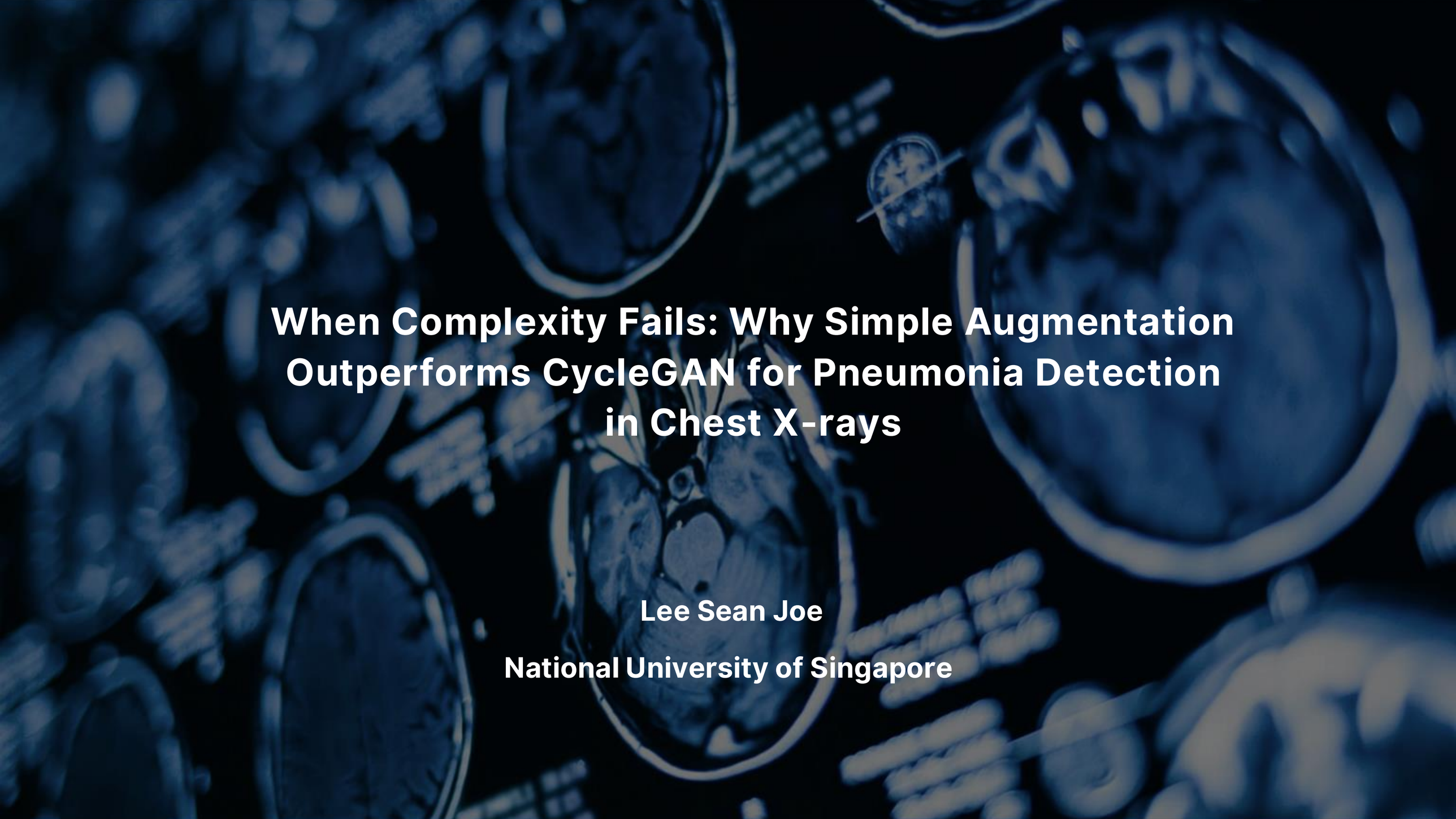


When Complexity Fails: Why Simple Augmentation Outperforms CycleGAN for Pneumonia Detection in Chest X-rays

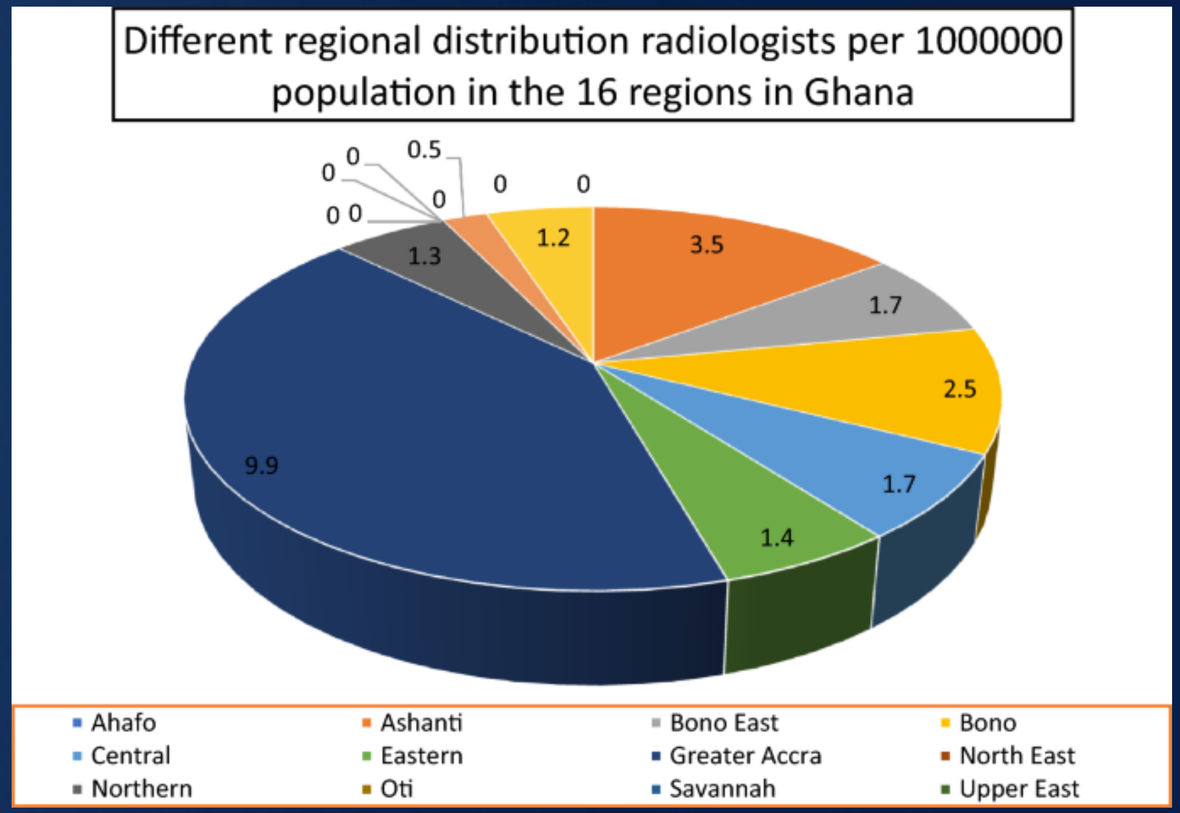
Lee Sean Joe

National University of Singapore

Introduction/Background

- Each year, pneumonia affects about 450 million people globally (7% of the population) and results in about 4 million deaths

- Most developing countries only have **1.2 to 3.9 radiologists per 100,000 people**, below the recommended target of 8 per 100,000 set by the Royal College of Radiologists.



Introduction/Background

- In current clinical practice, the **overwhelming proportion of medical imaging for chest X-rays uses simple methods as described.**
  - Simple: Geometric & intensity transforms
  - Complex: GANs generate synthetic pathology

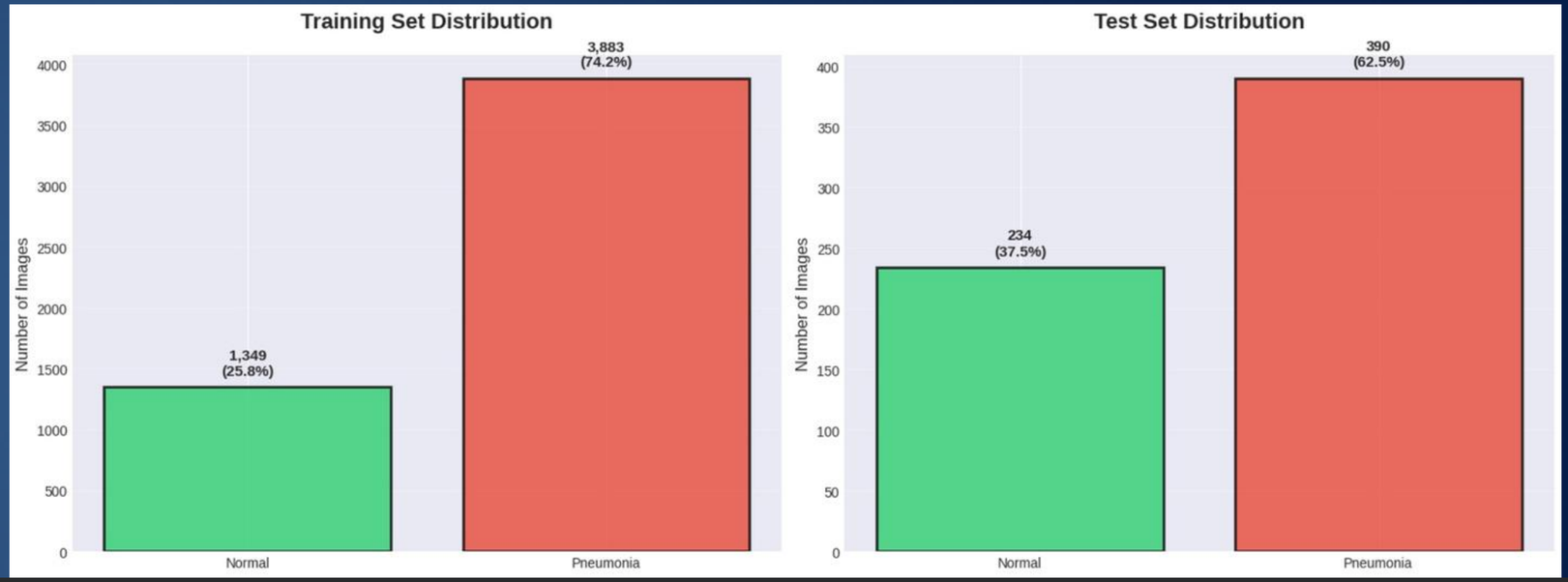
- The use of GANs to generate synthetic pathology is estimated to be **near 0% in actual, deployed clinical diagnostic pipelines**.
  - Challenges in ensuring the generated images do not introduce spurious artifacts or subtle errors that could mislead a diagnosis.

For this work, we aim to test if more sophisticated methods can give us better performance.

Research Questions

- How do complex methods such as CycleGAN compare to popular (and simpler) pretrained models?
- How can we employ/integrate these complex methods into anomaly detection tasks?
- How does synthetic data generation affect evaluation?
- Where do these generative models focus when creating synthetic pathology?

