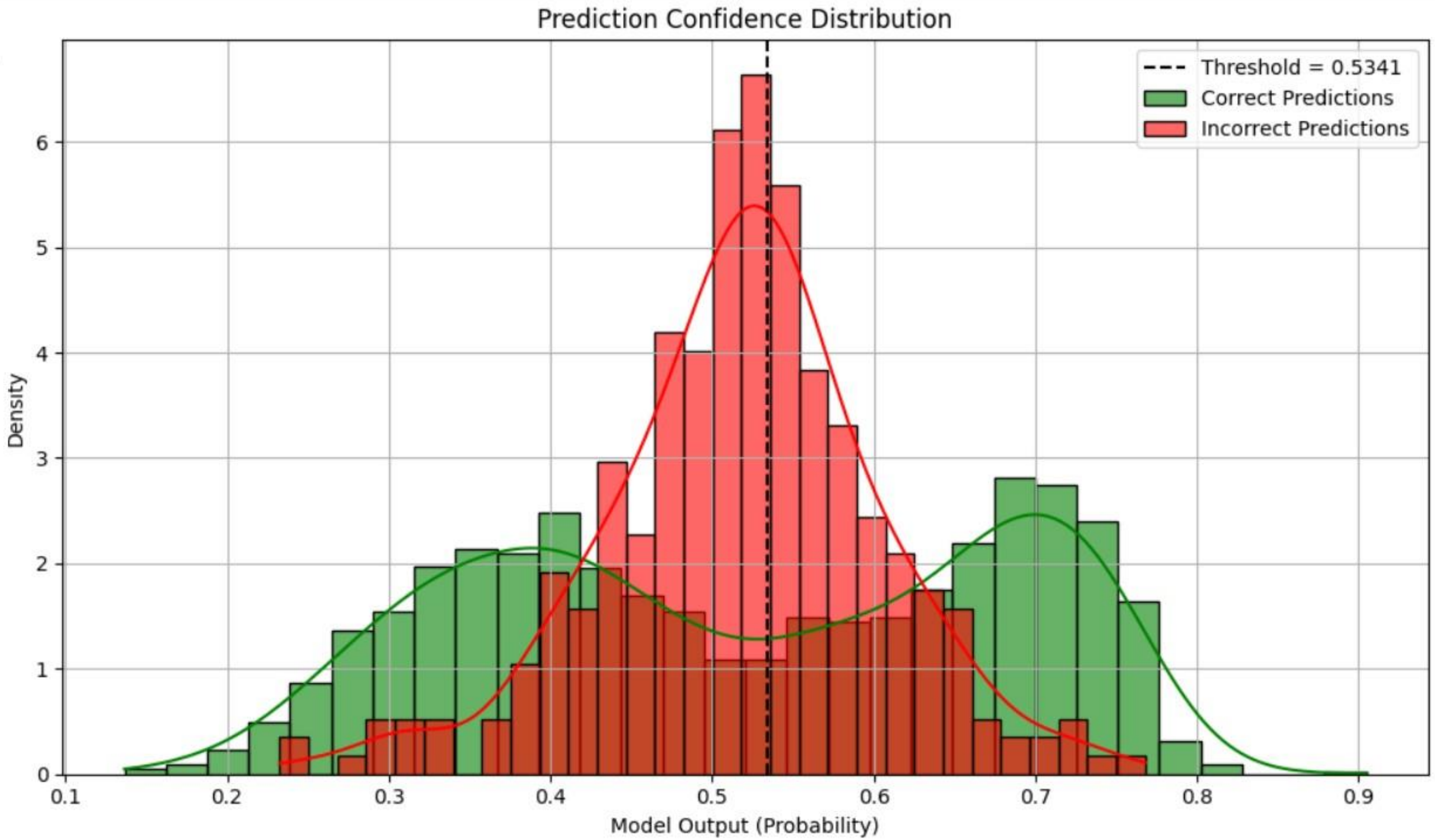


```
Classification Report on Test Set
Accuracy:  0.8664
Precision: 0.8810
Recall:    0.8448
F1 Score:  0.8625
```





(5) BEST MODEL FULL TRAINING



GREYSCALE GENERAL MODEL SUMMARY

Model Architecture

- Transferred Encoder: Unfrozen layers on training
- Layered Decoder: Batch Normalization
- Fully Connected Classifier: Dropout Layers & Global Average Pooling

