Final Project Report: Multi-Class Classification of Thoracic Diseases Using VinDr-CXR

By Tony Gibbons, Meric Ozcan, Mohak Buch, Alexander Caichen

1. Introduction

Early and accurate detection of thoracic diseases is crucial in medical diagnosis. Chest X-rays (CXR) remain one of the most common and cost-effective imaging modalities. In this project, we explore a multi-label classification problem using the publicly available VinDr-CXR dataset, applying both traditional machine learning methods with engineered features and deep learning approaches.

We selected the VinDr-CXR dataset for its comprehensive annotations by radiologists, including bounding boxes and disease labels across 15 different thoracic conditions. Our goal was to develop an accurate classification system that could effectively identify multiple pathologies from a single chest X-ray image.

This comprehensive approach employs:

- A variety of visual feature extraction techniques (HOG, Fourier, LBP, edge detection)
- Multiple classifiers (Logistic Regression, SVM)
- Feature engineering with and without CNN-based features
- Detailed performance analysis across five thoracic conditions

Our goal is to develop effective classification methods for distinguishing between common thoracic pathologies while investigating the trade-offs between model complexity, computational efficiency, and diagnostic accuracy.

2. Dataset Overview

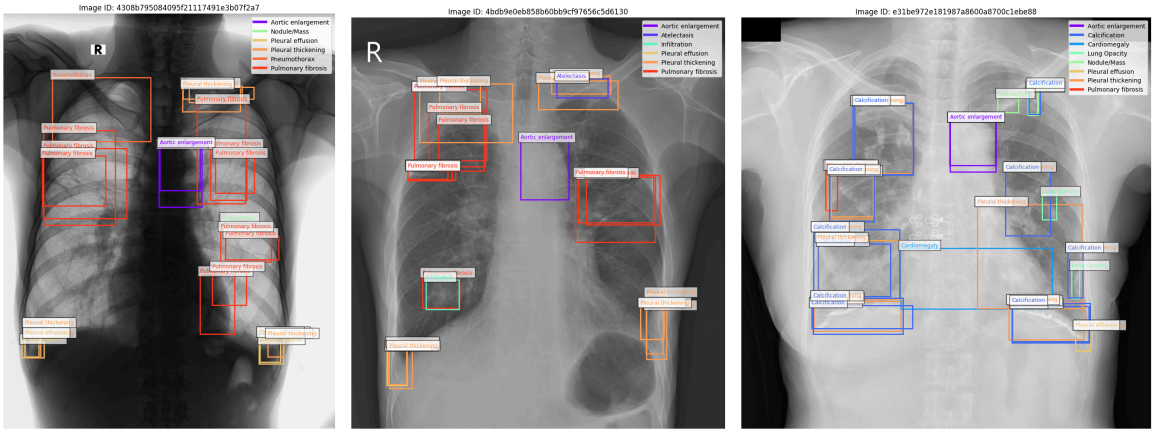
The VinDr-CXR dataset, available via GitHub, VinDr.ai, and Kaggle, contains over 18,000 postero-anterior chest X-ray images collected from two hospitals in Vietnam between 2018 and 2020. Each image is annotated by radiologists with one or more labels from a set of 14 thoracic disease classes plus "No Finding", listed below, totaling at over 50,000 available annotations.

Target Classes (15 total):

0. Aortic Enlargement
1. Atelectasis
2. Calcification
3. Cardiomegaly
4. Consolidation

5. Interstitial Lung Disease (ILD)
6. Infiltration
7. Lung Opacity
8. Nodule/Mass
9. Other Lesion
10. Pleural Effusion
11. Pleural Thickening
12. Pneumothorax
13. Pulmonary Fibrosis
14. No Finding

Shown Below: Example X-rays from the dataset illustrating the wide variation in disease combination/location, image brightness, and positioning.



3. Feature Extraction

3.1 Preprocessing

We applied the following preprocessing steps to standardize the dataset and enhance relevant visual signals:

- DICOM Processing: Read native DICOM files using the pydicom library and convert to standard image format.
- Resize: All images scaled to 512×512 resolution for feature extraction and 224×224 for CNN models.
- CLAHE (Contrast Limited Adaptive Histogram Equalization): Enhanced local contrast with adaptive histogram equalization.
- Normalization: Pixel values normalized to 0-1 range.

3.2 Feature Types

We implemented a comprehensive feature extractor class with multiple visual descriptors:

Simple Features:

- **Histogram of Oriented Gradients (HOG)**: Captures localized orientation patterns and edge gradients with:
    - 12 orientations (increased from standard 9 for finer angular resolution)
    - 16×16 pixel cells (optimized for medical structures)
    - L2-Hys block normalization
- **Fourier Transform**: Emphasizes global frequency structures through:
    - 2D FFT computation with fftshift
    - Radial binning (20 bins) with statistical measures per bin
    - Directional analysis (8 angle bins)
- **Local Binary Patterns (LBP)**: Encodes local texture descriptors with:
    - Multi-scale patterns (R=1,2,3 with P=8,16,24)
    - Uniform pattern encoding for rotation invariance
    - Normalized histogram features
- **Edge Detection**: Enhanced edge representation including:
    - Sobel gradients (magnitude and direction)
    - Canny edge detection with adaptive thresholding
    - Region-based edge density analysis (5 anatomical regions)
    - Direction histograms (36 bins)
- **Image Pyramid Decomposition**: Multi-scale representation with:
    - Gaussian pyramid (3 levels)
    - Per-level region statistics (3×3 grid)
    - Cross-scale textural comparisons
- **Spatial Features**: Structured metadata-based features derived from radiologist annotations:
    - Distance metrics between pathology regions
    - Angular relationships between findings
    - Size ratios and overlap measurements (IoU)
    - Global dispersion and convex hull metrics

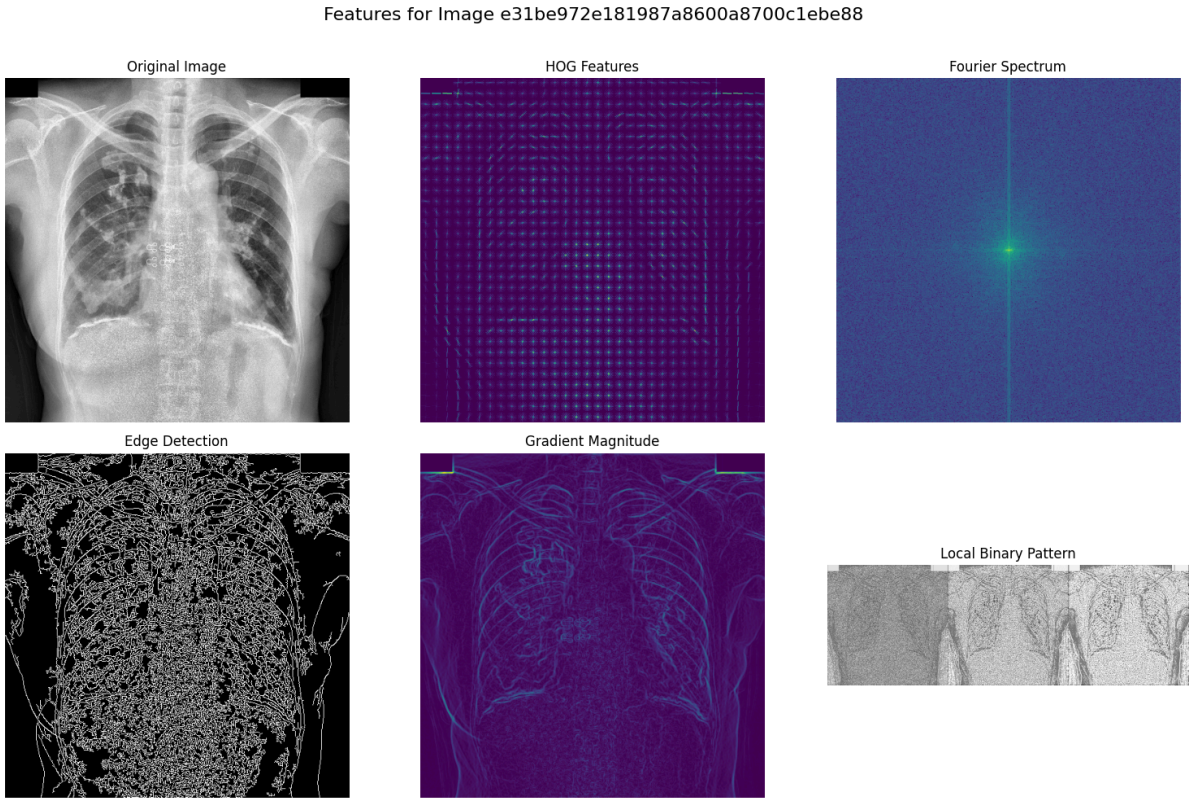
Complex Features:

- **DenseNet121 Features:** Extracted deep features from pretrained CNN as our complex feature:
    - Pretrained ImageNet weights
    - Global average pooling of feature maps
    - 1024-dimensional CNN feature vectors

Summary: This implementation includes multiple simple feature types (HOG, Fourier, LBP, Edge Detection, etc.) and one complex feature (DenseNet121 CNN features) from a pre-trained neural network. The CNN feature extraction provides a sophisticated representation learned from millions of images, which we leverage as input to our traditional classifiers as part of our hybrid approach.