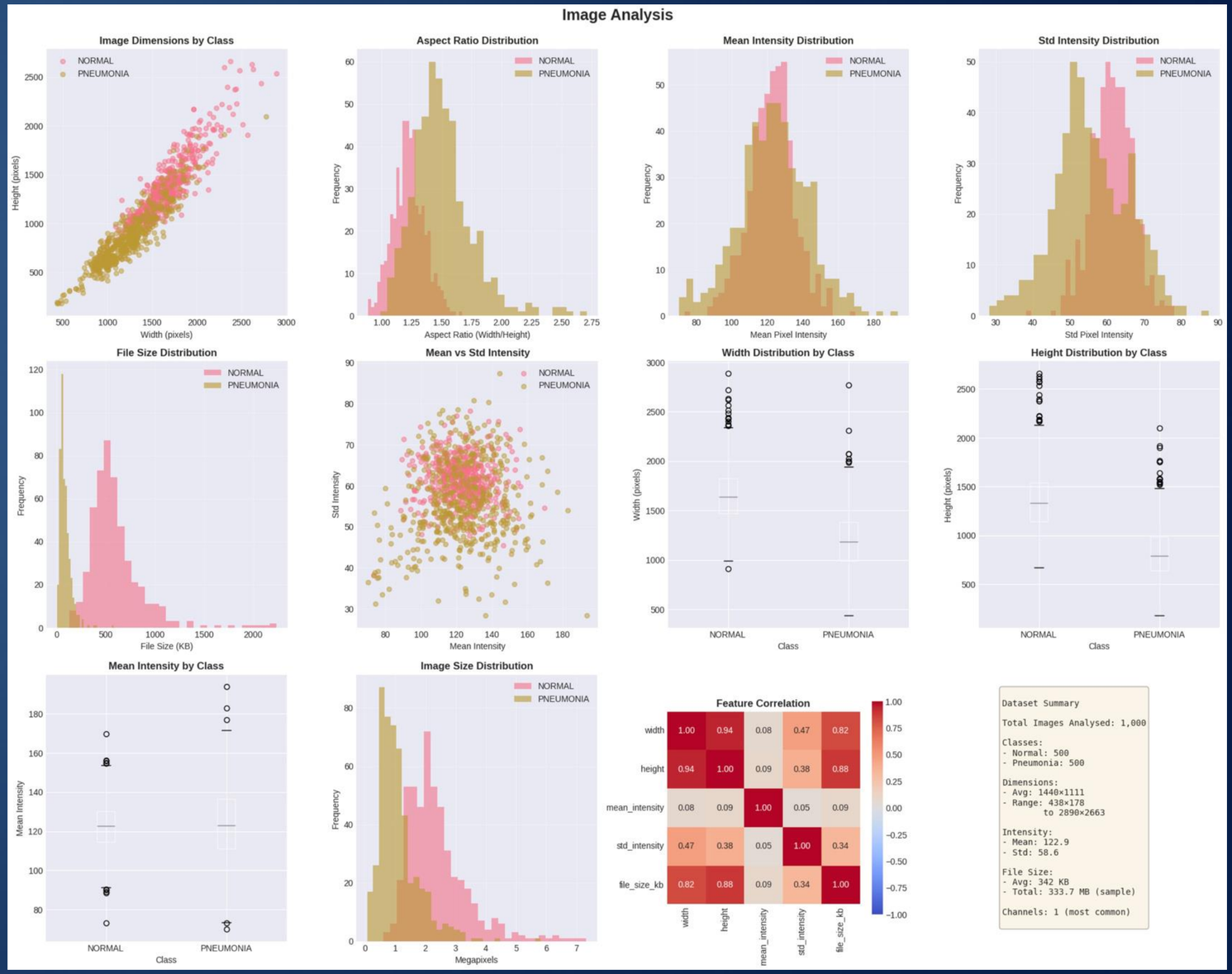
Exploratory Data Analysis



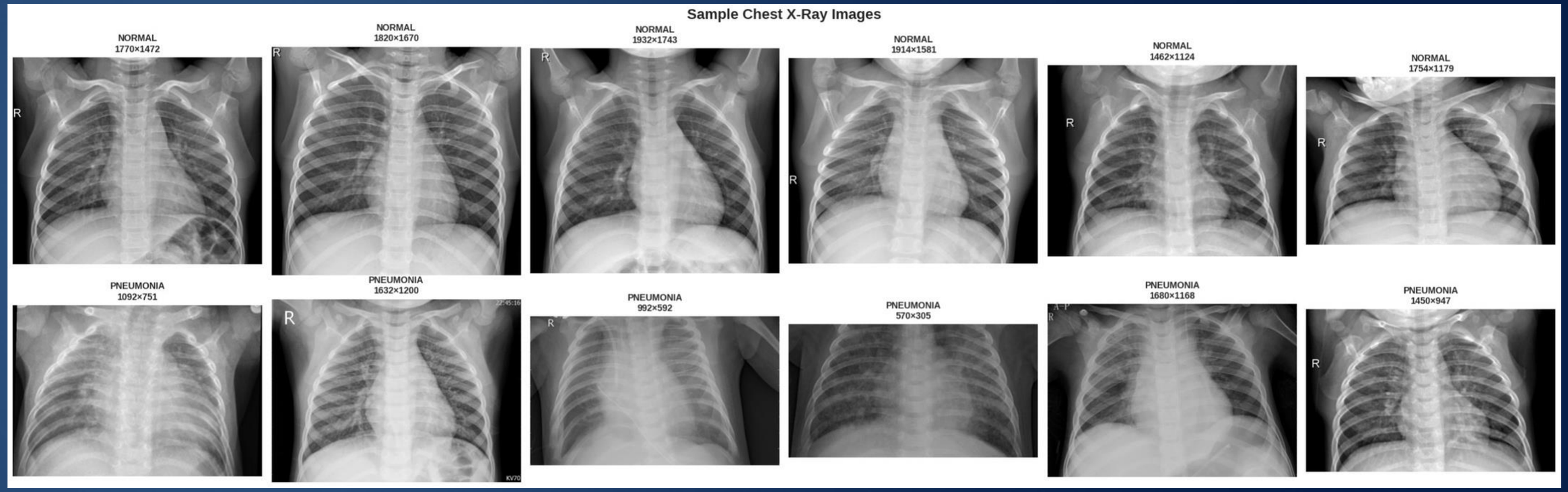
NORMAL:	
Count:	500 images analysed
Avg Dimensions:	1665 x 1381 pixels
Min Dimensions:	912 x 672 pixels
Max Dimensions:	2890 x 2663 pixels
Aspect Ratio:	1.227 (avg)
Channels:	1 (most common)
Mean Intensity:	122.49 ± 12.92
Std Intensity:	61.40 ± 5.55
File Size:	596.0 KB (avg)

PNEUMONIA:	
Count:	500 images analysed
Avg Dimensions:	1215 x 840 pixels
Min Dimensions:	438 x 178 pixels
Max Dimensions:	2772 x 2098 pixels
Aspect Ratio:	1.507 (avg)
Channels:	1 (most common)
Mean Intensity:	123.31 ± 19.61
Std Intensity:	55.86 ± 9.98
File Size:	87.4 KB (avg)

Exploratory Data Analysis

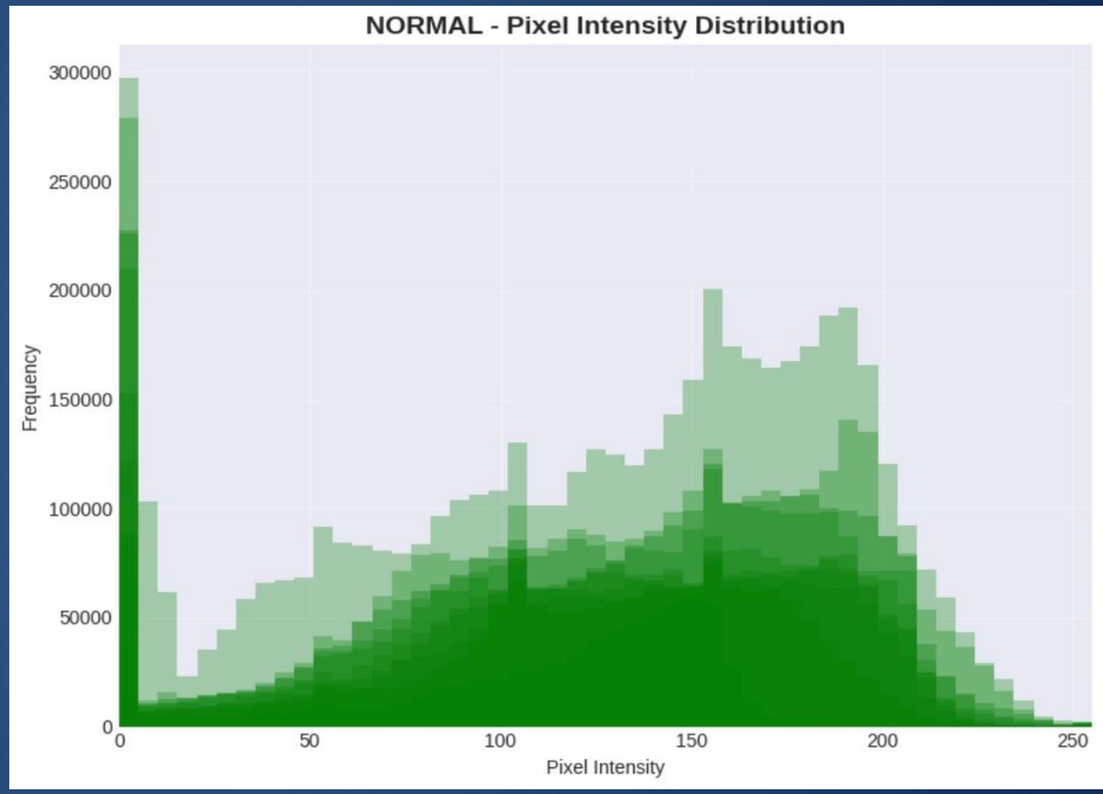


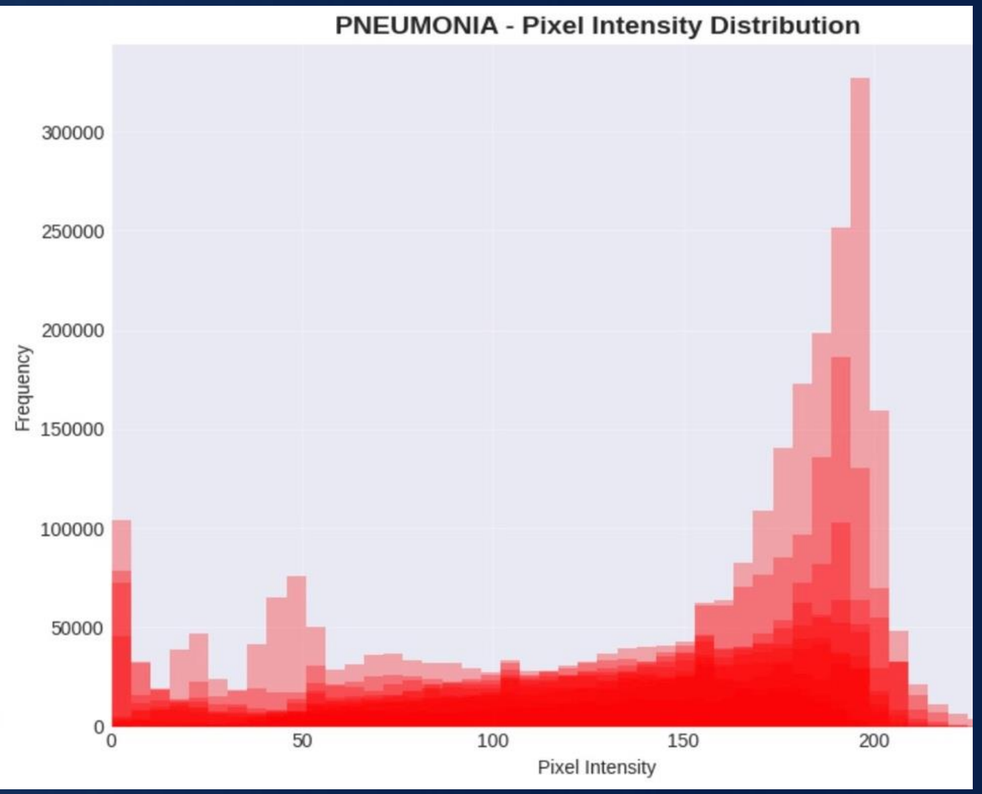
Exploratory Data Analysis



- Raw Normal vs Pneumonia X-ray images from the dataset

Exploratory Data Analysis





- **Normal X-rays (Green)**
- **Spike at 0**: ~280k pixels - black background outside body
- **Main peak**: 150-170 intensity - normal lung tissue
- **Distribution**: Bell-shaped, centred around 150
- **Range**: Mostly 50-200, relatively narrow
- **Interpretation**: Consistent, air-filled healthy lungs

- **Pneumonia X-rays (Red)**
- **Spike at 0**: ~300k pixels - black background (similar)
- **Main peak**: 200-220 intensity - fluid/consolidation
- **Distribution**: Bimodal, broader spread
- **Range**: 50-250, shifted right and wider
- **Interpretation**: Mix of normal lung + bright pneumonia opacities

Data Preprocessing

CLAHE:

- To understand how this works, we first have to understand the building blocks:
  - Histogram Equalization
  - Adaptive HE

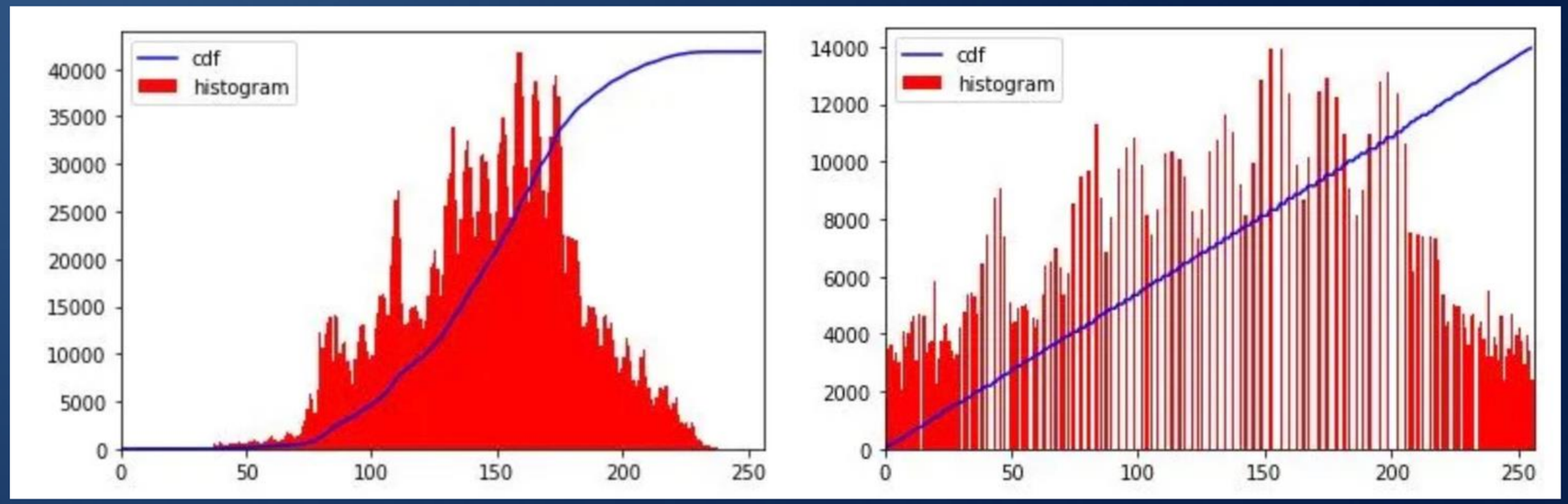
Data Preprocessing

Histogram Equalization:

How it works:

1. Calculate global histogram for entire image
2. Build cumulative distribution function (CDF)
3. Map: new_pixel = (CDF[old_pixel] / total_pixels) × 255
4. Apply same mapping to every pixel

The pixel intensity values now range from 0 to 255 on the x-axis. The original histogram has been stretched, and the cumulative distribution function (CDF) line is now linear as opposed to the original curved line.



Data Preprocessing

Issues with HE:

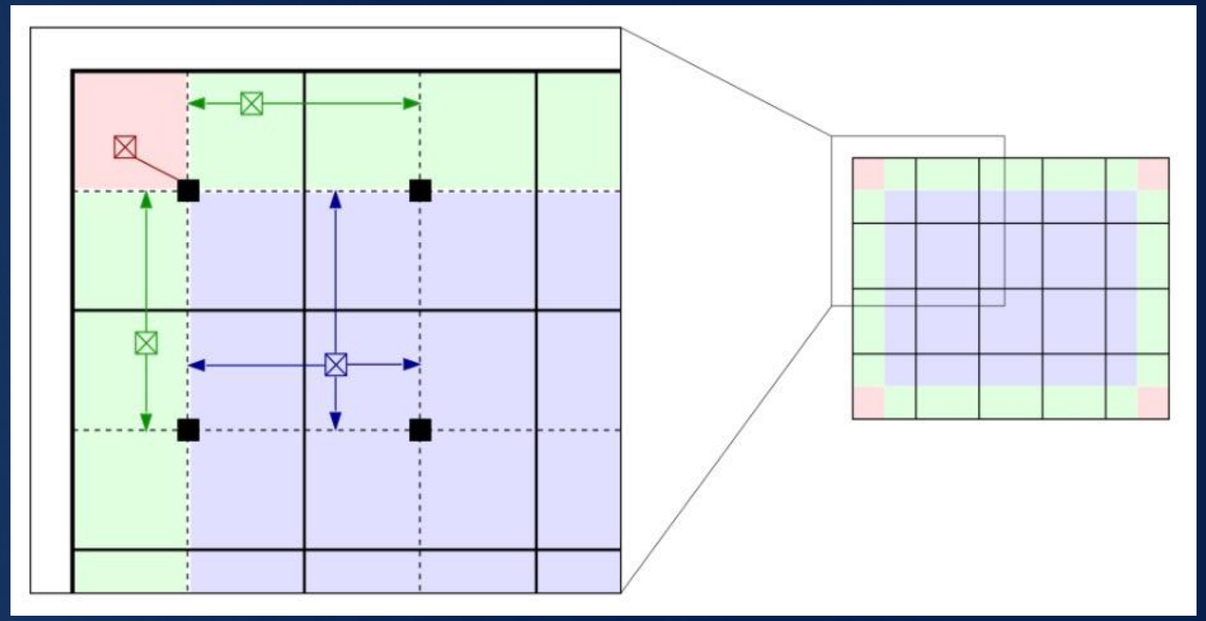
- **Global Transformation Problem -** Histogram equalization applies a single intensity mapping to the entire image, treating dark lung regions the same as bright bone areas despite their different enhancement needs. This one-size-fits-all approach means improving contrast in one area often ruins it in another, potentially hiding important pathologies while enhancing irrelevant regions.
- **No Adaptive Capability -** Histogram equalization blindly applies enhancement regardless of whether an image needs it, potentially degrading well-exposed X-rays that already have good contrast. The algorithm cannot assess image quality or selectively enhance only the regions that would benefit from adjustment.

Data Preprocessing

Adaptive Histogram Equalization:

How it works:

1. Split image into m x n tiles (e.g. 8 × 8 grid)
2. Calculate histogram for each tile separately
3. Equalize each tile using its own histogram
4. Interpolate to reduce tile artifacts / visible edges



- Pixels in the bulk of the image (blue) are bilinearly interpolated
- Pixels close to the boundary (green) are linearly interpolated
- Pixels near corners (red) are transformed with the transformation function of the corner tile.