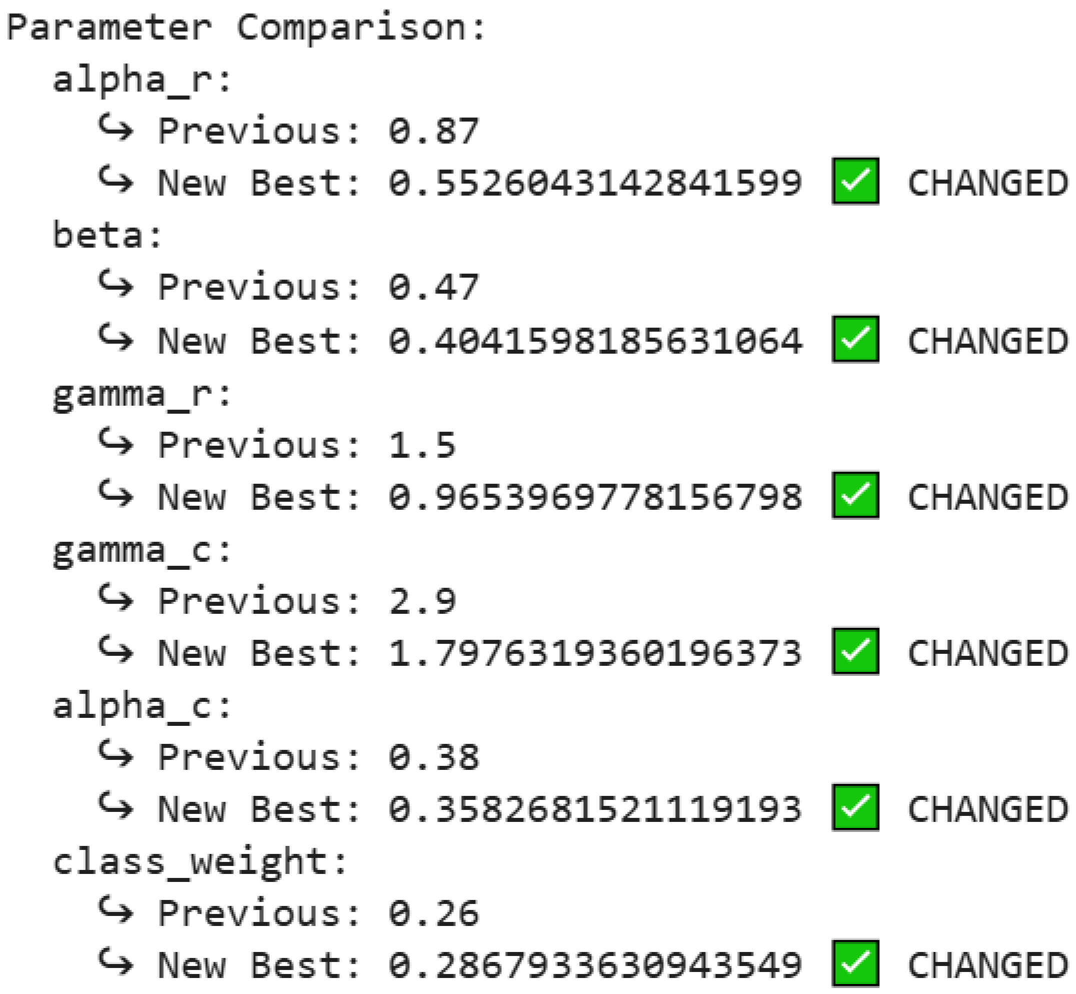
Loss Function

- Hybrid Loss w/ Optuna & Youden
- Reconstruction Loss: Hybrid Loss
- Classification Loss: Focal Loss
- Total Loss = recon loss * r_weight + class loss * c_weight (Progressive weighting and freeze epochs)

Model Features

- Grey-Scaled Images
- Unfreeze final 3 layers of transfer encoder during training
- Youden Threshold
- Trained on 5%, 10% and 100% sample, 100 epochs
- Save Checkpoints