Below a variety of the extracted visual features are visualized:

- HOG visualization illustrating orientation gradients over lung fields.
- Fourier magnitude spectrum highlighting high-frequency structural detail.
- Multi-level Gaussian image pyramid showing spatial structure at coarse-to-fine resolution.
- LBP visualization revealing texture patterns across localized lung regions.
- Spatial relationship feature heatmap derived from bounding box metadata.

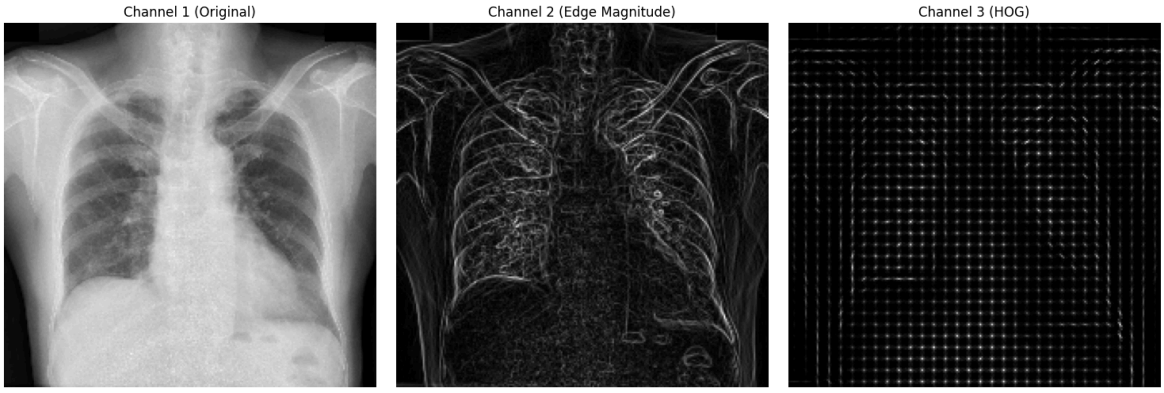


Shown Above: Examples of extracted visual features along with original image.

Multi-channel Input Preparation

We also created specialized input channels for the CNN (example shown below):

- Channel 1: Original grayscale image
- Channel 2: Edge magnitude maps
- Channel 3: HOG visualization

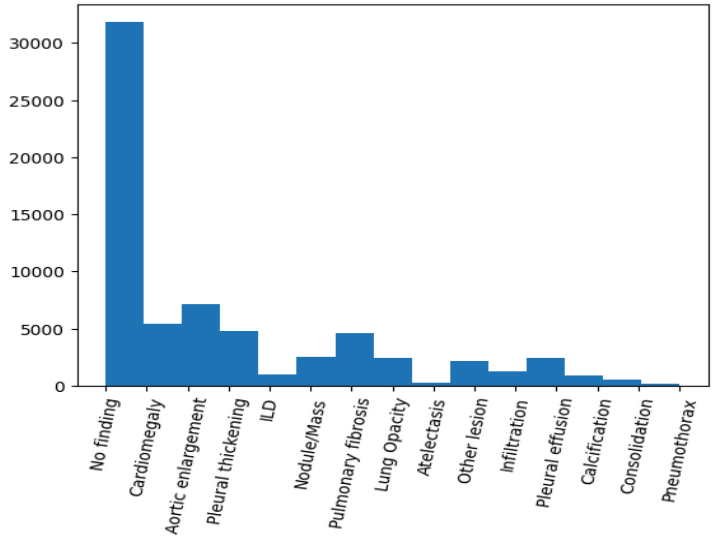


Shown Above: Feature visualization showing the original image alongside HOG features, Fourier spectrum, edge detection, gradient magnitude, and LBP features. These visualizations are generated in the notebook using the `visualize_features()` function which creates a comprehensive grid of feature representations.

4. Pathology Selection, Sample Size, and Class Balancing

Looking at the distribution of class instances present in the dataset, we see a majority of the X-rays have “No Finding”. The non-”No Finding” classes on the other hand rarely have more than 5,000 samples. To simplify image selection and result evaluation and reduce training times, only the top 4 most frequent non-”No Finding” classes were selected: Aortic Enlargement, Cardiomegaly, Pleural Thickening, and Pulmonary Fibrosis. This choice also ensured sufficient sample support for robust model training and evaluation.

Shown Below: Histogram displaying distribution of class instances within dataset.

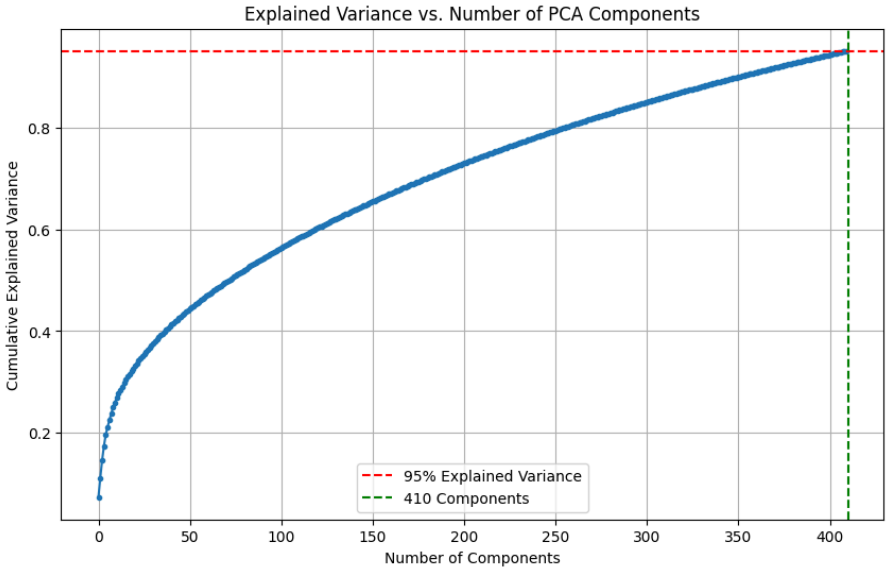


We built a smaller batch of 802 images with training runtime and equal class distribution in mind: X-rays were gradually added to the batch until the desired class sample count and distribution was reached. The equal distributions ensured the models we trained were not overfitted towards any particular class while the high class sample count ensured each model learned enough patterns for each class. We ended up with approximately 1000 instances of each class spread throughout the 802 images, with most images containing more than one class label. This batch of images was then split in a 8:1:1 ratio between training/validation/testing sets respectively (480 training images, 161 validation images, and 161 testing images) while ensuring class instance counts are still evenly spread within each set.