Data Preprocessing

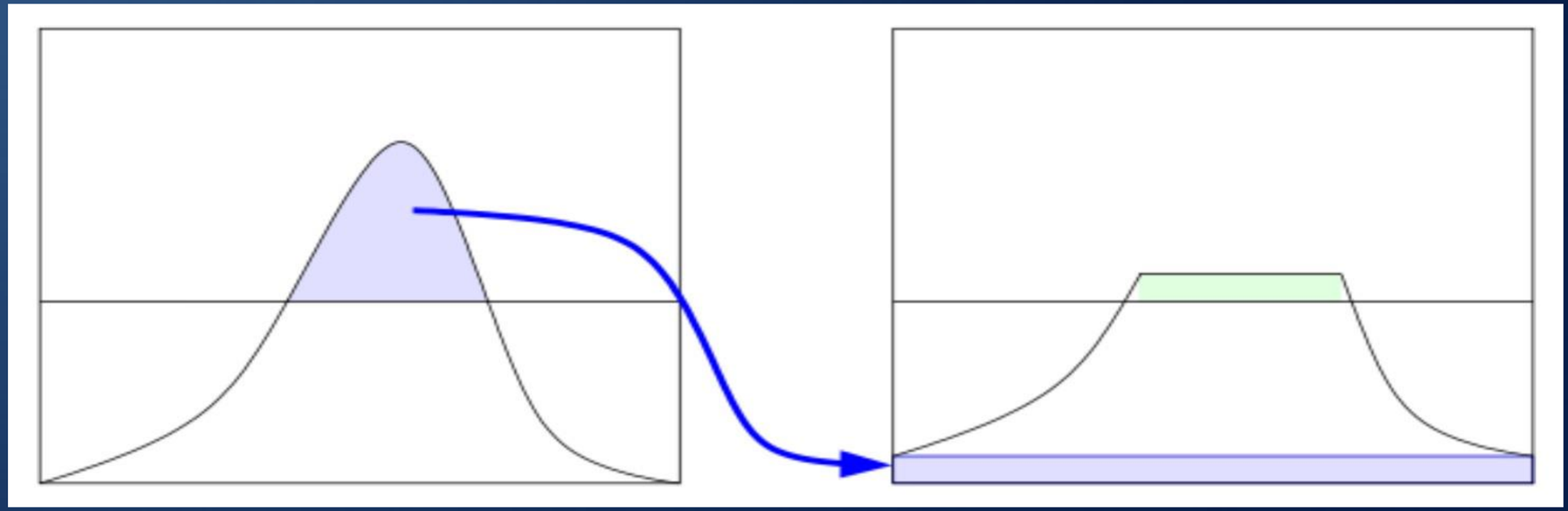
Issues with AHE:

- Ordinary AHE tends to overamplify the contrast in near-constant regions of the image, since the histogram in such regions is highly concentrated.
  - In a way, ‘forcing contrast’ that might not even be there, creating an inaccurate representation of the true image.
- Since AHE may cause noise to be amplified in near-constant regions, we turn to Contrast Limited AHE (CLAHE), in which the contrast amplification is limited, so as to reduce this problem of noise amplification.

Data Preprocessing

CLAHE:

- Introduces a **clip limit** - a maximum height for any histogram bin. When a bin exceeds this limit, the excess pixels are "clipped off" and redistributed evenly across all bins. This prevents any single intensity from dominating the transformation.
  - Low clip limit (1-2): More clipping, less contrast, preserves uniformity
  - Medium clip limit (3-4): Balanced enhancement, good for medical imaging
  - High clip limit (5+): Less clipping, approaches regular AHE behaviour



Data Preprocessing

Focal Loss:

- In the case of pneumonia detection (or any imbalanced classification), standard cross-entropy loss treats all examples equally. This causes the model to be dominated by easy, well-classified examples (usually from the majority class).
- Focal loss modifies cross-entropy by adding a modulating factor that reduces the loss contribution from easy examples and focuses training on hard, misclassified examples.

$$CE(p, y) = \begin{cases} -\log(p) & \text{if } y = 1 \\ -\log(1-p) & \text{if } y = 0 \end{cases}$$

$$FL(p, y) = \begin{cases} -\alpha(1-p)^{\gamma}\log(p) & \text{if } y = 1 \\ -(1-\alpha)p^{\gamma}\log(1-p) & \text{if } y = 0 \end{cases}$$

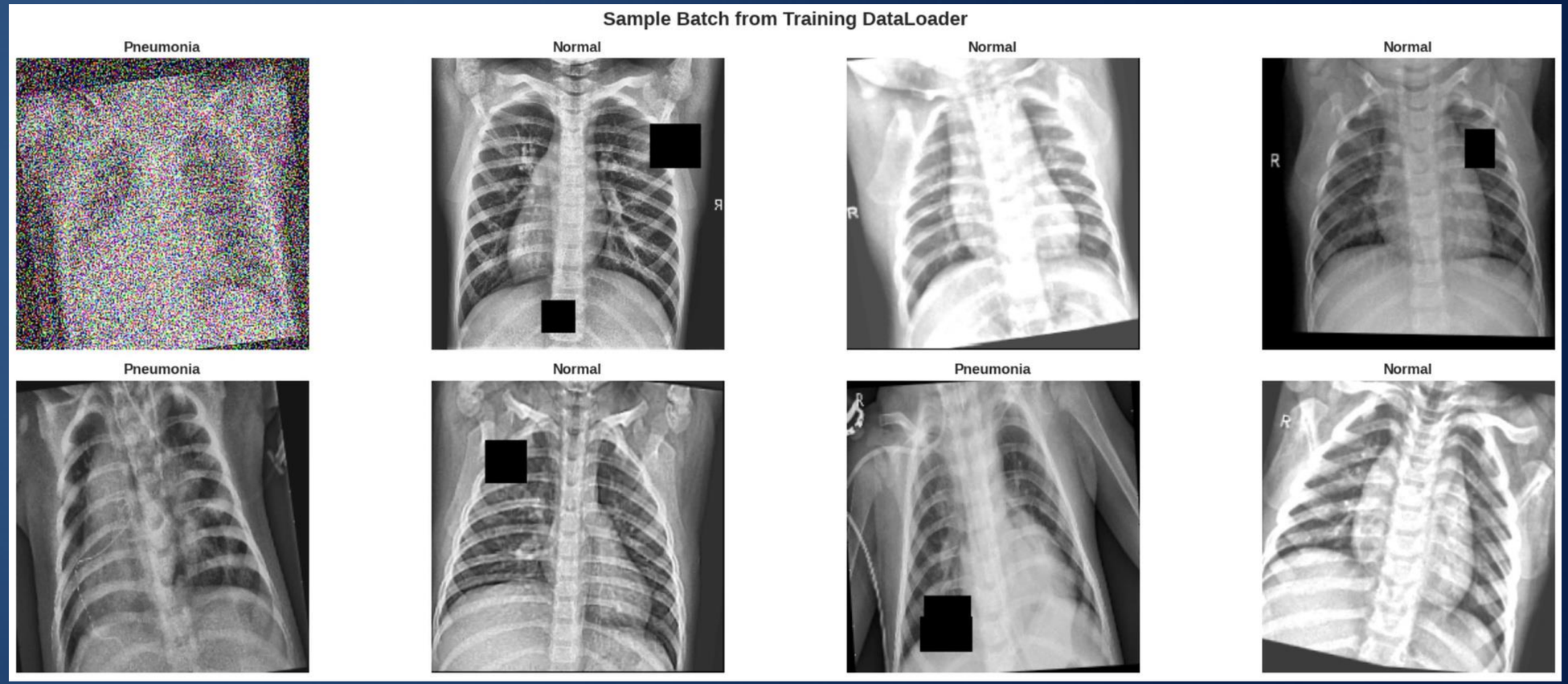
$p =$ predicted probability for the true class

$\alpha =$ weighting factor for class balance

$\gamma =$ focusing parameter (typically 2)

Data Preprocessing

Sample of DataLoader data for models 1-3, the data for models 4 and 5 only included normalization



Methods

Models Implemented and analysed:

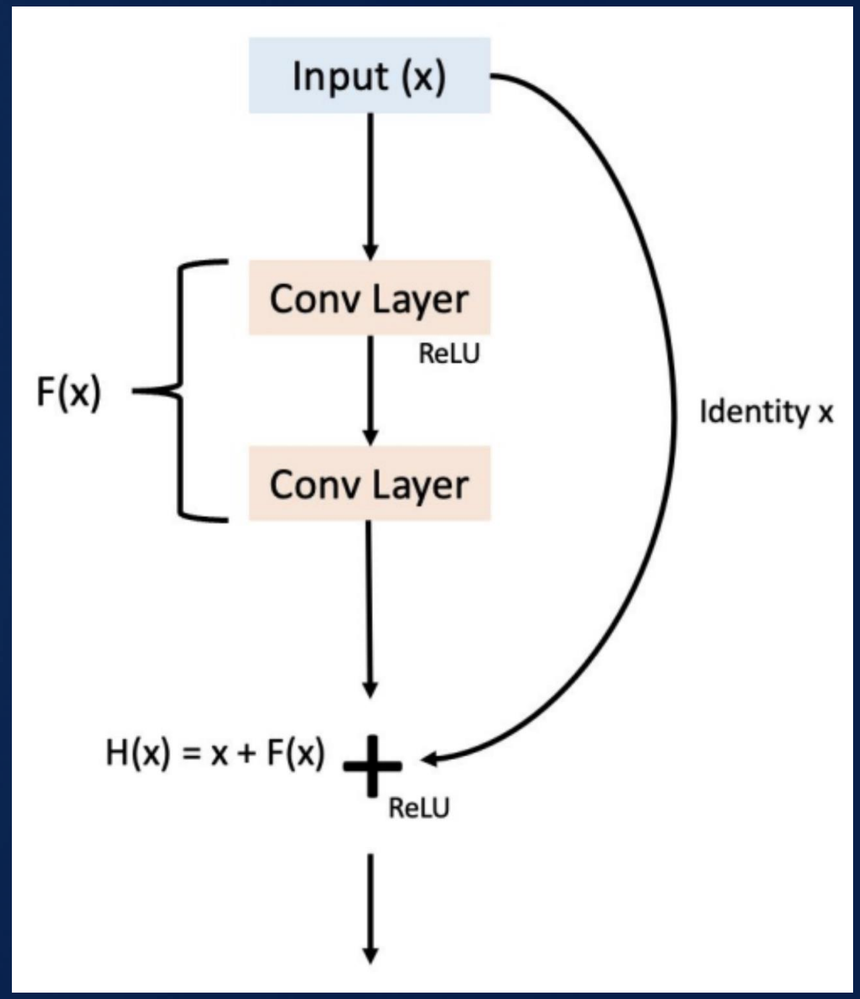
- Model 1 - Baseline ResNet50 (No Augmentation)
- Model 2 - ResNet50 + Traditional Augmentation
- Model 3 - ResNet50 + CycleGAN Synthetic Data + Traditional Augmentation
- Model 4 - Identity mapping (Normal↔Normal) CycleGAN
- Model 5 - Proper (Normal↔Pneumonia) CycleGAN

Evaluation Framework:

- Performance metrics: Sensitivity, Specificity, AUROC, F1
- Interpretability: GradCAM visualization
- Synthetic data quality: Intensity distribution analysis

Models (ResNet50)

- Input: x
  - The input feature map enters the residual block
  - Information the network already knows from earlier layers
- Residual Function: $F(x) = Conv_2(BN(ReLU(Conv_1(BN(x)))))$
  - F(x) is the output of the two convolutional layers
- Identity (Skip) Connection
  - Sends the original input x directly to the output, skipping convolution layers (x added back to F(x) at the end)
- Addition: H(x) = x + F(x)
  - Two paths merge via element-wise addition, result H(x) is the output of the block ("already know" + "just learnt")
- Activation
  - ReLU introduces non-linearity before passing output forward



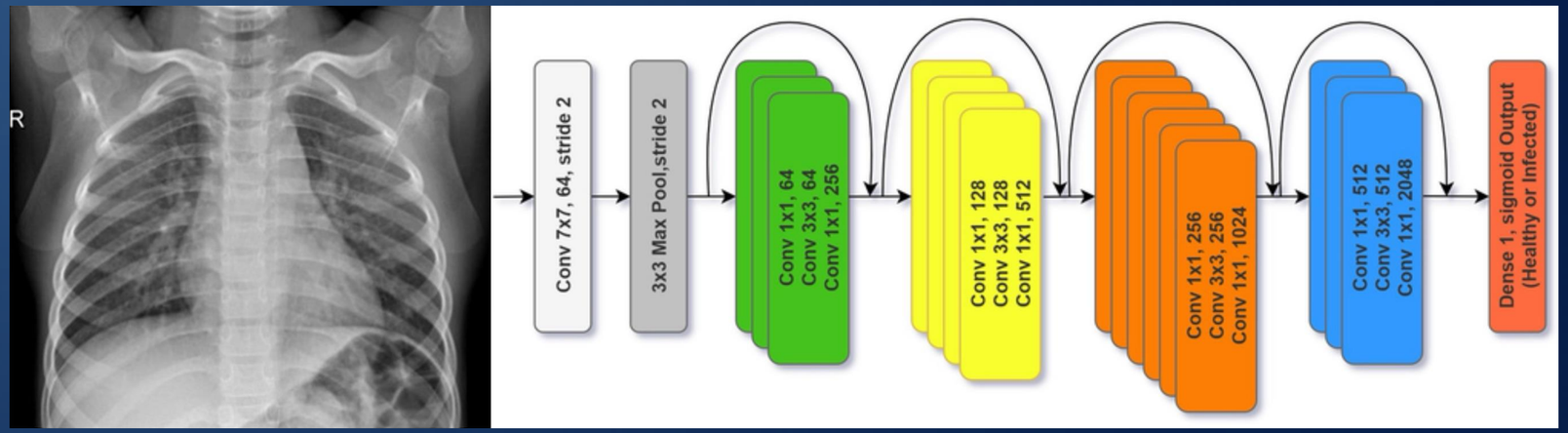
Models (ResNet50)

1. Input Processing
- 224×224×3 RGB image input (or 224×224×1 for grayscale X-rays)
- Initial 7×7 conv, 64 filters, stride 2
- 3×3 max pooling, stride 2

2. Four Residual Stages
- **Stage 1**: 3 blocks, 64 filters, 56×56 output
- **Stage 2**: 4 blocks, 128 filters, 28×28 output
- **Stage 3**: 6 blocks, 256 filters, 14×14 output
- **Stage 4**: 3 blocks, 512 filters, 7×7 output

3. Key Features
- Total of 50 layers (hence ResNet50)
- Skip connections every 2-3 layers
- Batch normalization after each convolution
- ReLU activation throughout

4. Output
- Global average pooling
- 1000-way fully connected layer (ImageNet)
- Softmax for classification



Models (CycleGAN)