OPTUNA

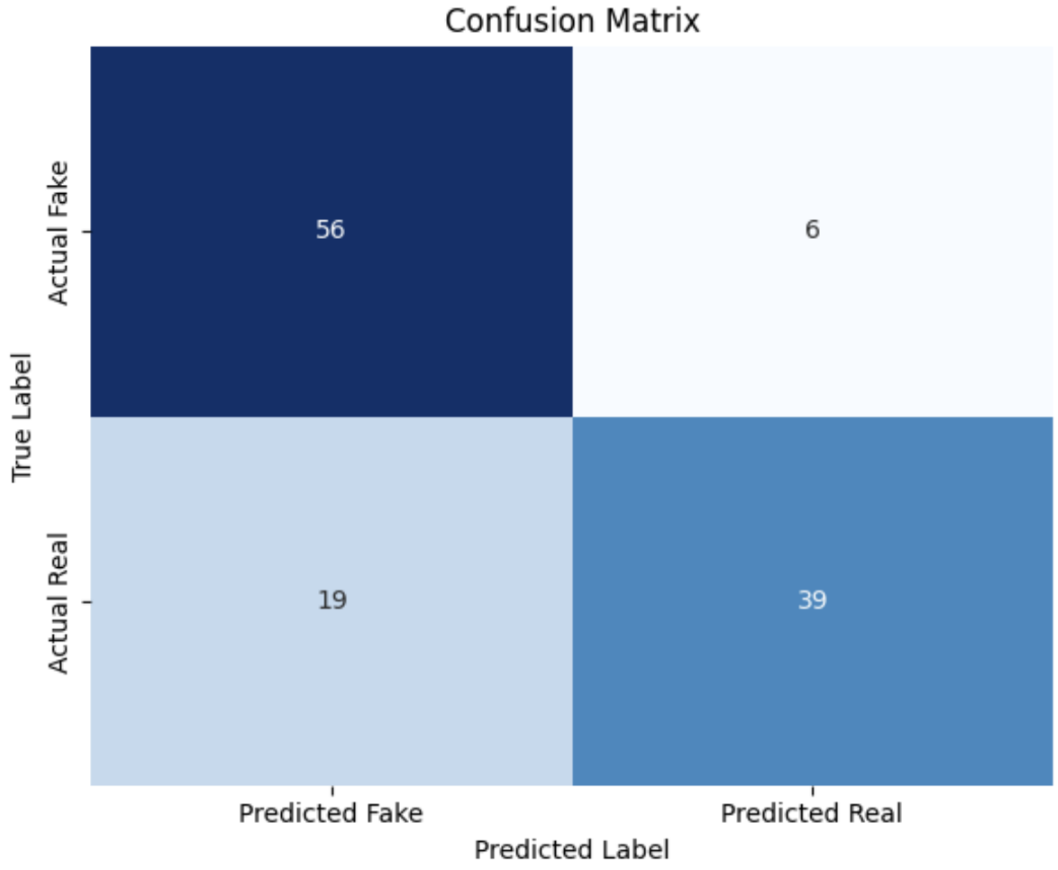


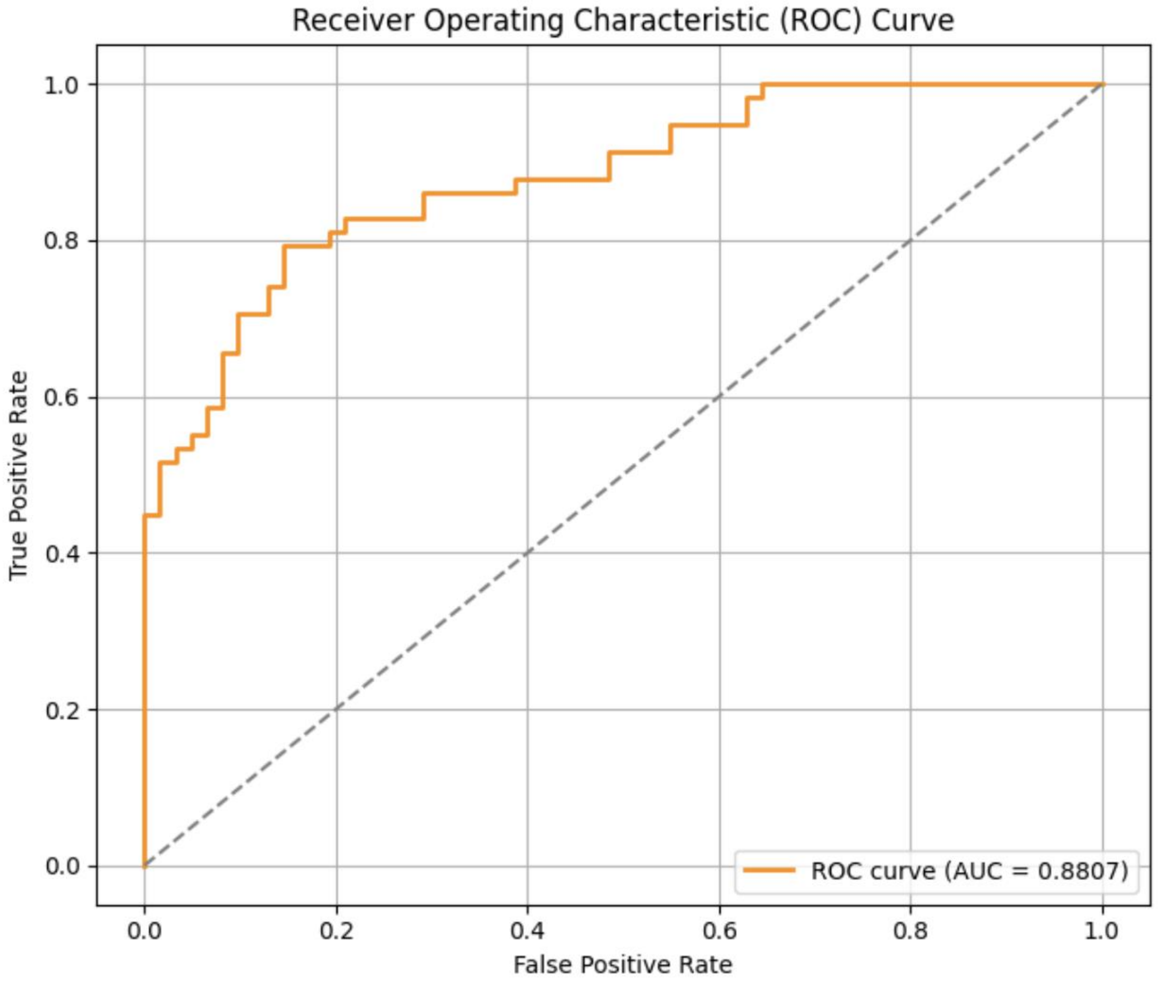
```
Parameter Comparison:
  alpha_r:
    ↪ Previous: 0.87
    ↪ New Best: 0.5526043142841599 ✅ CHANGED
  beta:
    ↪ Previous: 0.47
    ↪ New Best: 0.4041598185631064 ✅ CHANGED
  gamma_r:
    ↪ Previous: 1.5
    ↪ New Best: 0.9653969778156798 ✅ CHANGED
  gamma_c:
    ↪ Previous: 2.9
    ↪ New Best: 1.7976319360196373 ✅ CHANGED
  alpha_c:
    ↪ Previous: 0.38
    ↪ New Best: 0.3582681521119193 ✅ CHANGED
  class_weight:
    ↪ Previous: 0.26
    ↪ New Best: 0.2867933630943549 ✅ CHANGED
```