5. Dimensionality Reduction and Visualization

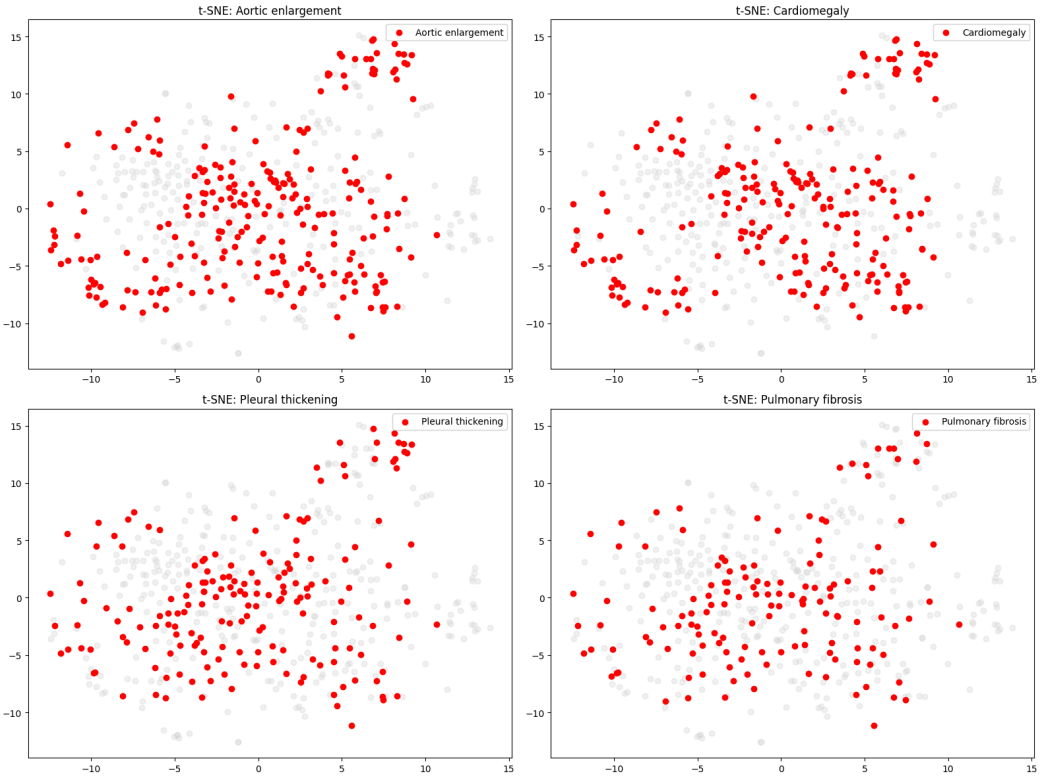
To better understand the structure of the feature space and assess class separability, both PCA and t-SNE were applied to reduce feature dimensionality for visualization.

Principal Component Analysis (PCA) was used to retain 95% of the variance across samples, or 410 of the original 512 image components. The implementation dynamically adjusted the number of components based on sample size. For very small datasets (fewer than 10 samples), a conservative default number of components was selected to avoid overfitting and dimensional instability.



Shown Above: PCA variance explained plot showing the cumulative variance ratio curve with a red dashed line at 95% and a green vertical line indicating the 410 components selected. This visualization demonstrates how the optimal number of PCA components was determined to capture most of the variance while reducing dimensionality from the original 47,503 features.

t-Distributed Stochastic Neighbor Embedding (t-SNE) provided a nonlinear 2D projection of the feature space. The algorithm automatically tuned the perplexity value based on dataset size, ensuring appropriate neighborhood modeling for both small and large sample sets.



Shown Above: t-SNE projection of selected pathologies. Each class is color-coded (grey indicates classes not selected for a current graph), and clusters demonstrate partial separation in feature space. The visualizations revealed partial separation between classes, with some overlap between similar pathologies (e.g., Pleural Thickening and Pulmonary Fibrosis showing greater overlap than Cardiomegaly and Aortic Enlargement).

These adaptive mechanisms ensured robust dimensionality reduction across all stages of experimentation, regardless of dataset size constraints.

6. Classifier Architectures and Implementation

We implemented and evaluated multiple classifier architectures to understand the trade-offs between model complexity, interpretability, and performance:

6.1 Logistic Regression:

A regularized, interpretable classifier trained on flattened 1D feature vectors composed of handcrafted features. Each feature vector was created by concatenating HOG cell descriptors, LBP maps, Fourier spectrum components, and spatial geometric features (e.g., bounding box co-occurrence). This model was efficient, robust to overfitting, and delivered strong baseline performance, after applying Principal Component Analysis (PCA) to the full feature vector. PCA was used to reduce dimensionality while preserving 95% of the original variance, which helped reduce overfitting and computational burden. This approach, combined with class-normalized sampling, enabled the logistic regression model to generalize effectively from limited data.

- **Features**: Applied to both standard engineered features and CNN-enhanced features
- **Implementation**: Used scikit-learn with L2 regularization
- **Training**: Separate binary classifiers for each disease class
- **Hyperparameter Tuning**: Grid search for regularization strength (C)
    - Aortic enlargement: Best C = 0.01 (with CNN), C = 0.1 (without CNN)
    - Cardiomegaly: Best C = 0.1 (both variants)
    - Pleural thickening: Best C = 10 (with CNN), C = 100 (without CNN)
    - Pulmonary fibrosis: Best C = 10 (both variants)
    - No finding: Best C = 1 (with CNN), C = 10 (without CNN)
- **Class Balancing**: Applied balanced class weights to handle imbalance
- **Multilabel Approach**: Independent binary classifiers with threshold calibration

6.2 Support Vector Machine (SVM)

- **Features**: Applied to both standard engineered features and CNN-enhanced features
- **Implementation**: Used scikit-learn with RBF kernel
- **Training**: Separate binary classifiers for each disease class
- **Hyperparameter Tuning**: Grid search for C and gamma parameters

    - Aortic enlargement: Best C = 10, gamma = scale (with CNN), C = 1, gamma = scale (without CNN)
    - Cardiomegaly: Best C = 10, gamma = scale (both variants)
    - Pleural thickening: Best C = 1, gamma = scale (both variants)
    - Pulmonary fibrosis: Best C = 1, gamma = scale (both variants)
    - No finding: Best C = 1, gamma = scale (with CNN), C = 10, gamma = scale (without CNN)
- **Probability Calibration**: Platt scaling for probabilistic outputs
- **Multilabel Handling**: Independent SVMs with threshold adjustment

6.3 Convolutional Neural Networks

- **Base Model**: Implemented DenseNet121 architecture
- **Implementation Details**:
    - Transfer learning with ImageNet weights
    - Fine-tuning of top 20 layers (based on hyperparameter tuning)
    - Multi-label classification with sigmoid outputs
    - Binary cross-entropy loss function
- **Hyperparameter Tuning**: Used Keras Tuner with Bayesian optimization
    - Best hyperparameters found:
        - Learning rate: 0.0005
        - Dropout rate: 0.3
        - Fine-tune layers: 20
        - Dense units: 256
- **CNN Training Pipeline**:
    - Learning rate scheduling with ReduceLROnPlateau
    - Early stopping with patience=5
    - Batch size of 16
    - Total training time: 315.61 seconds (5+ minutes)

6.4 Mutual Exclusivity Handling

Since "No Finding" is mutually exclusive with other pathologies (a patient cannot simultaneously have "No Finding" and any disease), we implemented special handling:

- **For Logistic Regression and SVM**:
    - Post-processing rules to enforce mutual exclusivity
    - If "No Finding" probability > 0.5, suppress other class probabilities
    - If any other class probability > 0.5, suppress "No Finding" probability
- **For CNN**:
    - Applied similar post-processing to network outputs
    - Experimented with custom loss functions penalizing simultaneous activation

7. Experimental Results

7.1 Model Performance Comparison

We evaluated all models using consistent metrics on the test set:

Model	Accuracy	F1 Score	AUC	Training Time (s)	Inference Time (s)
LogReg w/o CNN	0.840	0.801	0.909	1.08	0.001
LogReg w/ CNN	0.853	0.816	0.918	3.65	0.002
SVM w/o CNN	0.799	0.769	0.905	5.87	0.156
SVM w/ CNN	0.814	0.783	0.911	6.03	0.159
CNN (DenseNet121)	0.543	0.309	0.450	315.61	5.09

Our results clearly show that traditional models (Logistic Regression and SVM) significantly outperformed the CNN model, while benefiting from the inclusion of CNN features as part of their feature vectors. Logistic Regression consistently outperformed SVM models, while requiring significantly less training and inference time.

In terms of inference time, both Logistic Regression variants were extremely efficient (near-instant inference), while SVM models required approximately 0.16 seconds for prediction, and the CNN model needed over 5 seconds.