(a) Basic Concept

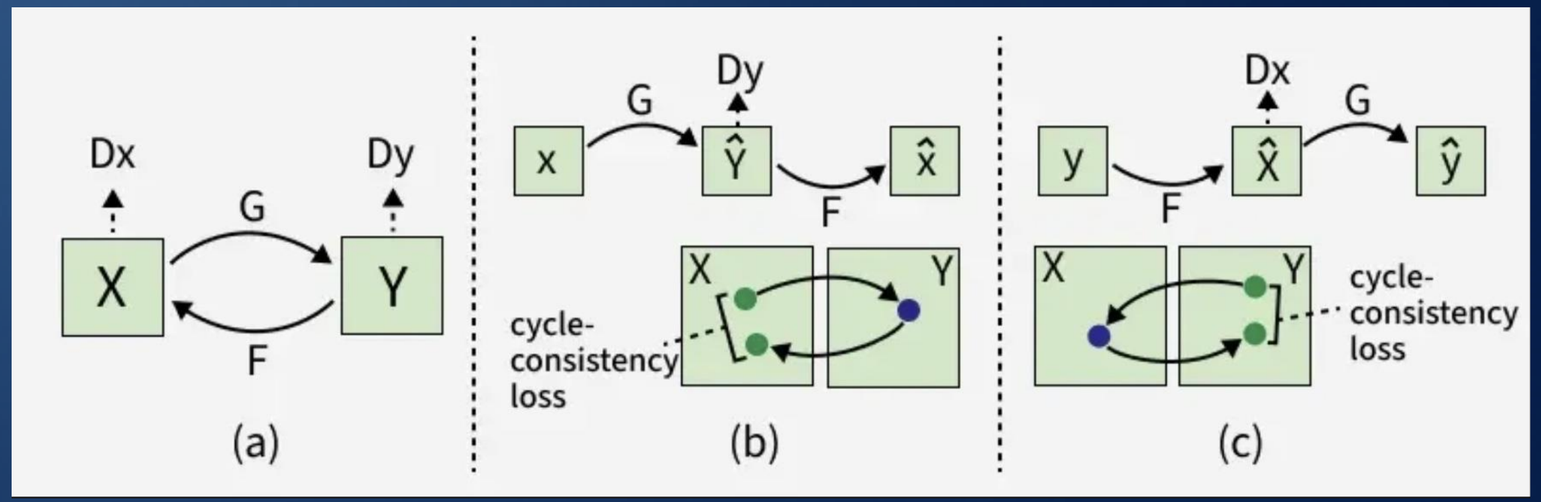
- Generator G: X→Y (Normal→Pneumonia)
- Generator F: Y→X (Pneumonia→Normal)
- Two discriminators validate authenticity of each domain

(b) Forward Cycle Consistency

- $X \rightarrow G(X) = \hat{Y} \rightarrow F(\hat{Y}) = \hat{X}$
- Normal → Fake Pneumonia → Reconstructed Normal
- Loss: $||\hat{X} - X||_1$ (should return to original)

(c) Backward Cycle Consistency

- $Y \rightarrow F(Y) = \hat{X} \rightarrow G(\hat{X}) = \hat{Y}$
- Pneumonia → Fake Normal → Reconstructed Pneumonia
- Loss: $||\hat{Y} - Y||_1$ (should return to original)



Models (CycleGAN)

Generators - ResNet Architecture

- 9 Residual Blocks (configurable 6-9 in official)
- Instance Normalization for training stability
- Reflection Padding to avoid border artifacts
- Input/Output: 256×256 grayscale chest X-rays
- Tanh activation for [-1,1] output range

Discriminators - PatchGAN (70×70)

- 5 convolutional layers in our implementation
- Classifies 70×70 patches as real/fake
- LeakyReLU(0.2) activation
- Output: 30×30 feature map
- Forces local texture realism

Loss Functions

- Generator Loss = L_adversarial + 10*L_cycle + 5*L_identity
- Discriminator Loss: MSE (LSGAN variant)
- Cycle Consistency: L1, λ=10
- Identity Loss: L1, λ=5

Evaluation Metrics

- **Sensitivity (Recall, True Positive Rate)**
  - Percentage of actual positives correctly identified
  - **Formula**: TP / (TP + FN)
  - **Medical context**: How many pneumonia cases did we catch?

- **AUROC (Area Under ROC Curve)**
  - Overall measure of classifier performance across all thresholds (0.5 (random) to 1.0 (perfect))
  - Threshold-independent performance measure

- **Specificity (True Negative Rate)**
  - Percentage of actual negatives correctly identified
  - **Formula**: TN / (TN + FP)
  - **Medical context**: How many healthy patients were correctly identified as healthy?

- **F1 Score**
  - Harmonic mean of precision and recall
  - **Formula**: 2 × (Precision × Recall) / (Precision + Recall)
  - Balances false positives and false negatives

Evaluation Metrics

GradCAM (Gradient-weighted Class Activation Mapping):

- Visualizes which parts of an image the CNN focuses on for its decision
- Creates a heatmap highlighting important regions
- Makes "black box" neural networks interpretable

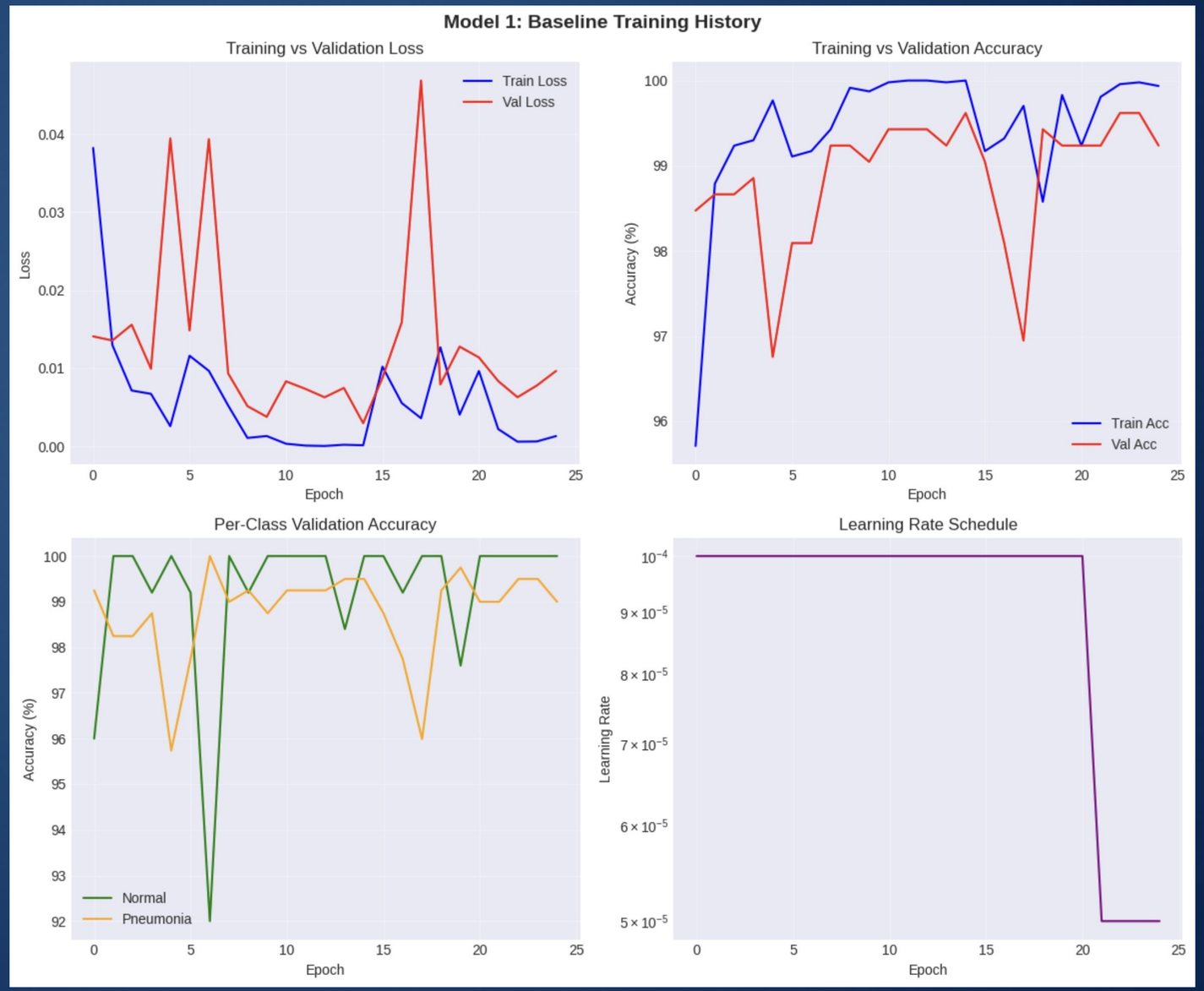
How it works:

- Forward pass through CNN to get prediction
- Backpropagate gradients to last convolutional layer
- Weight feature maps by gradients
- Generate heatmap showing importance of each region

Medical Importance:

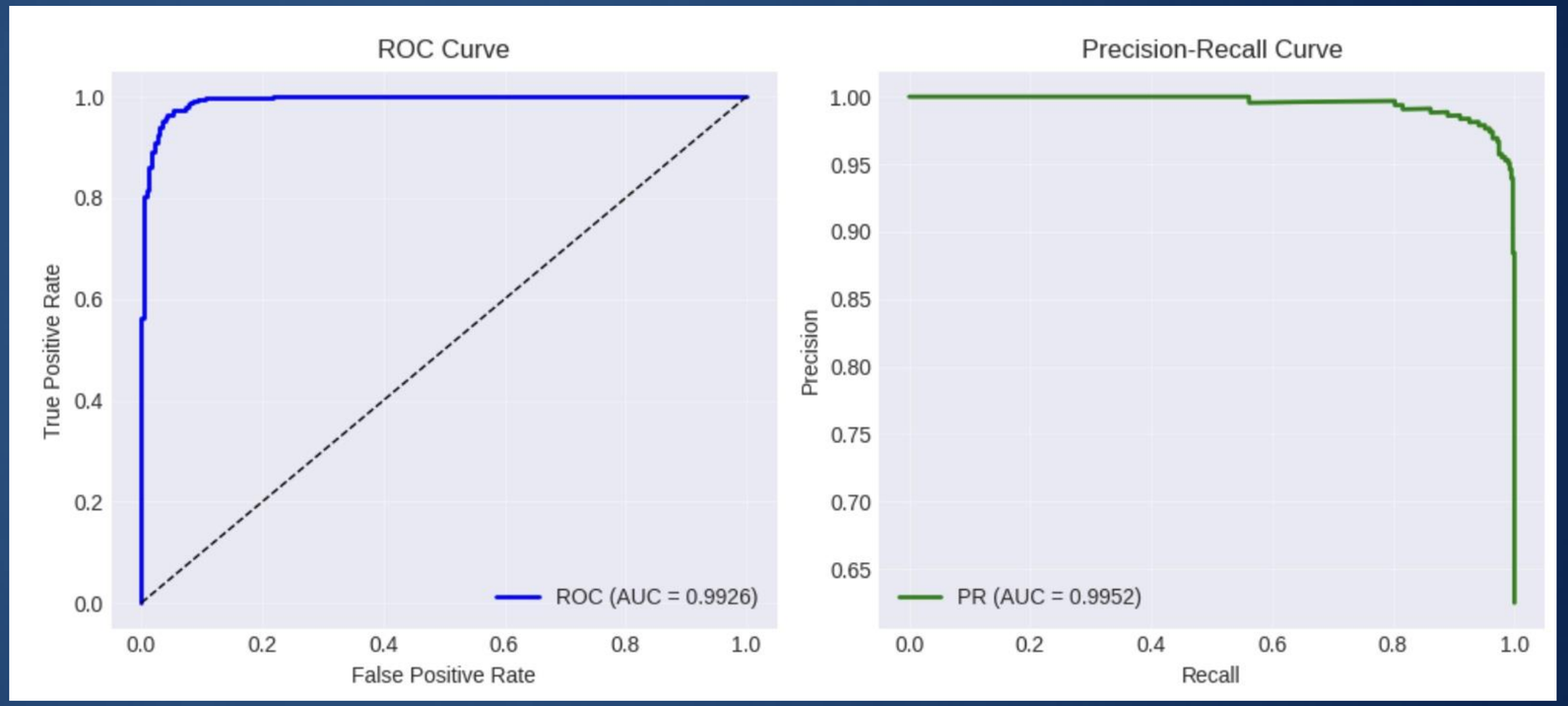
- Validation: Ensures model looks at lungs, not irrelevant features
- Trust: Doctors can see why model made its decision

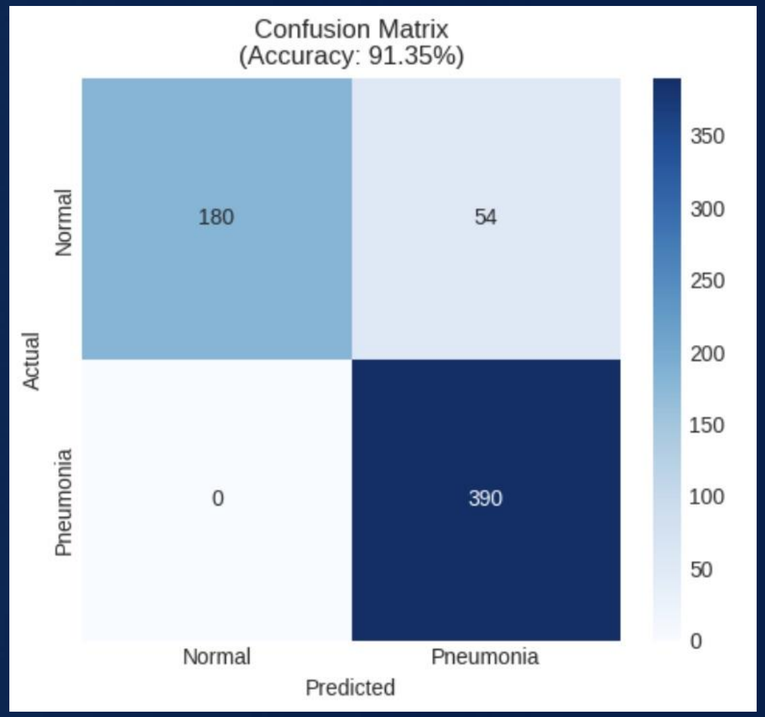
Results (Model 1)