* From 5% Sample, 10 epochs, 10 trials

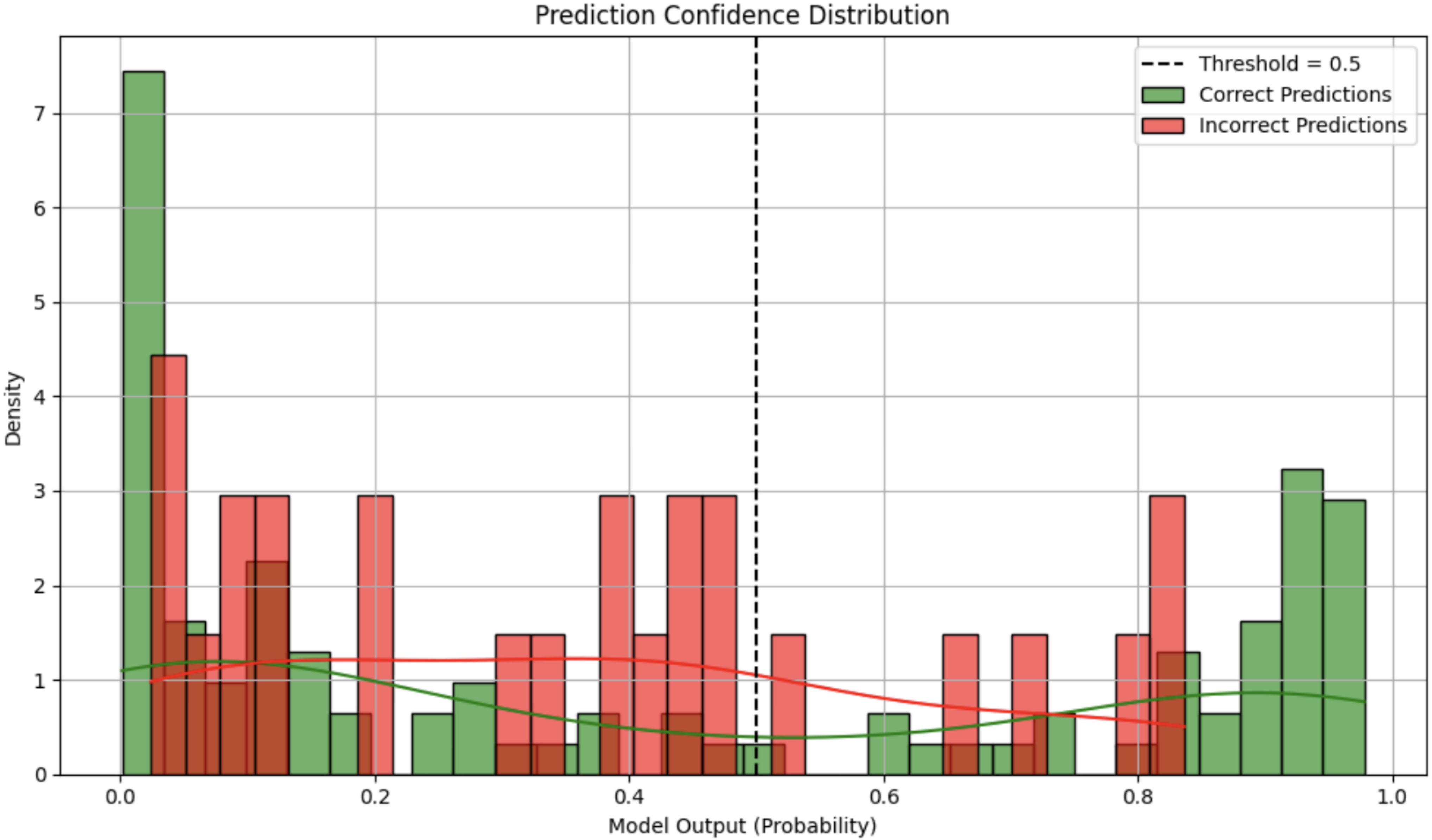
GREYSCALED (5% - 2 HRS)

```
Classification Report on Test Set
Accuracy:  0.7917
Precision: 0.8667
Recall:    0.6724
F1 Score:  0.7573
```