7.2 Per-Class Performance Analysis

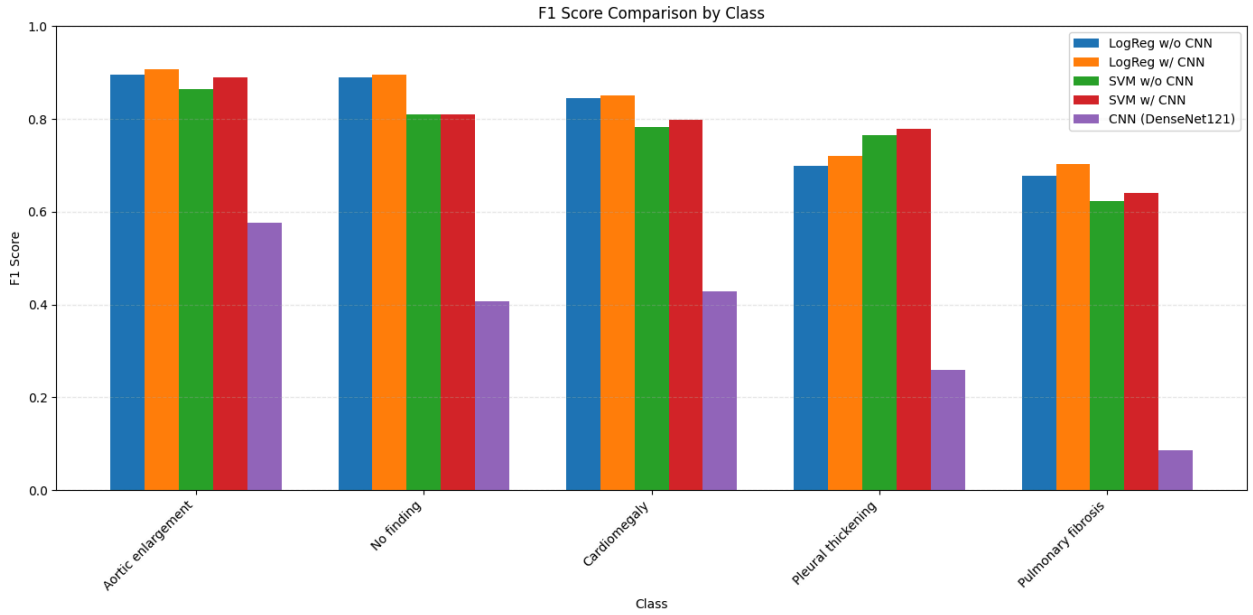
Performance varied significantly across disease classes (results for Test Set):

		Aortic Enlargement	Cardiomegaly	Pleural Thickening	Pulmonary Fibrosis	No Finding
Logistic Regression w/ CNN	Accuracy	0.894	0.913	0.776	0.801	0.938
	AUC	0.937	0.940	0.832	0.870	0.941
	F1	0.887	0.897	0.660	0.660	0.932
SVM w/ CNN Features	Accuracy	0.870	0.876	0.745	0.745	0.857
	AUC	0.939	0.960	0.818	0.870	0.860
	F1	0.876	0.870	0.661	0.610	0.813

CNN (DenseNet 121)	Accuracy	0.503	0.528	0.522	0.702	0.460
	AUC	0.474	0.462	0.374	0.494	0.445
	F1	0.494	0.406	0.135	0.143	0.365

These results reveal that "No Finding" and the cardiac conditions (Aortic Enlargement and Cardiomegaly) were consistently easier to classify than the lung pathologies (Pleural Thickening and Pulmonary Fibrosis). This pattern was consistent across all models tested, though the CNN model struggled significantly more with all classes.

Shown Below: Bar chart comparing F1 scores by disease class for the Logistic Regression model. The chart groups metrics by class, with bars for each metric type, showing the performance variation across different pathologies. The chart illustrates how "No Finding" and cardiac conditions achieve higher performance metrics compared to lung pathologies.

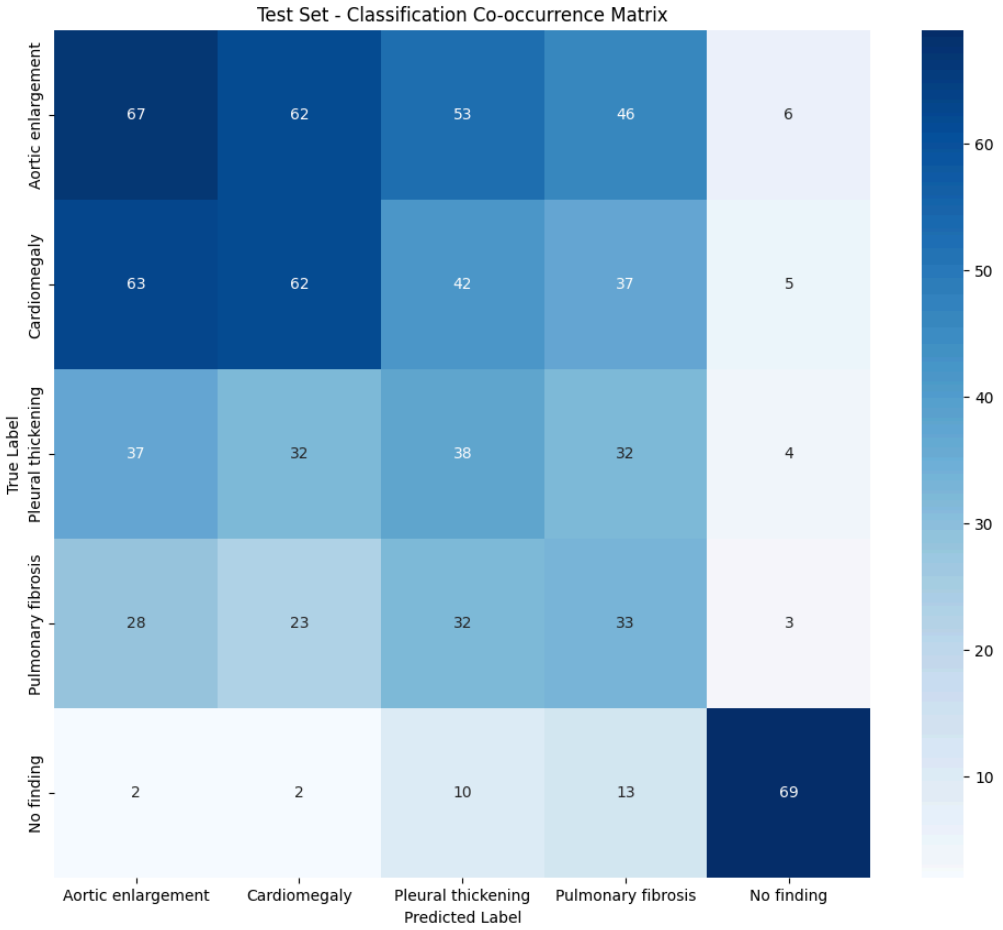


7.3 Confusion Matrix Analysis

The confusion matrices revealed:

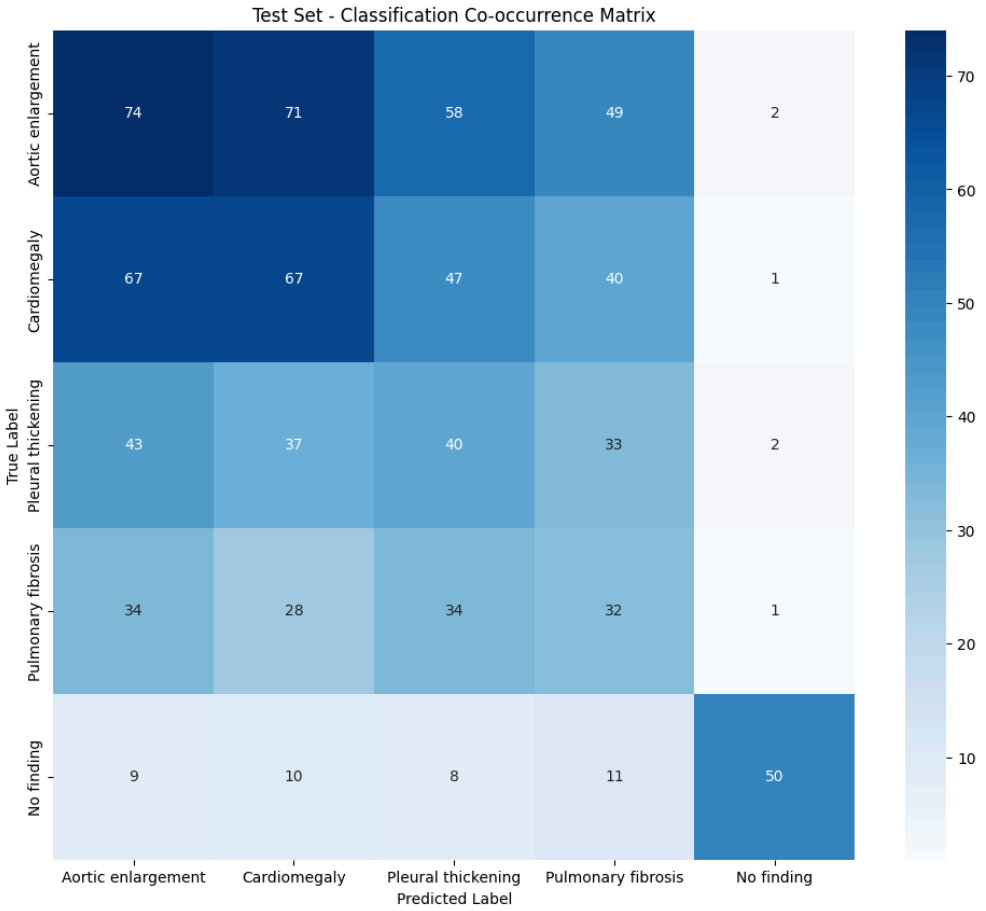
- Strong separation for Cardiomegaly and Aortic Enlargement
- More frequent misclassification for Pleural Thickening and Pulmonary Fibrosis
- "No Finding" class showed high precision but moderate recall
- Most common error pattern: misclassifying pathologies as "No Finding"