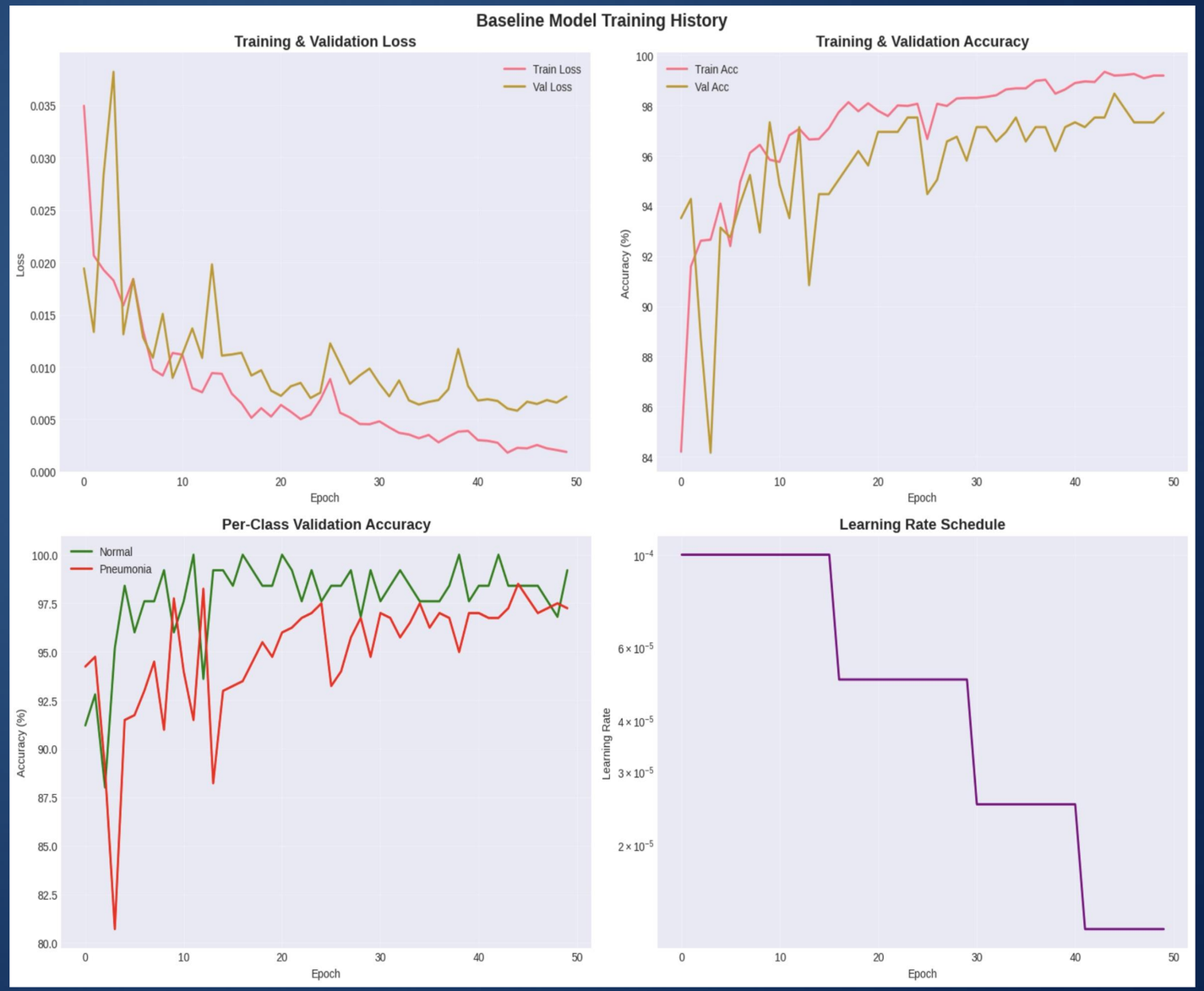

```
Key Metrics:
 Sensitivity: 1.0000 (100.00%)
 Specificity: 0.7692 (76.92%)
 AUROC: 0.9926
 AUPRC: 0.9952
 F1 Score: 0.9024
```

Results (Model 1)



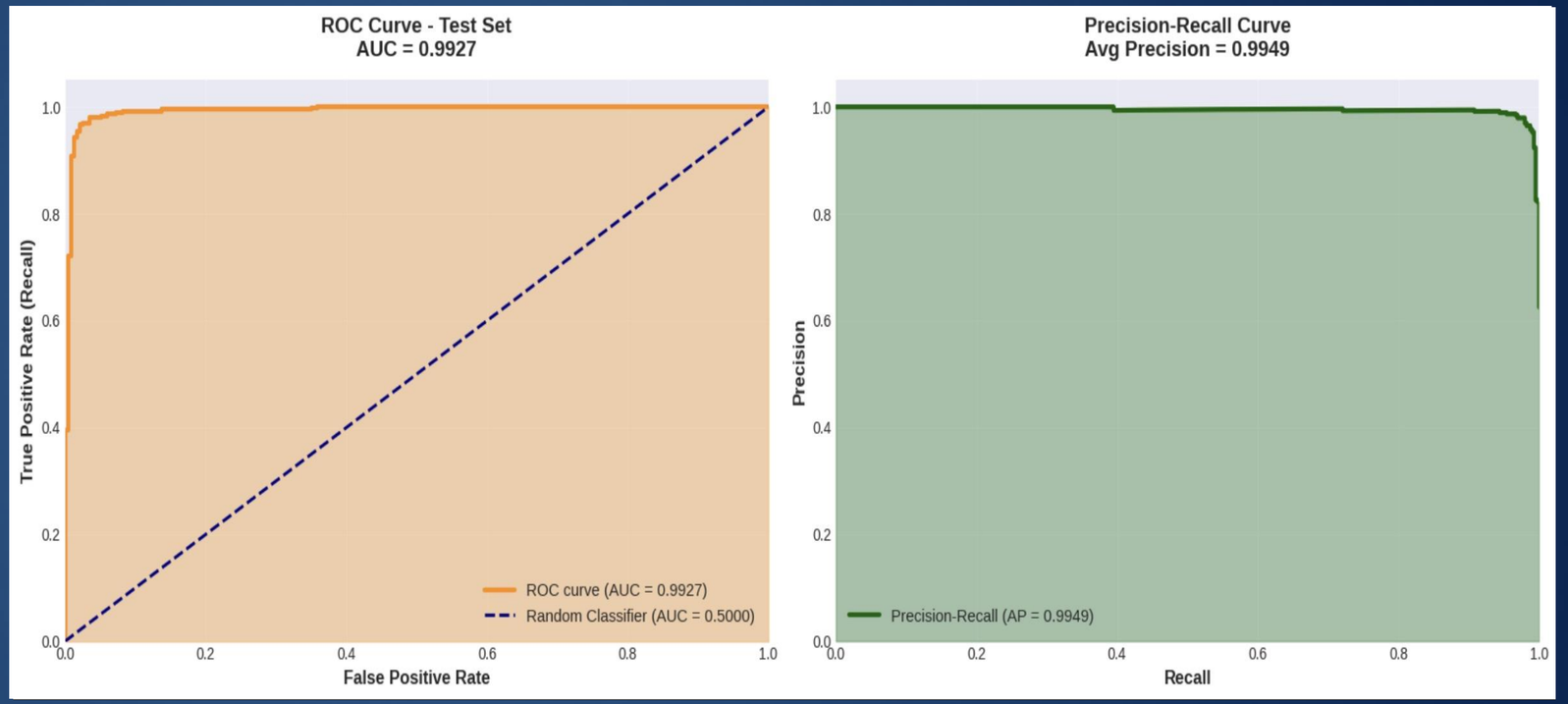


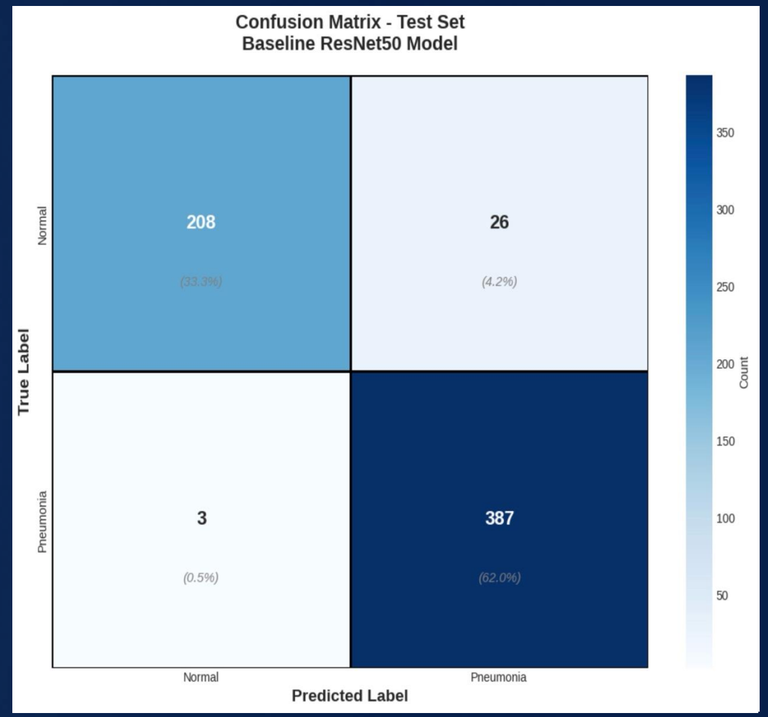
Results (Model 2)




```
Model 2 Results:
--------------------------------
Sensitivity:      0.9923
Specificity:      0.8889
Precision:        0.9371
F1 Score:         0.9642
Accuracy:         0.9535
```

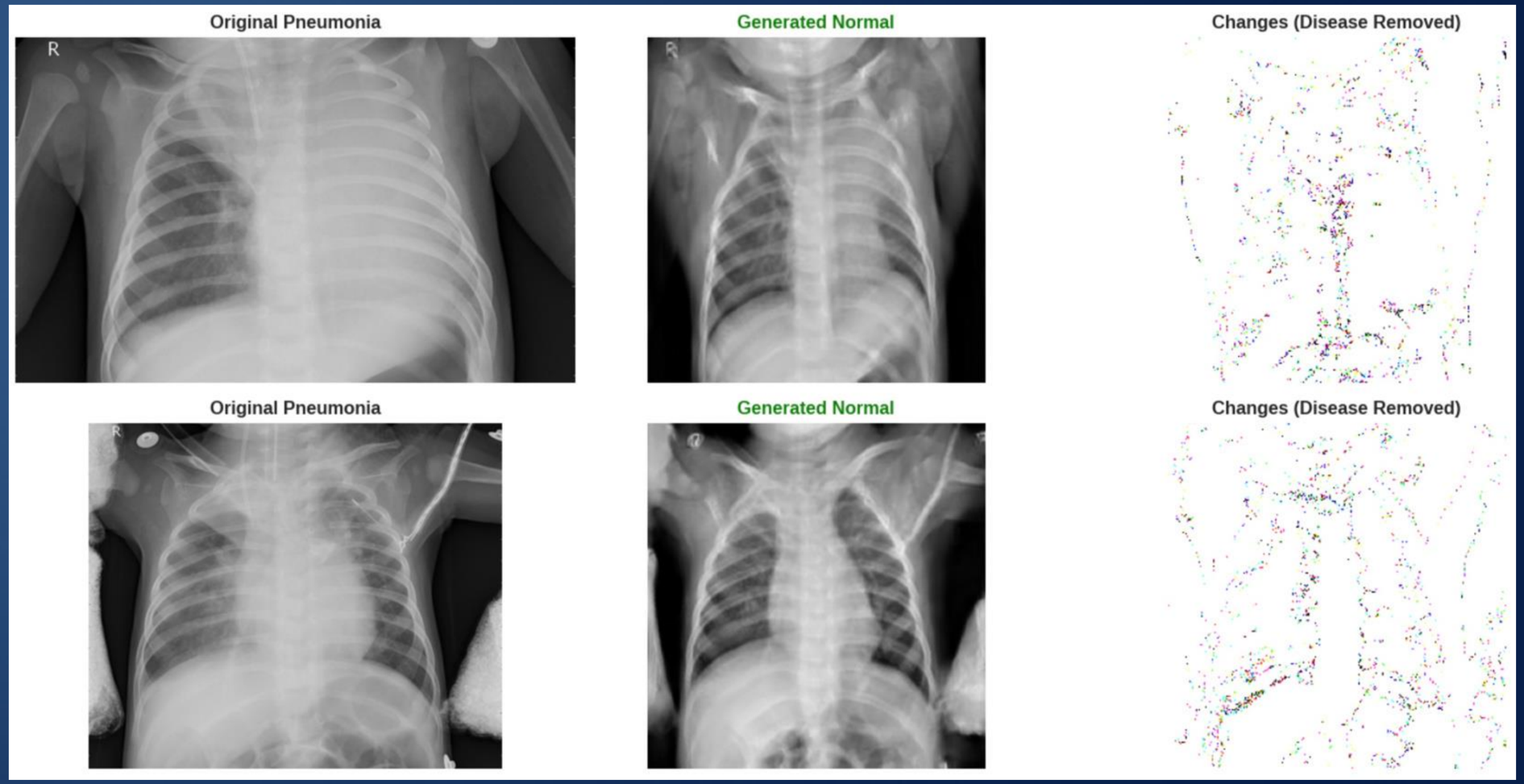
Results (Model 2)





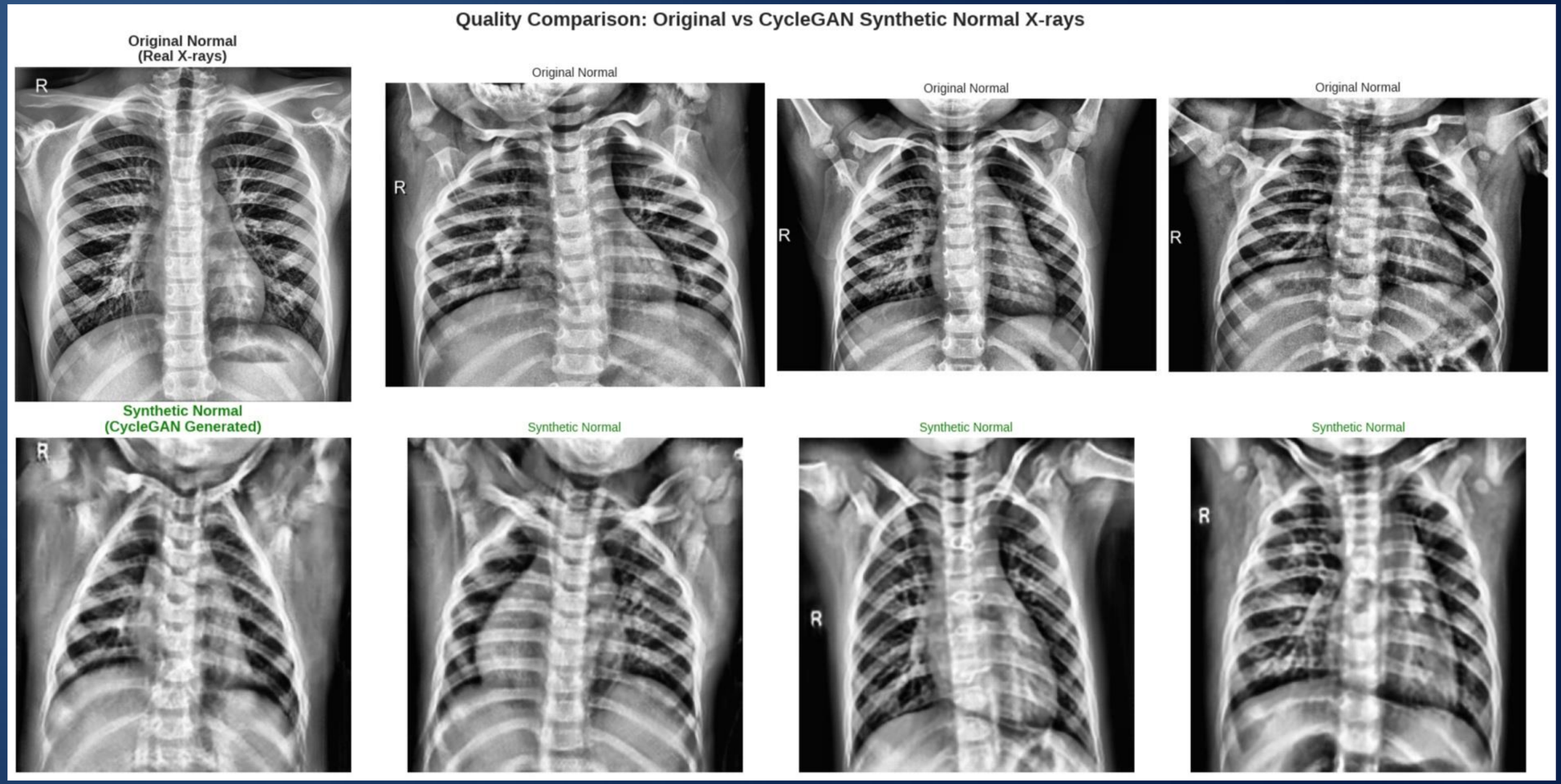
Results (Model 3)