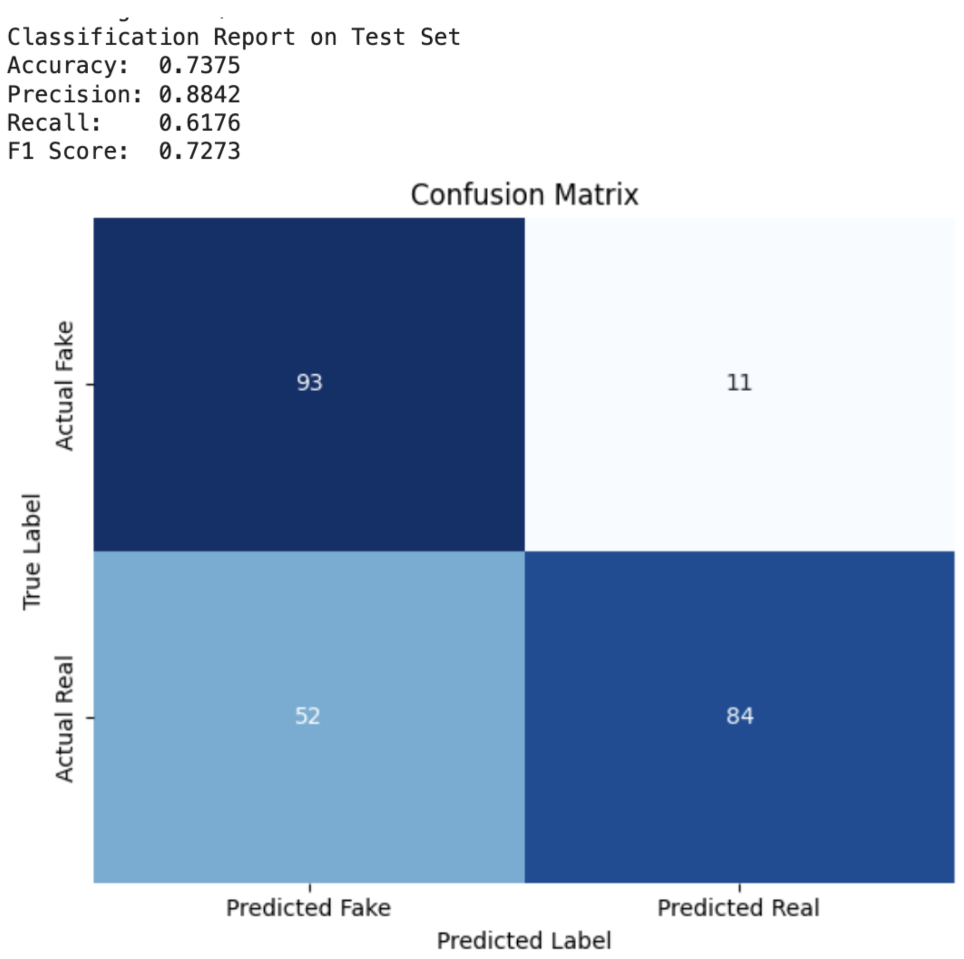




5% TRAINING (2 HOURS)



GREYSCALED (10% - 4 HRS)

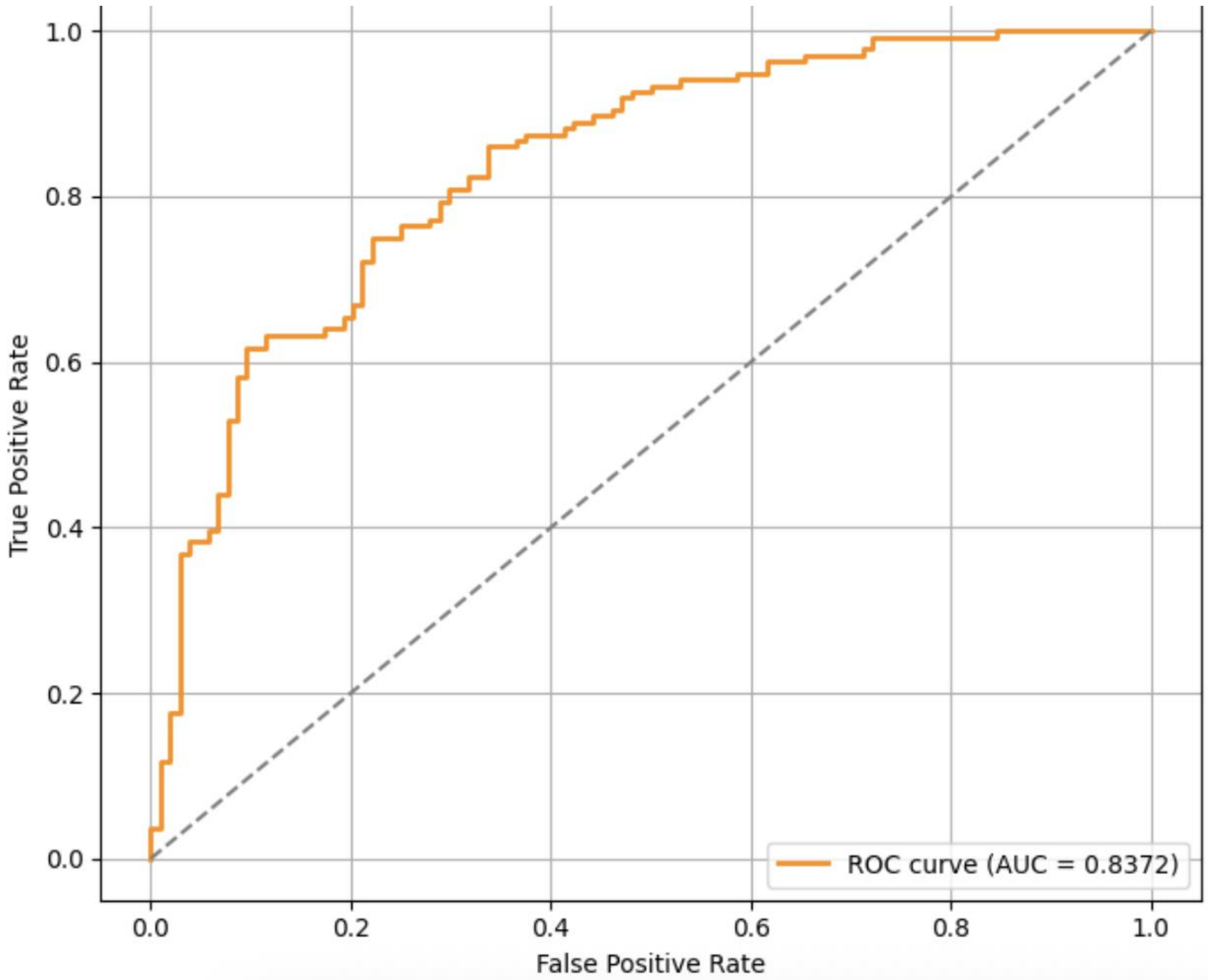


```
Classification Report on Test Set
Accuracy:  0.7375
Precision: 0.8842
Recall:    0.6176
F1 Score:  0.7273
```