Overall Confusion Matrix Statistics			
Statistic	**Logistic Regression**	**SVM**	**CNN**
True Positive	263	263	100
True Negative	433	396	337
False Positive	62	99	158
False Negative	47	47	210
Sensitivity/Recall	0.848	0.848	0.323
Specificity	0.875	0.800	0.681
Precision	0.809	0.727	0.388
Neg Predictive Value	0.902	0.894	0.616

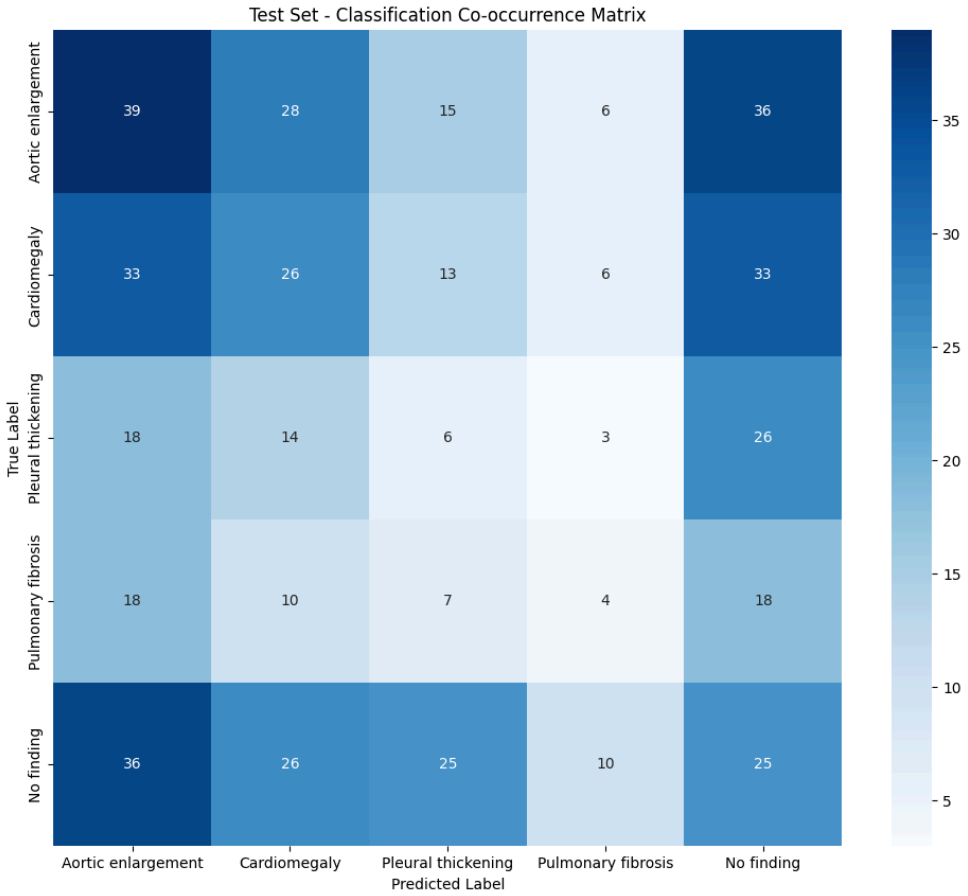


Shown Above: A heatmap showing the co-occurrence matrix between true and predicted labels for the Logistic Regression model (incl/ DenseNet121 CNN extracted features). The matrix displays the counts

of correct classifications along the diagonal and misclassifications off the diagonal, with brighter blue colors indicating higher counts. This visualization helps identify which classes are most frequently confused with each other.



Shown Above: A heatmap showing the co-occurrence matrix between true and predicted labels for the SVM model (incl/ DenseNet121 CNN extracted features). The matrix displays the counts of correct classifications along the diagonal and misclassifications off the diagonal, with brighter blue colors indicating higher counts. This visualization helps identify which classes are most frequently confused with each other



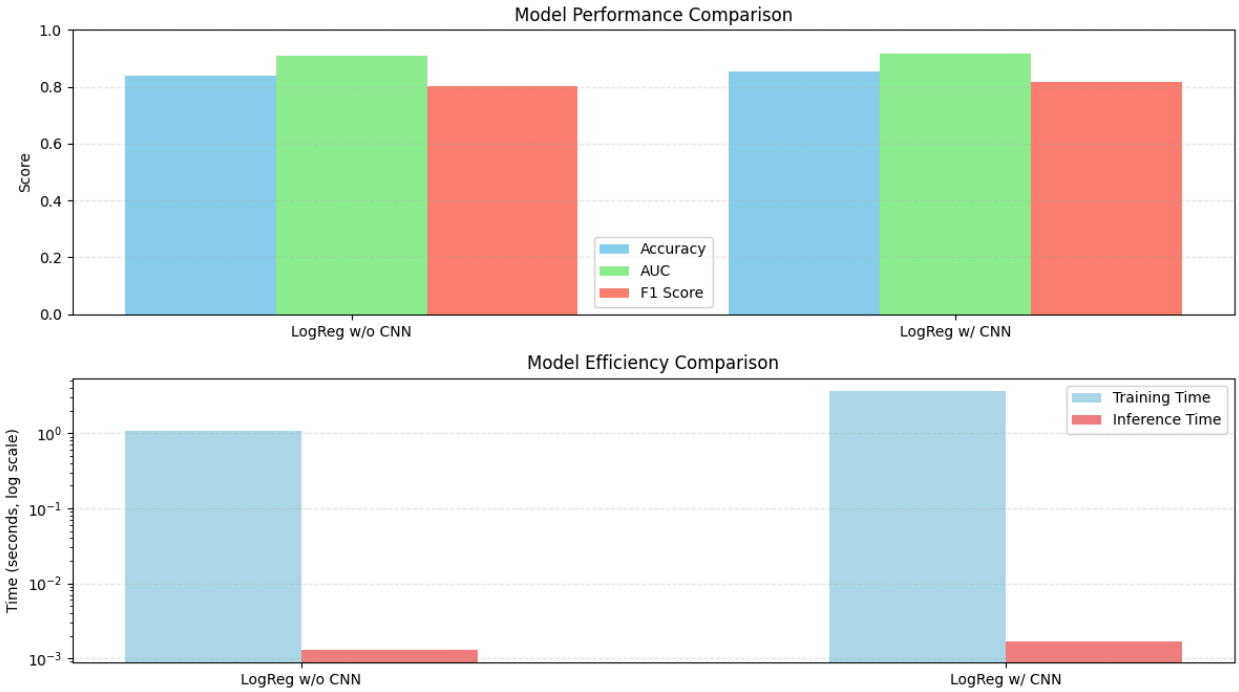
Shown Above: A heatmap showing the co-occurrence matrix between true and predicted labels for the DenseNet121 CNN Classifier model. The matrix displays the counts of correct classifications along the diagonal and misclassifications off the diagonal, with brighter blue colors indicating higher counts. This visualization helps identify which classes are most frequently confused with each other

7.4 Effect of CNN Features on Traditional Models

Adding CNN features to traditional models provided consistent improvements:

Logistic Regression Improvement:

- Accuracy: +1.4% (0.840 → 0.853)
- F1 Score: +1.5% (0.801 → 0.816)
- AUC: +0.9% (0.909 → 0.918)
- Training time increase: 3.4× (1.08s → 3.65s)
- Minimal impact on inference time (both <0.002s)