Visualization of test generator quality:

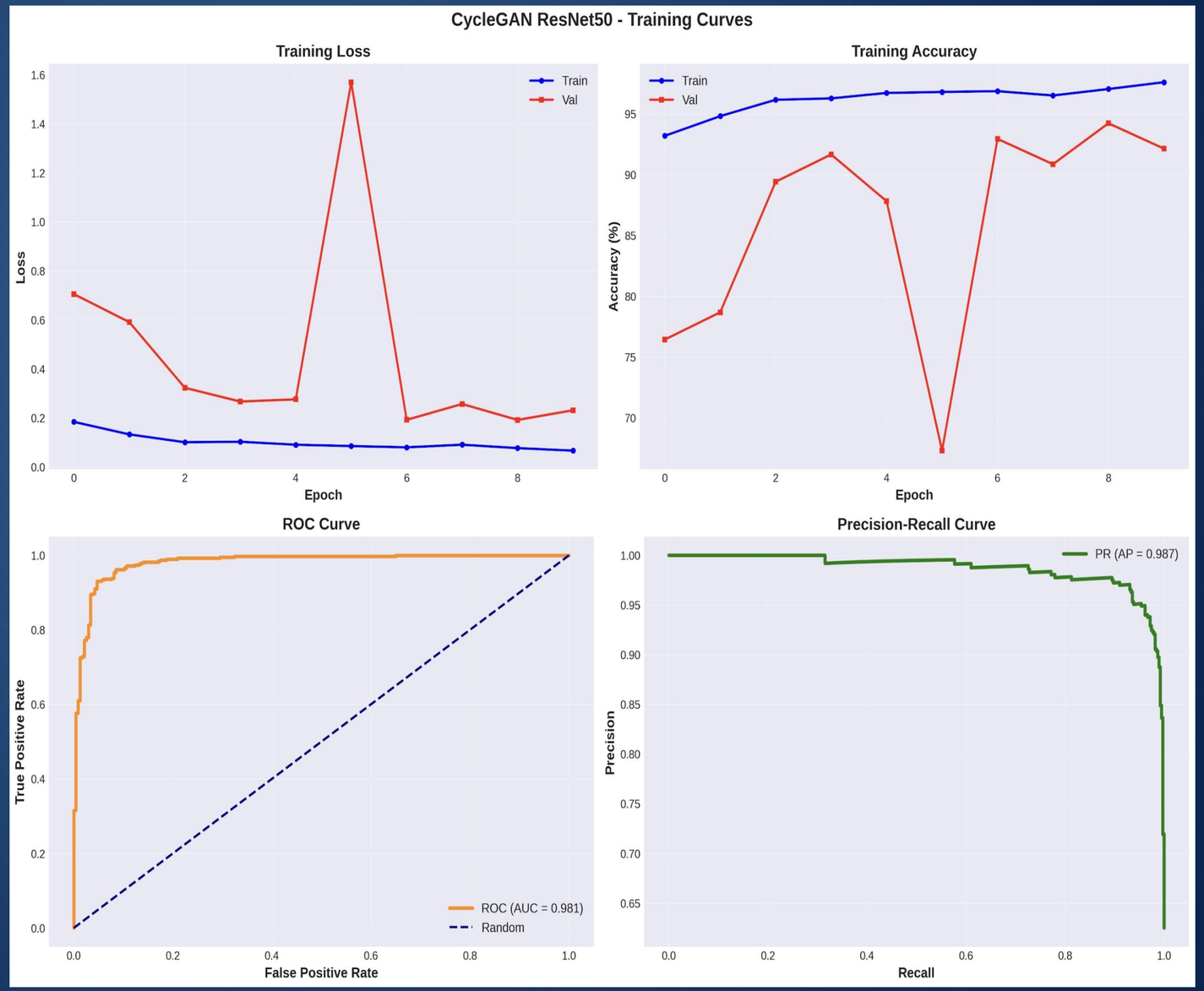


Results (Model 3)

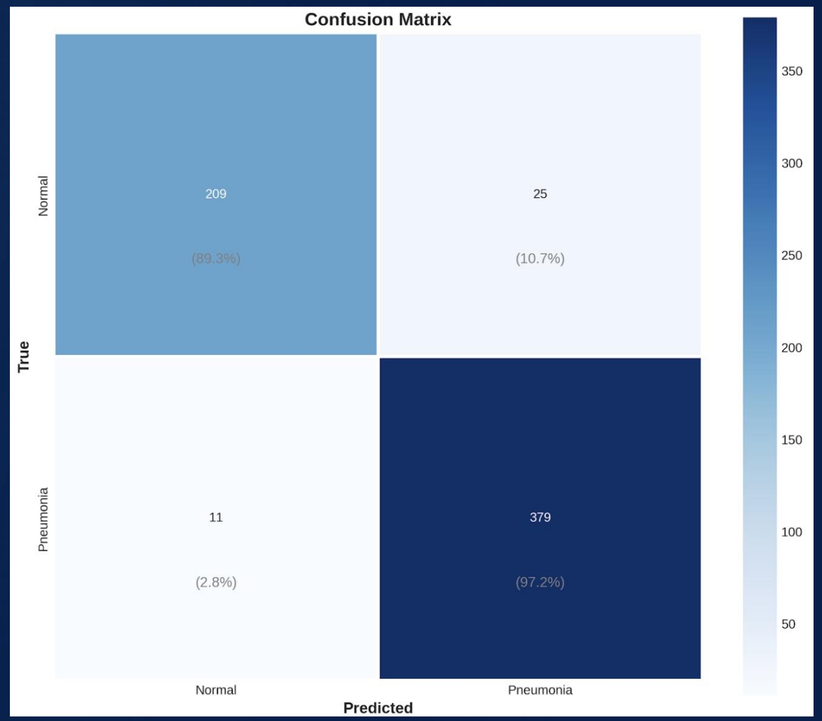
Comparison of Original and Synthetic Normal X-Rays:



Results (Model 3)

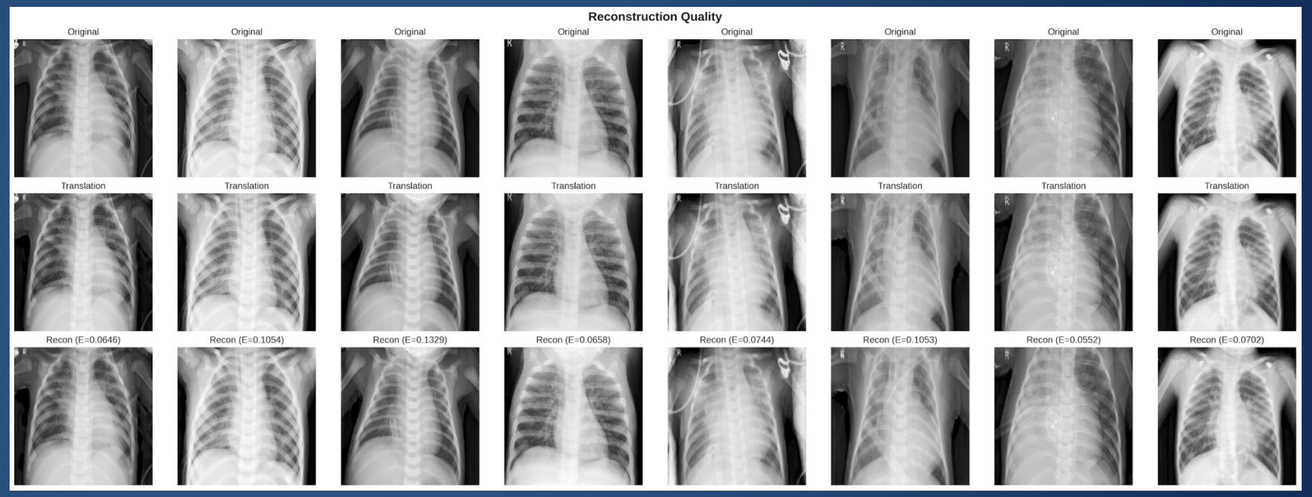



```
Model 3 Results:
------------------------
Sensitivity:      0.9718
Specificity:      0.8819
Precision:        0.9312
F1 Score:         0.9511
Accuracy:         0.9378
```

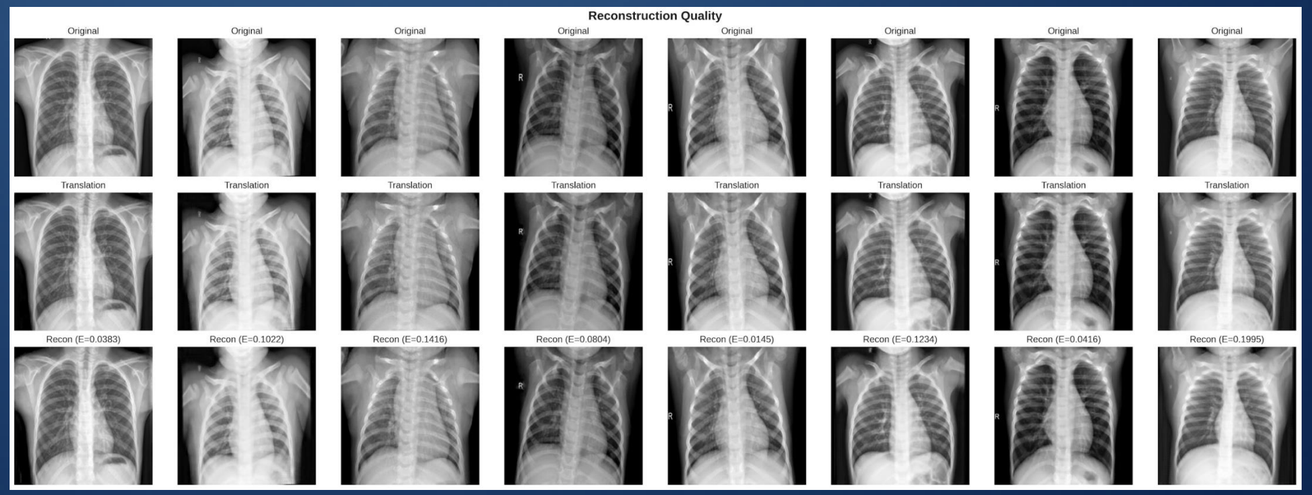


Results (Model 4)