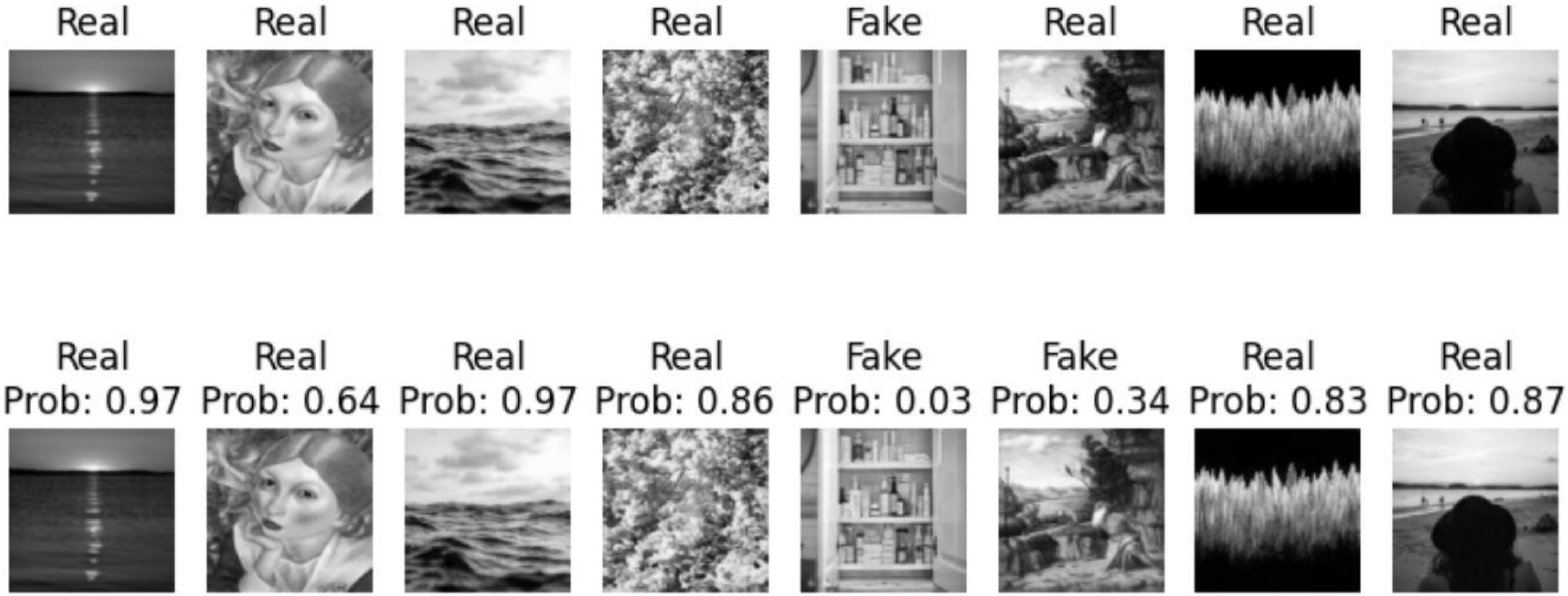
Confusion Matrix

True Label	Predicted Fake	Predicted Real
Actual Fake	93	11
Actual Real	52	84

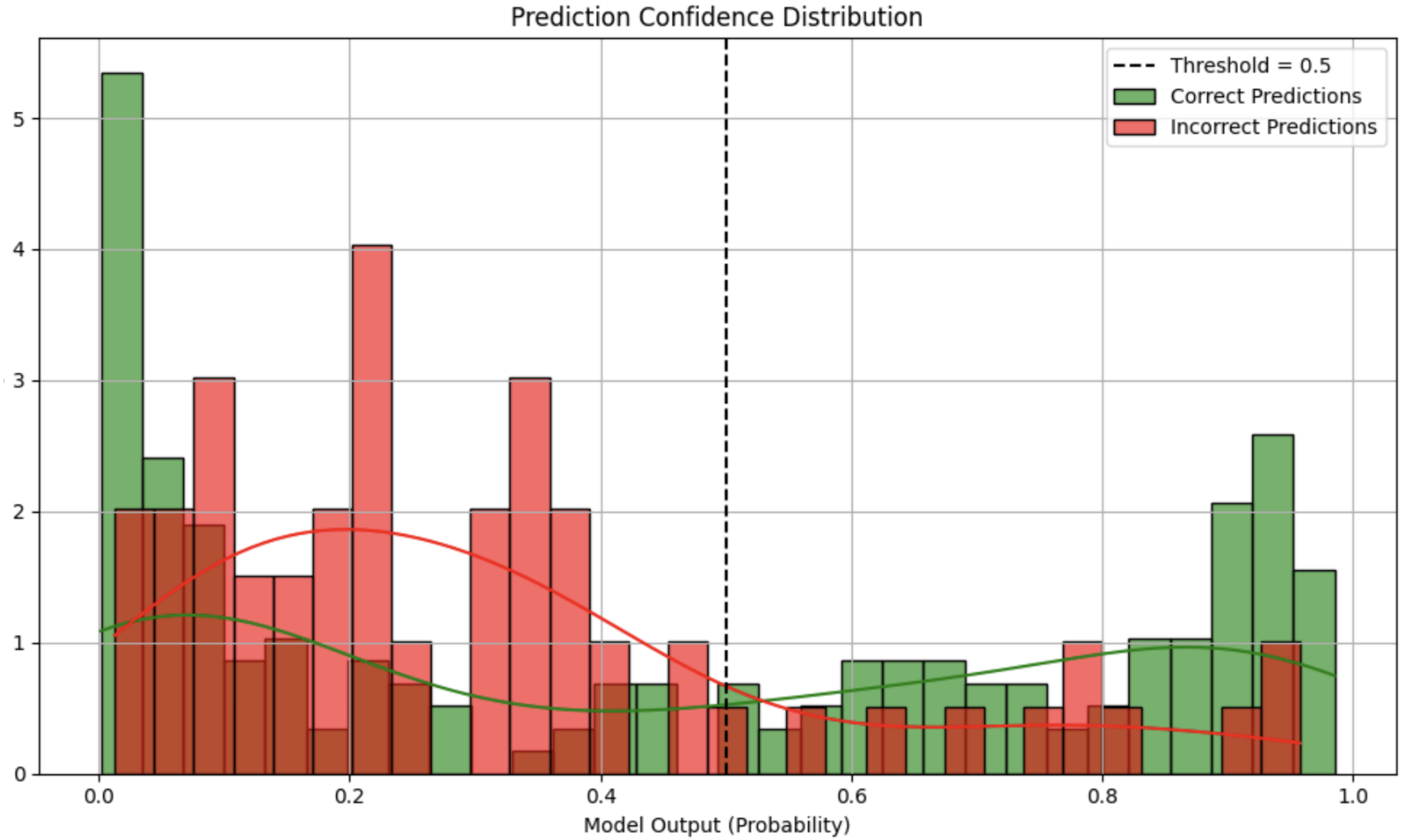
Predicted Label



GREYSCALED (10% - 4 HRS)



10% TRAINING (4 HOURS)



MODEL RESULTS/RUNTIMES

Sample Size	AE - Model 1	AE - Model 2	AE - Model 3	AE - Model 4	Grey Scaled
5% Sample	58% / 15m (1%)	75% / 1.5 h	78 % / 3 h	x	79% / 2 h
10% Sample	x	x	x	75 % / 5h	74% / 4 h
50% Sample	x	x	x	84.9 % / 12h	x
100% Sample	X	x	X	86.6 % / 24h	19h – accuracy not recorded

*Last semester best results with 100% sample: 83% / 6h (CNN – Pytorch)

OPPORTUNITIES FOR IMPROVEMENT

Image Distribution

Use an unbalanced dataset to prepare for real life applications

Greyscale Images

Test the RGB model with Greyscale images

Use Full Kaggle Set

Train on the entire Kaggle dataset. Our "full" training run is still only 20% of available images

Pre-Classify Images

Classify images into classes like dog, cat, face, etc. Then build one AE for each class. Multiple AEs for each class may work better

CONCLUSIONS

THANK YOU