Pneumonia Reconstruction

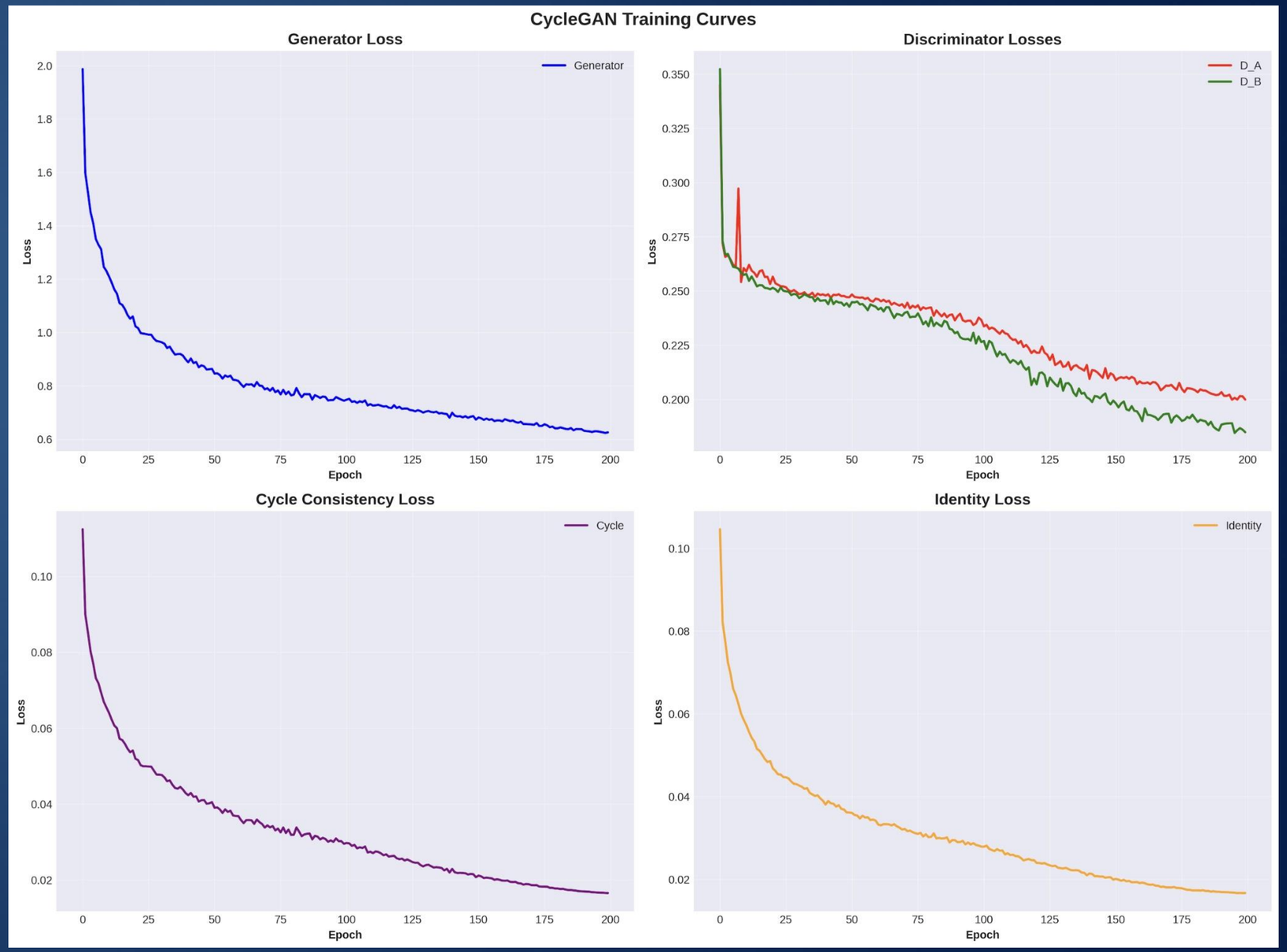


Normal Reconstruction

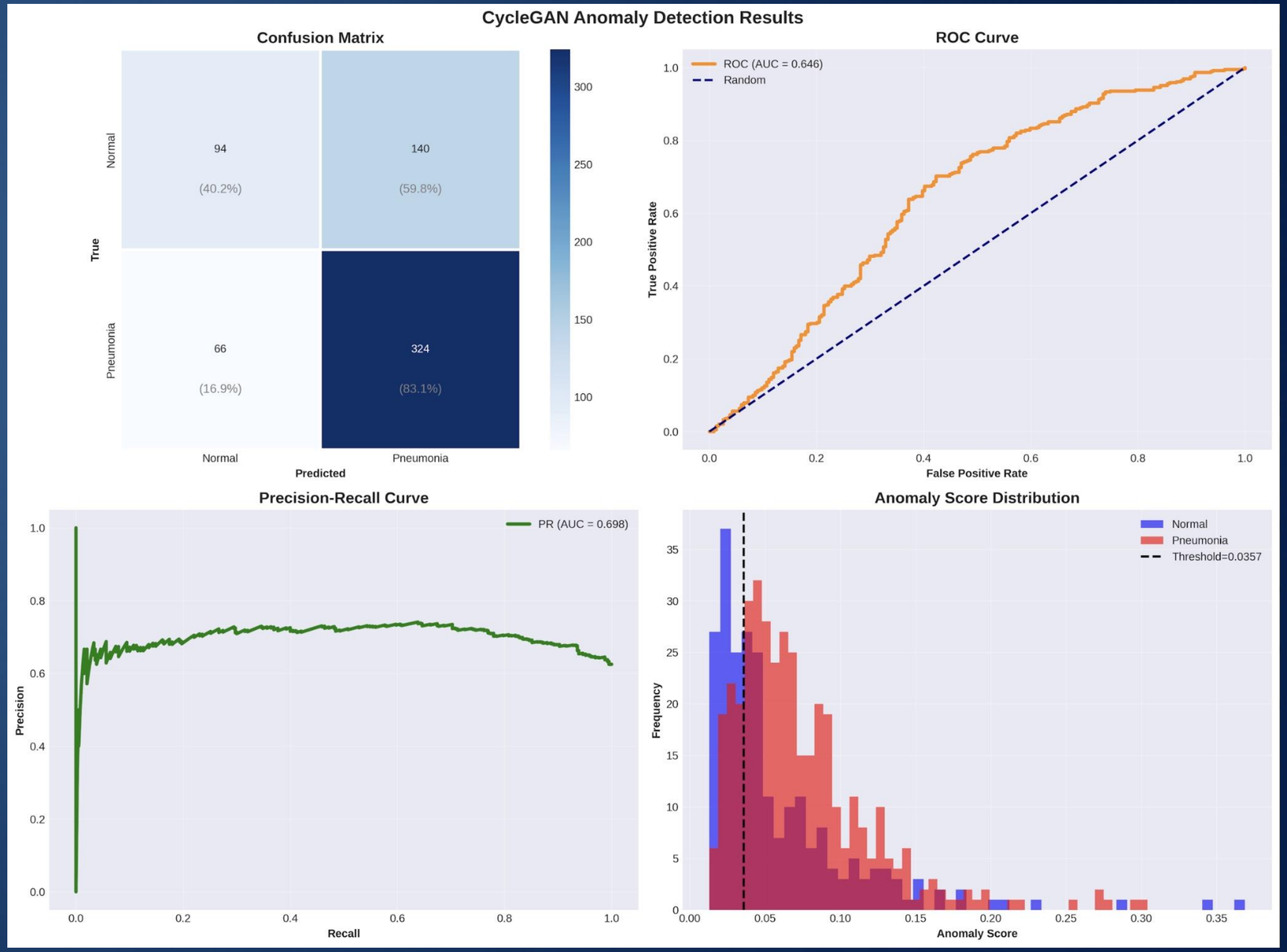



```
Model 4 Results:
--------------------------------
Metric          Value
--------------------------------
Sensitivity     0.9265
Specificity     0.2567
Precision       0.6760
F1 Score        0.7817
AUROC           0.6513
--------------------------------
```

Results (Model 4)

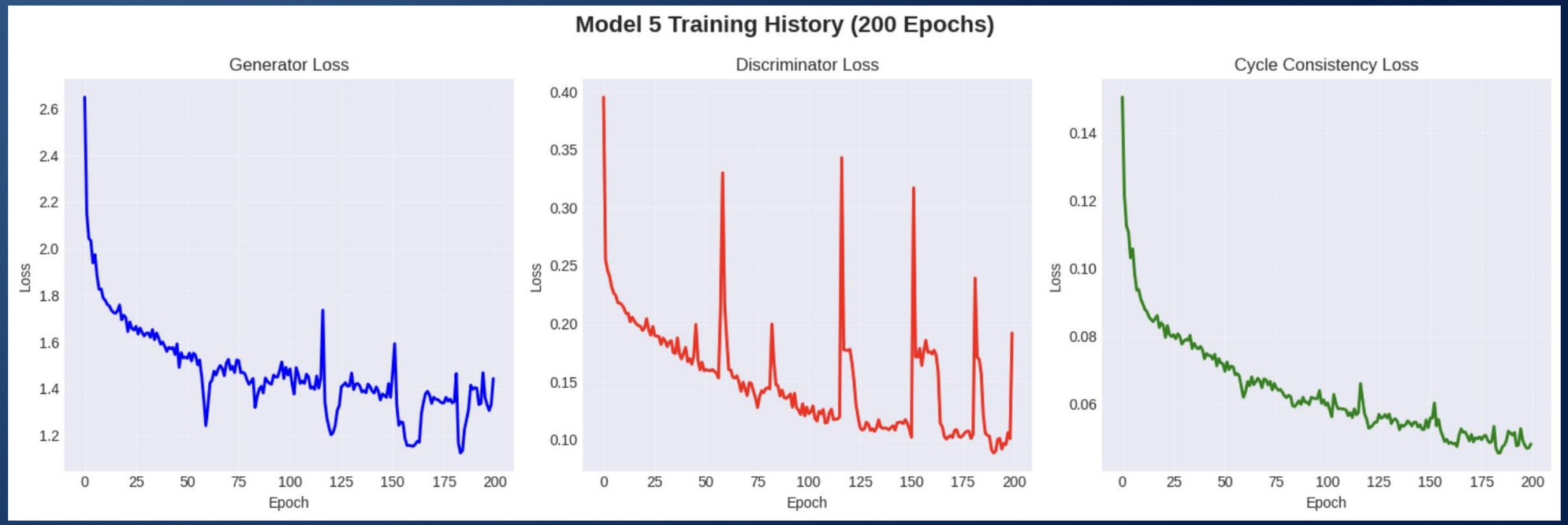


Results (Model 4)

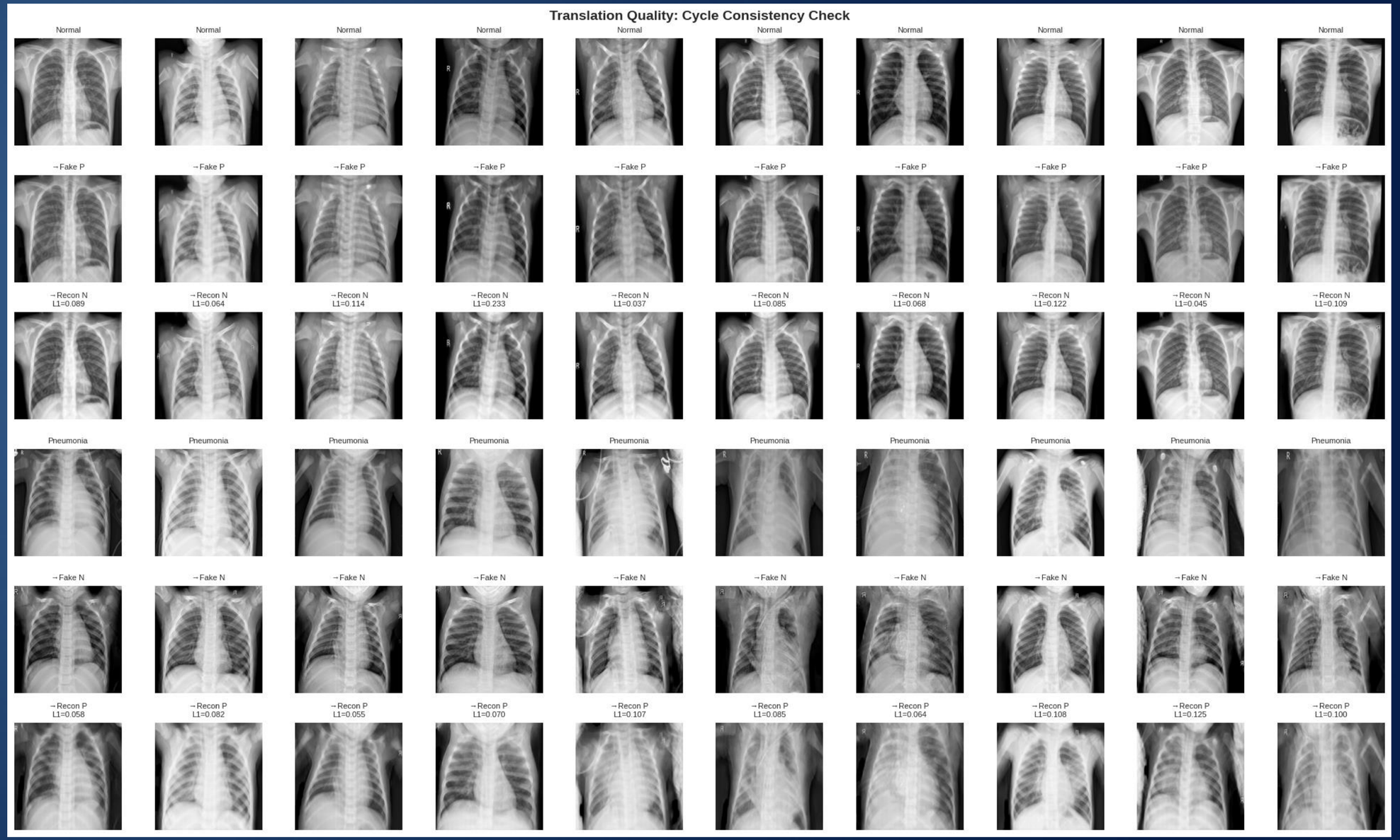


Results (Model 5)

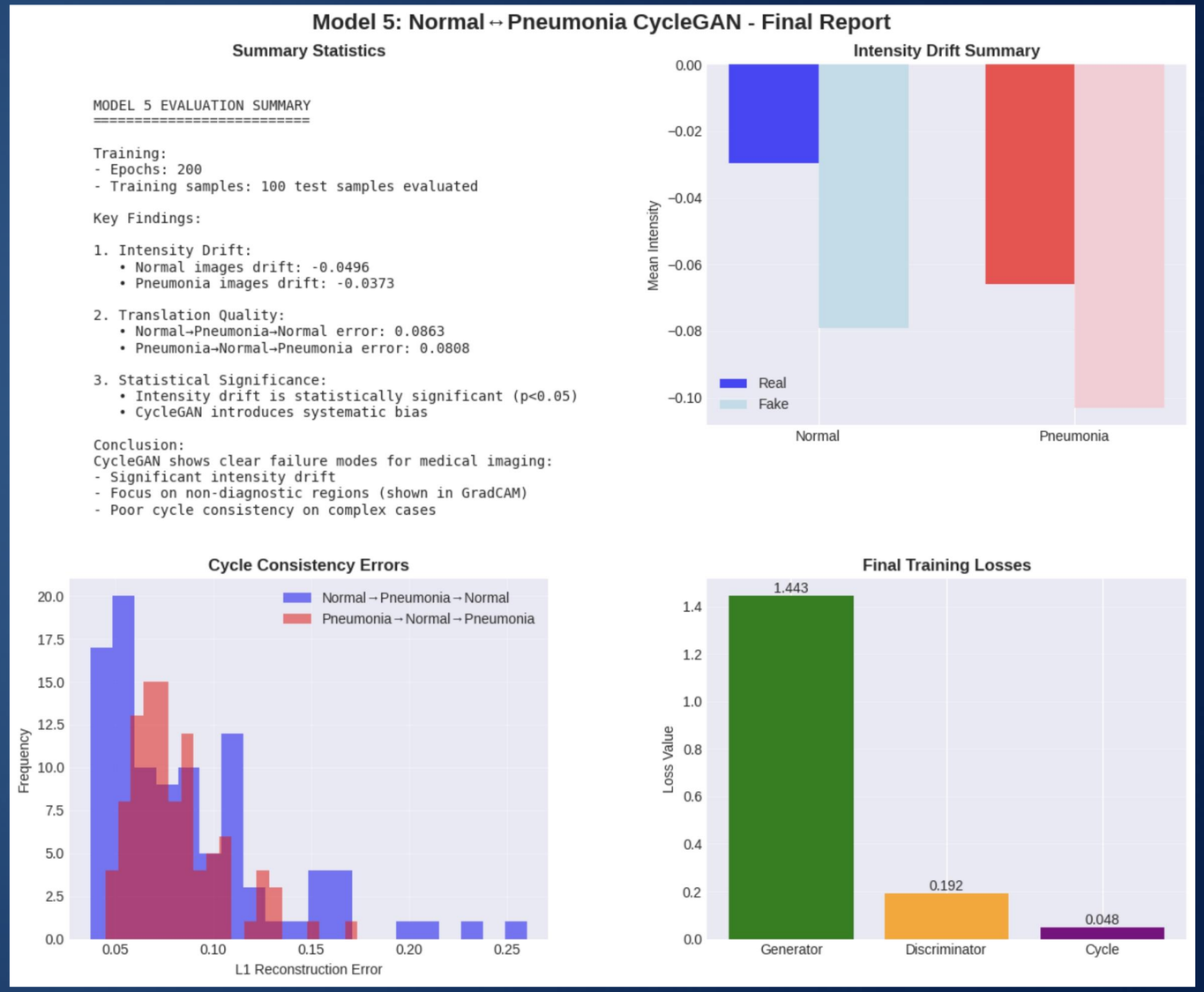
```
📊 Model 5 Anomaly Detection Performance:
AUROC: 0.5298
Sensitivity: 0.8600
Specificity: 0.3800
Precision: 0.5811
F1: 0.6935
```



Results (Model 5)



Results (Model 5)



Overall Results

Classification Performance:

Model	Sensitivity	AUROC	Specificity	F1	Precision
Model 1	1.0000	0.9926	0.7692	0.9024	0.8784
Model 2	0.9923	0.9927	0.8889	0.9639	0.9371
Model 3	0.9718	0.9803	0.8819	0.9511	0.9312
Model 4	0.9265	0.6513	0.2567	0.7817	0.6760
Model 5	0.8600	0.5298	0.3800	0.6935	0.5811

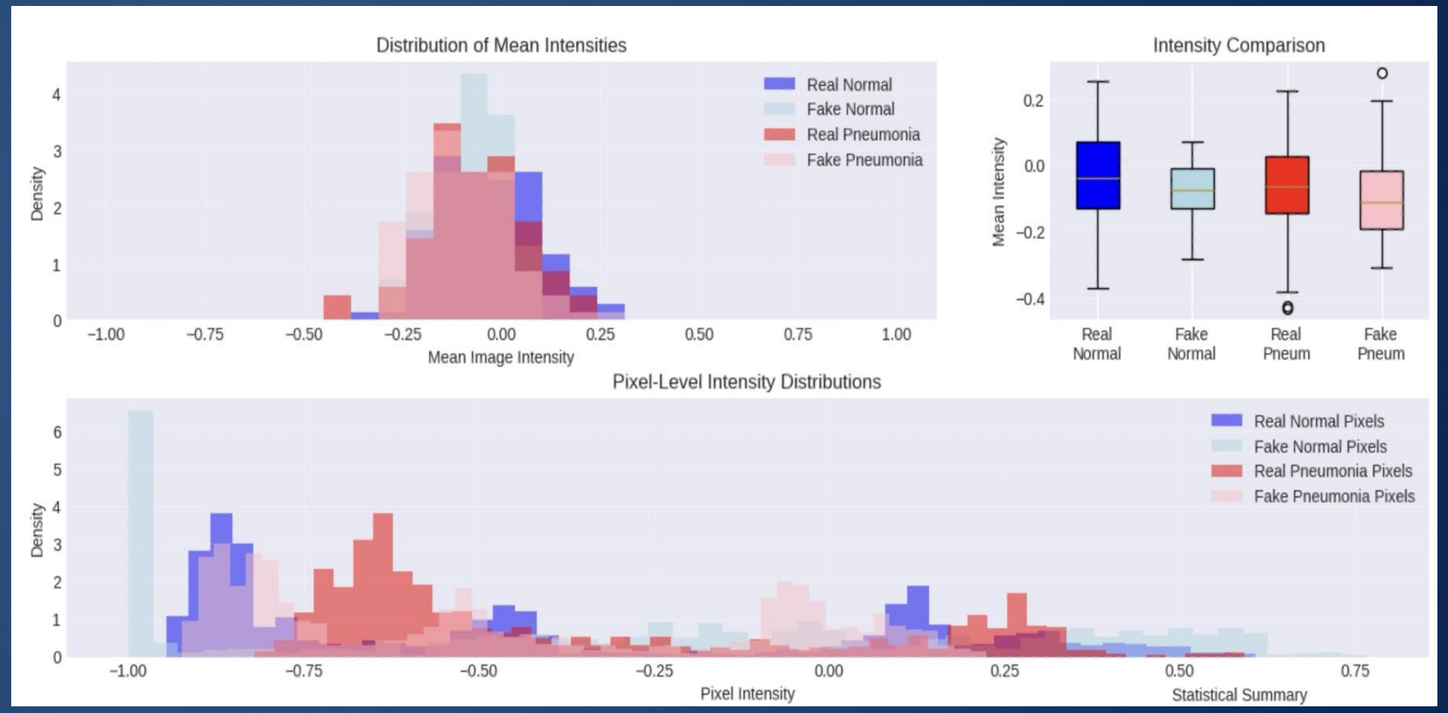
Analysis: Why did CycleGAN Fail?

We see a drop in performance when it comes to employing CycleGAN:

```
Model 2 (Traditional Aug): Sensitivity 99.2%, AUROC 0.993
Model 3 (CycleGAN Aug):    Sensitivity 97.2%, AUROC 0.980
Model 5 (CycleGAN A.D.):   Sensitivity 86.0%, AUROC 0.530
```

Why is this the case?

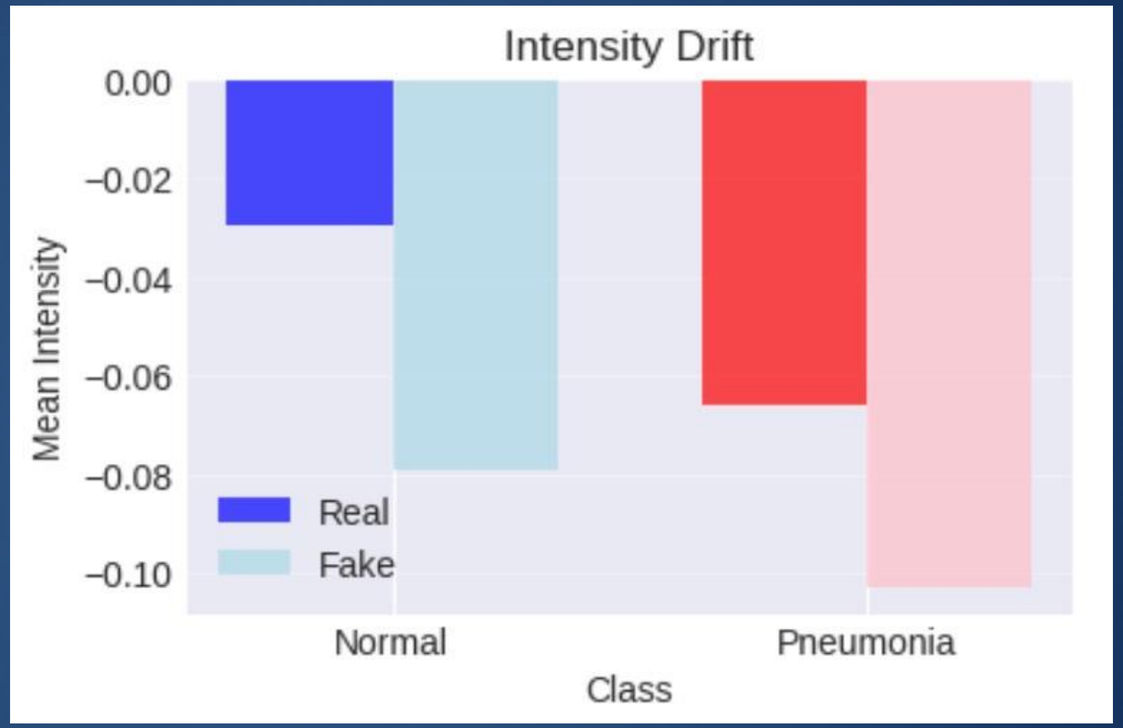
Analysis: Why did CycleGAN Fail?



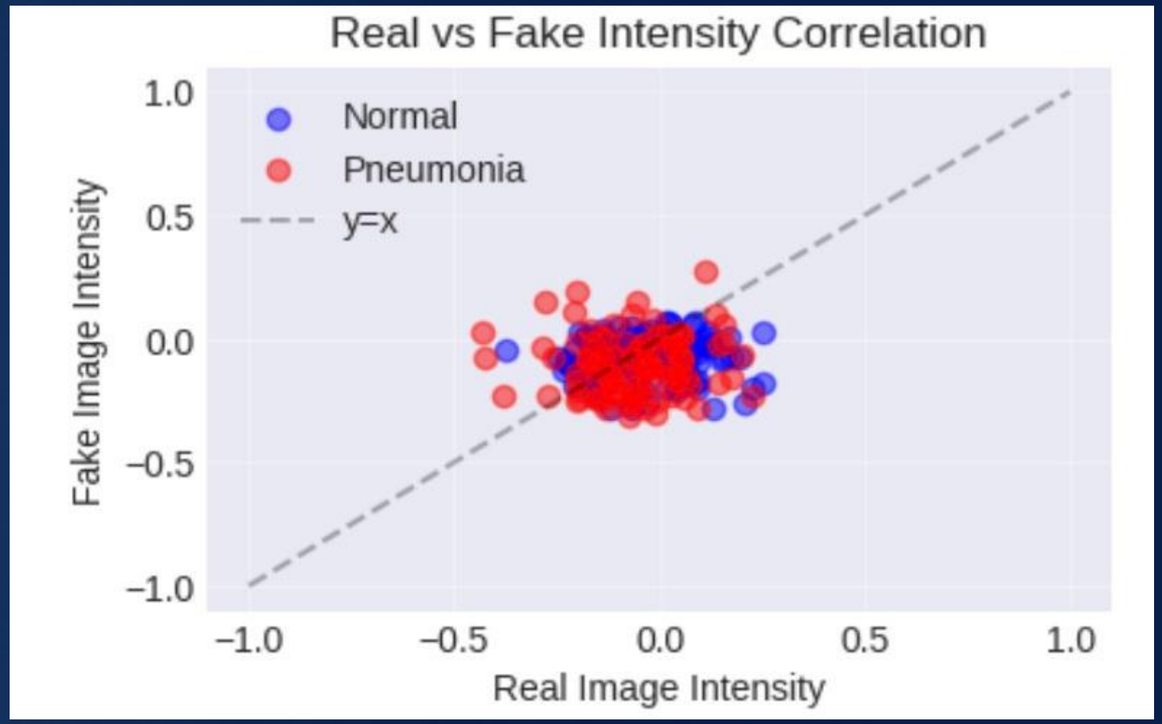

```
Mean Pixel Intensity (sample of 100 images):
 Original Normal:  122.94 ± 6.93
 Synthetic Normal: 129.66 ± 6.86
 Pneumonia:        128.04 ± 10.46
```

- **Clear separation in real data:** Normal (-0.05) vs Pneumonia (0.00) peaks distinct
- **Synthetic images collapsed:** Both fake classes cluster at -0.08 intensity
- **Systematic darkening** across all CycleGAN outputs
- **Diagnostic distinction lost:** Significant overlap between fake Normal/Pneumonia distributions

Analysis: Why did CycleGAN Fail?

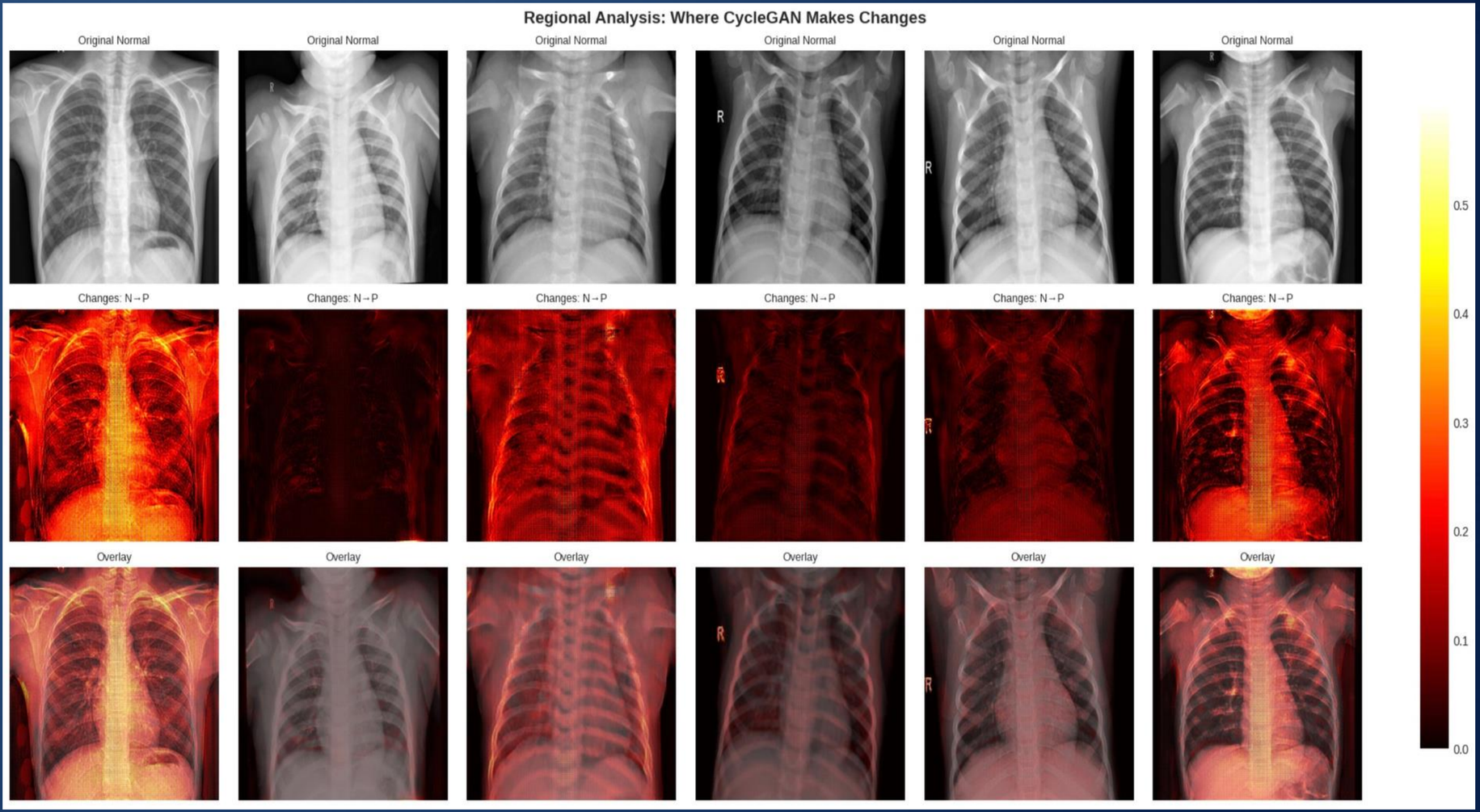


- **Normal images:** -0.08 intensity shift (largest corruption)
- **Pneumonia images:** -0.065 intensity shift
- Synthetic Normals now resemble pneumonia opacity levels!



- **Expected:** Points along y=x line (perfect preservation)
- **Observed:** Horizontal cloud at y≈0 (complete decorrelation)
- **Mode collapse:** All inputs compressed to narrow output range [-0.2, 0.2] - The model learned to generate only one "safe" intensity range that fooled the discriminator
- **Information loss:** Original intensity values unrecoverable from synthetic images

Regional Analysis: Where CycleGAN Makes Changes



- **Highest changes (yellow/bright):** Mediastinum, heart borders, spine
  - Anatomical landmarks, not pathology indicators
- **Moderate changes (orange/red):** Rib edges, shoulder regions, diaphragm
  - Structural elements that don't indicate pneumonia
- **Lowest changes (dark red/black):** Central lung fields
  - Where pneumonia actually appears
  - Critical diagnostic region left "unchanged" but intensity-corrupted

Regional Analysis: Where CycleGAN Makes Changes

```
📊 Intensity Drift Statistics:
================================================
Normal Images:
  Real:    -0.0296 ± 0.1265
  Fake:    -0.0792 ± 0.0882
  Drift:   -0.0496 (167.8%)

Pneumonia Images:
  Real:    -0.0658 ± 0.1263
  Fake:    -0.1030 ± 0.1187
  Drift:   -0.0373 (56.7%)

Statistical Significance (t-test):
  Normal drift:    t=3.200, p=0.001601 **
  Pneumonia drift: t=2.139, p=0.033650 *
```

CycleGAN learned to fool discriminators by altering high-contrast edges (ribs, mediastinum) rather than the subtle opacity changes that indicate pneumonia!

Conclusions

Complexity ≠ Better

- Traditional augmentation (99.2% sens) outperformed all GAN methods

Systematic failure modes identified:

- Intensity drift: 8% darkening destroyed diagnostic contrast
- Wrong focus: Modified bones/edges instead of lung pathology
- Mode collapse: Compressed all outputs to narrow intensity range

Clinical impact: CycleGAN corrupted the exact features radiologists need

- Generative models optimize for realism, not diagnostic fidelity

Future Work

1. Diagnostic-Aware GANs

- **Medical loss functions:** Penalize changes in lung density histogram
- **Regional weighting:** Loss = $\alpha \cdot$lung_region + $\beta \cdot$other ($\alpha >> \beta$)
- **Texture preservation:** Gram matrix matching for lung parenchyma patterns
- **Clinician feedback loop:** Radiologist scores as additional discriminator

2. Ensemble with High-Specificity Models

- **Combine Model 2 + Model 4:** High sensitivity gatekeeper + specificity filter
- **Benefits:** Reduces false positives while maintaining safety net
- **Model 4:** Could serve as the "specificity specialist" in ensemble

3. Vision-Language Models for Interpretability

- **Generate findings:** "Opacity in right lower lobe, air bronchograms present"
- **Attention visualization:** Show model reasoning alongside prediction
- **Clinical integration:** Reports that match radiologist workflow
- **Dataset opportunity:** Pair your X-rays with MIMIC-CXR reports

4. Optimal Transport (OT) Theory Framework

- **Bounded augmentation:** Prove max Wasserstein distance between real/synthetic
- **Formal guarantee:** $W(P_{real}, P_{synthetic}) < \varepsilon$ preserves diagnosis
- **Sinkhorn regularization:** Efficient OT computation for medical constraints

Thank you!