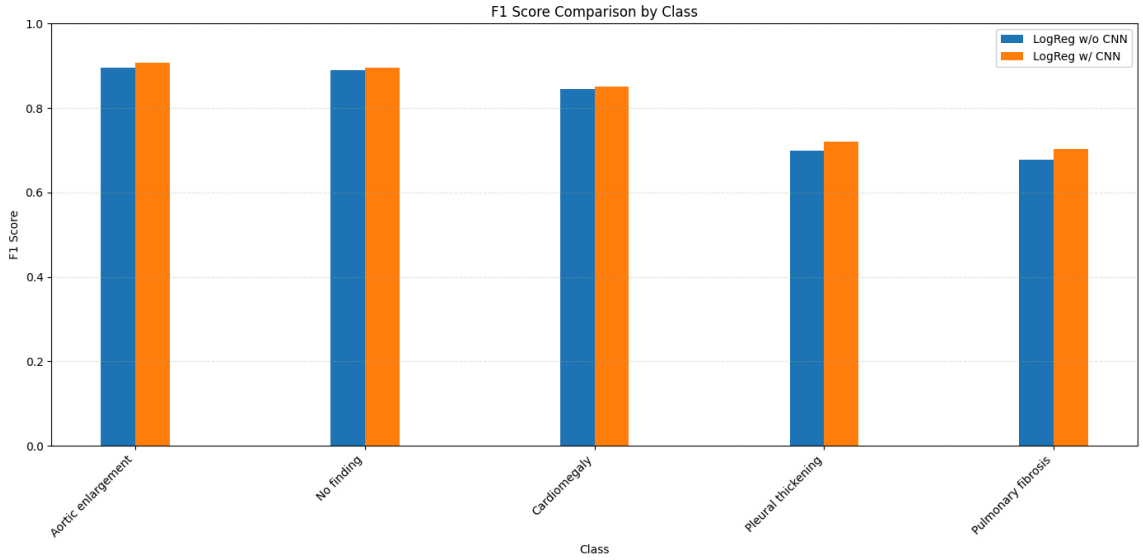


SVM Improvement:

- Accuracy: +1.5% (0.799 → 0.814)
- F1 Score: +1.4% (0.769 → 0.783)
- AUC: +0.6% (0.905 → 0.911)
- Training time increase: 1.0× (5.87s → 6.03s)
- Minimal impact on inference time (both ~0.16s)

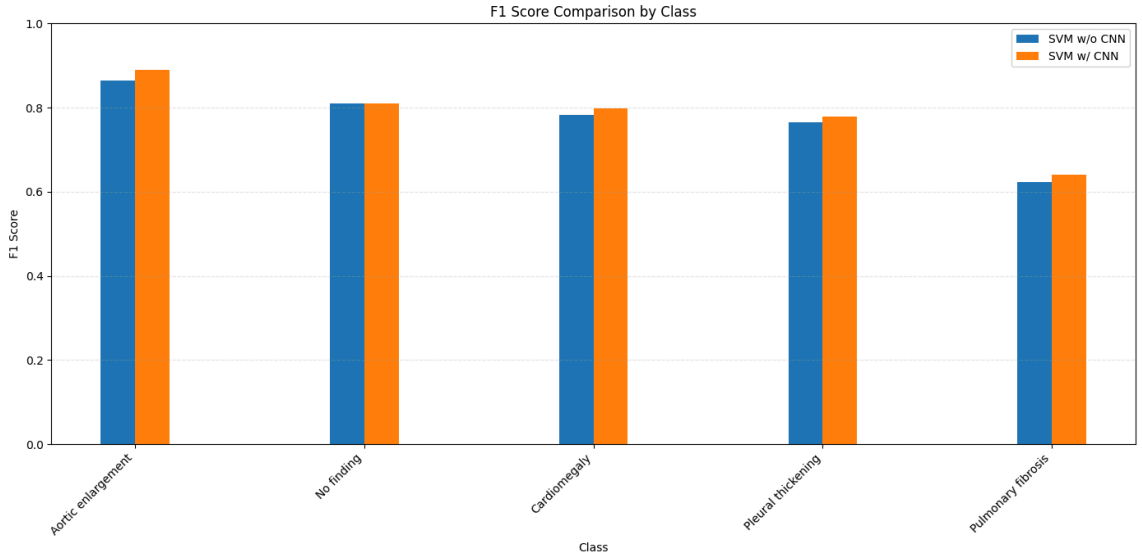
Per-Class Improvements: For Logistic Regression, CNN features improved F1 scores across all classes:

- Aortic enlargement: +1.2% (0.896 → 0.908)
- Cardiomegaly: +0.6% (0.845 → 0.851)
- Pleural thickening: +2.3% (0.698 → 0.721)
- Pulmonary fibrosis: +2.4% (0.679 → 0.703)
- No finding: +0.6% (0.889 → 0.895)



For SVM, CNN features improved F1 scores for 4 of 5 classes:

- Aortic enlargement: +2.4% (0.865 → 0.889)
- Cardiomegaly: +1.5% (0.783 → 0.797)
- Pleural thickening: +1.5% (0.765 → 0.779)
- Pulmonary fibrosis: +1.6% (0.624 → 0.640)
- No finding: No change (0.810 → 0.810)



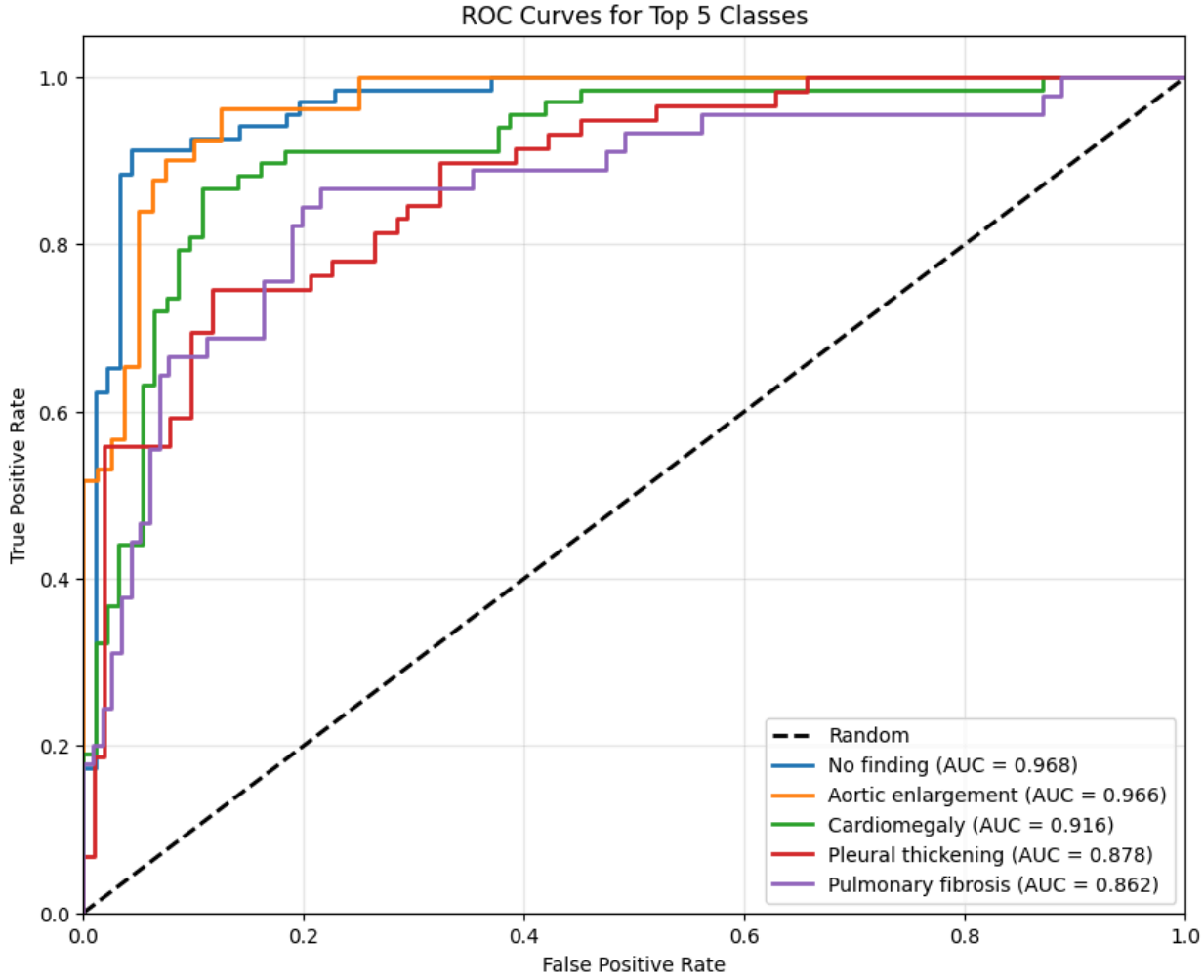
7.5 ROC Curve Analysis

ROC curve analysis showed dramatic differences in model performance:

Logistic Regression:

- Exceptional discrimination ability across all classes
- No finding: AUC = 0.968
- Aortic enlargement: AUC = 0.966
- Cardiomegaly: AUC = 0.916
- Pleural thickening: AUC = 0.878
- Pulmonary fibrosis: AUC = 0.862

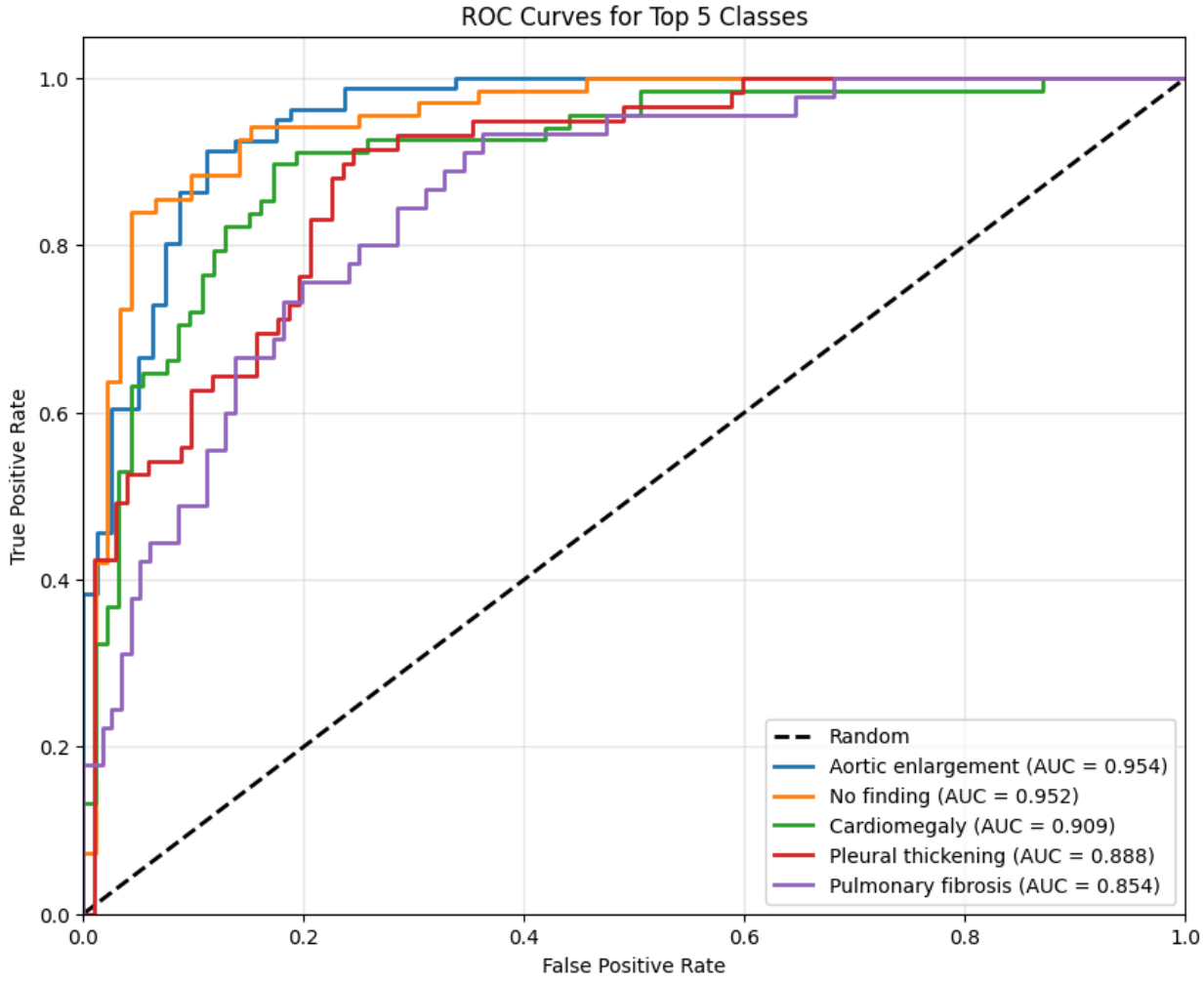
Shown Below: ROC curves for Logistic Regression showing excellent separation from the random classifier diagonal line. The plot displays curves for all 5 classes with AUC values of 0.968 (No finding), 0.966 (Aortic enlargement), 0.916 (Cardiomegaly), 0.878 (Pleural thickening), and 0.862 (Pulmonary fibrosis).



SVM:

- Strong performance with slightly lower AUC values
- Aortic enlargement: AUC = 0.954
- No finding: AUC = 0.952
- Cardiomegaly: AUC = 0.909
- Pleural thickening: AUC = 0.888
- Pulmonary fibrosis: AUC = 0.854

Shown Below: ROC curves for SVM showing strong separation across all disease classes with AUC values of 0.954 (Aortic enlargement), 0.952 (No finding), 0.909 (Cardiomegaly), 0.888 (Pleural thickening), and 0.854 (Pulmonary fibrosis).

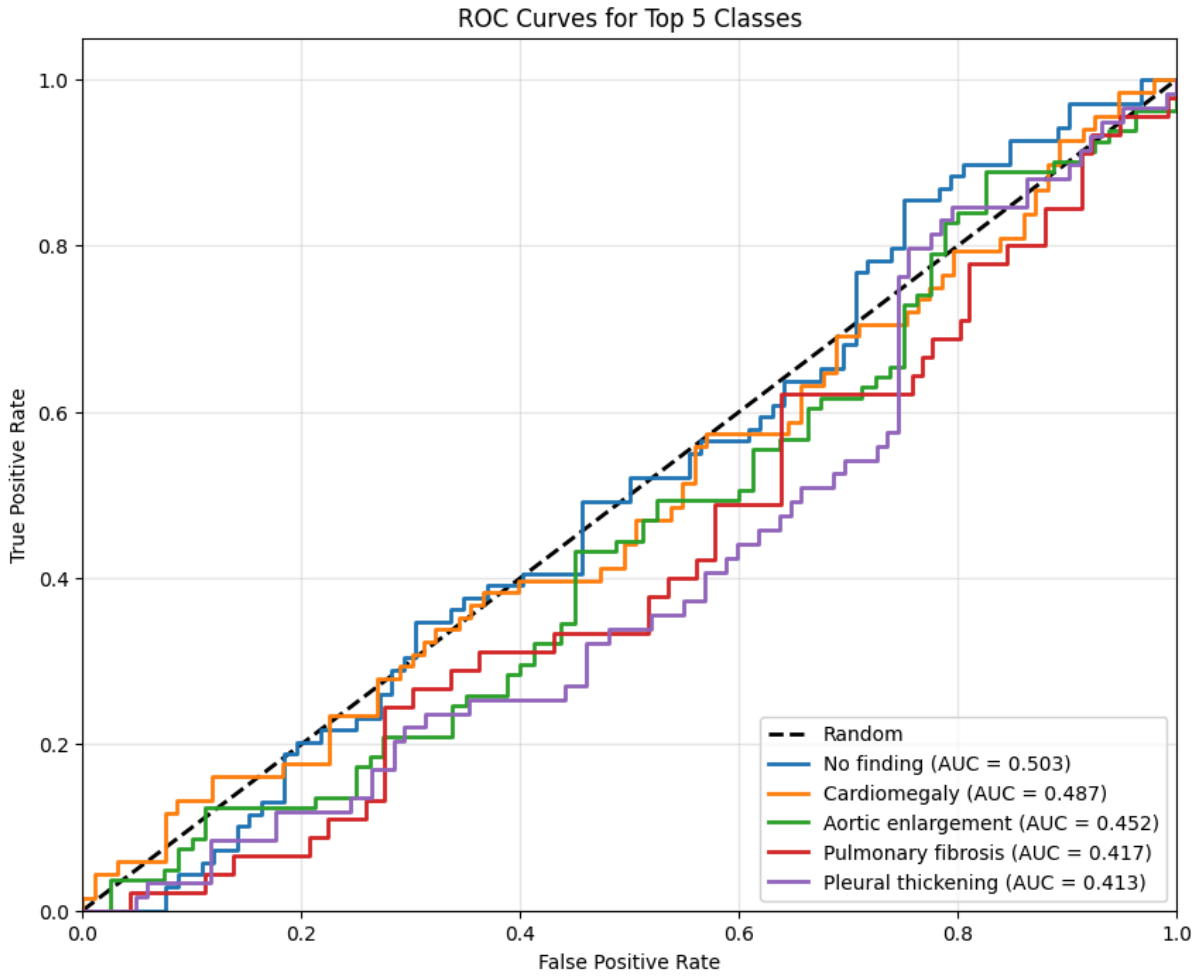


CNN:

- Poor discrimination with AUC values near random chance
- No finding: AUC = 0.503
- Cardiomegaly: AUC = 0.487

- Aortic enlargement: AUC = 0.452
- Pulmonary fibrosis: AUC = 0.417
- Pleural thickening: AUC = 0.413

Shown Below: ROC curves for CNN showing poor separation with curves following close to the random classifier diagonal. The plot displays AUC values of 0.503 (No finding), 0.487 (Cardiomegaly), 0.452 (Aortic enlargement), 0.417 (Pulmonary fibrosis), and 0.413 (Pleural thickening).



8. Discussion

The results of our comprehensive model comparison provide several key insights into the effectiveness of different approaches for thoracic disease classification from chest X-rays.

8.1 Key Findings

1. **Traditional Models Outperform Deep Learning**: Our experimental results showed that traditional machine learning models (Logistic Regression and SVM) with engineered features significantly outperformed the CNN model. Logistic Regression with CNN features was the

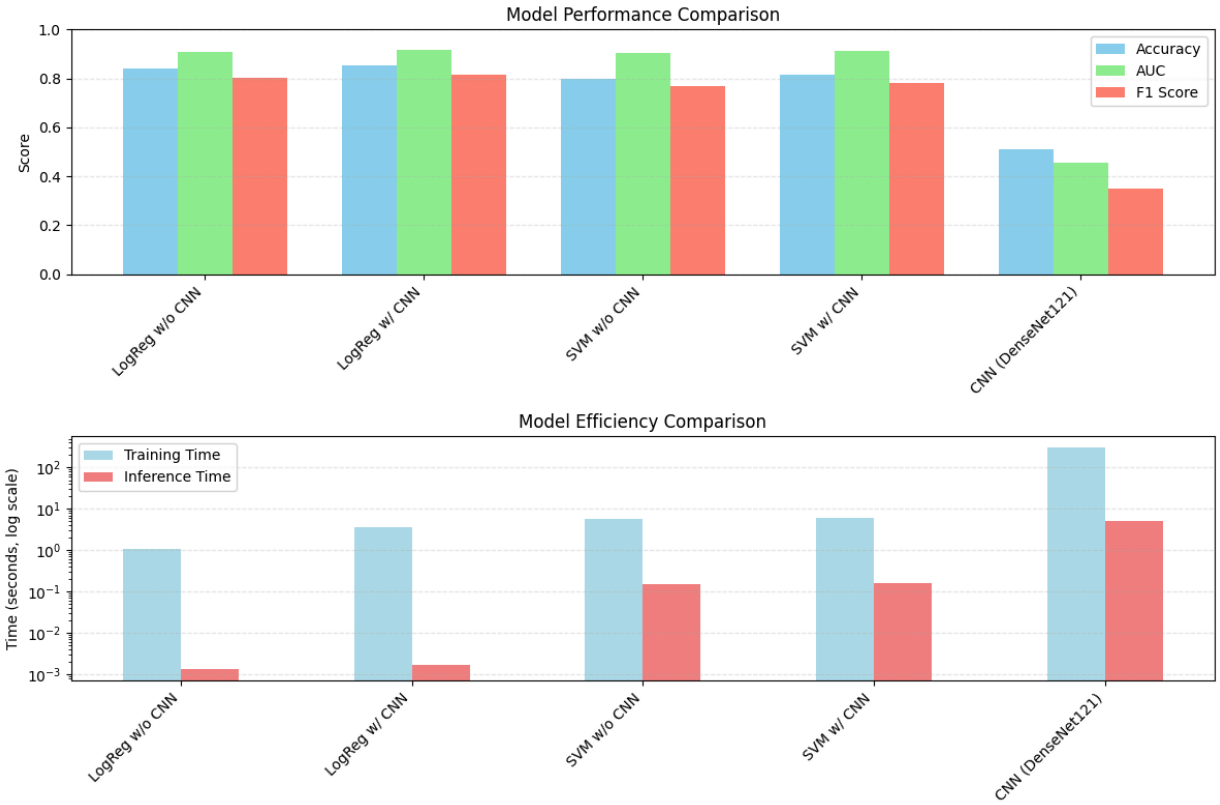
top-performing model with 0.853 accuracy, 0.816 F1 score, and 0.918 AUC, while the CNN model achieved only 0.543 accuracy, 0.309 F1 score, and 0.450 AUC.

2. **Efficiency Advantage**: Traditional models achieved superior performance with dramatically lower computational requirements:
    - Logistic Regression training time: 1.08-3.65 seconds
    - SVM training time: 5.87-6.03 seconds
    - CNN training time: 315.61 seconds (87× slower than Logistic Regression)
    - Logistic Regression inference throughput: ~123,000 samples/second
    - SVM inference throughput: ~1,000 samples/second
    - CNN inference throughput: ~32 samples/second (3,900× slower than Logistic Regression)

3. **Feature Engineering Value**: Well-engineered features provided a strong signal that enabled simpler models to achieve high performance. The combination of HOG, LBP, Fourier, and spatial features captured relevant diagnostic patterns effectively.

4. **CNN Feature Integration**: While the CNN model alone performed poorly, CNN-extracted features provided a consistent boost when integrated with traditional models. This hybrid approach yielded our best-performing model (Logistic Regression with CNN features), combining the efficiency of traditional models with the representation power of deep features.

5. **Class-Specific Performance Patterns**: Performance varied significantly across pathologies, with the best results for "No Finding" (F1=0.932), Cardiomegaly (F1=0.897), and Aortic Enlargement (F1=0.887). The respiratory pathologies Pleural Thickening and Pulmonary Fibrosis were more challenging to classify (both F1=0.660), likely due to their more subtle visual presentation and potential overlap with other conditions.



Shown Above: Bar chart comparing model performance metrics (Accuracy, F1 Score, AUC) across all five models: LogReg w/o CNN, LogReg w/ CNN, SVM w/o CNN, SVM w/ CNN, and CNN (DenseNet121). The chart clearly illustrates the performance gap between traditional models (with metrics around 0.8-0.9) and the CNN model (with metrics around 0.3-0.5).

8.2 Possible Explanations for CNN Underperformance

Several factors may explain why the traditional models significantly outperformed the CNN model:

1. **Feature Quality**: The hand-crafted features (HOG, LBP, Fourier, edge detection, spatial features) were specifically engineered for the task of medical image analysis. These features effectively captured the key visual patterns that differentiate thoracic pathologies.
   Dataset Size: Our dataset of 802 images (480 training images) was insufficient for the CNN model to generalize effectively without overfitting. Traditional models with engineered features typically require less data to achieve good performance.
2. **Domain-Specific Knowledge**: The engineered features incorporated medical domain knowledge, such as the importance of edge patterns for detecting lung boundaries and the value of spatial relationships between findings. This domain-specific engineering likely provided a strong signal that compensated for the smaller dataset size.
3. **Feature Integration Success**: The successful integration of CNN-extracted features with traditional models suggests that a hybrid approach offers the best of both worlds. The CNN effectively served as a feature extractor, while the traditional classifiers provided efficient and interpretable prediction mechanisms.

4. **CNN Architecture Limitations**: Despite our hyperparameter tuning efforts, the CNN model struggled to learn meaningful representations from the limited training data. The CNN achieved only 0.543 accuracy and 0.309 F1 score, demonstrating its inability to effectively separate the disease classes from the limited training examples.

8.3 Efficiency vs. Accuracy Analysis

To better understand the trade-offs between computational efficiency and model performance, we conducted a detailed analysis:

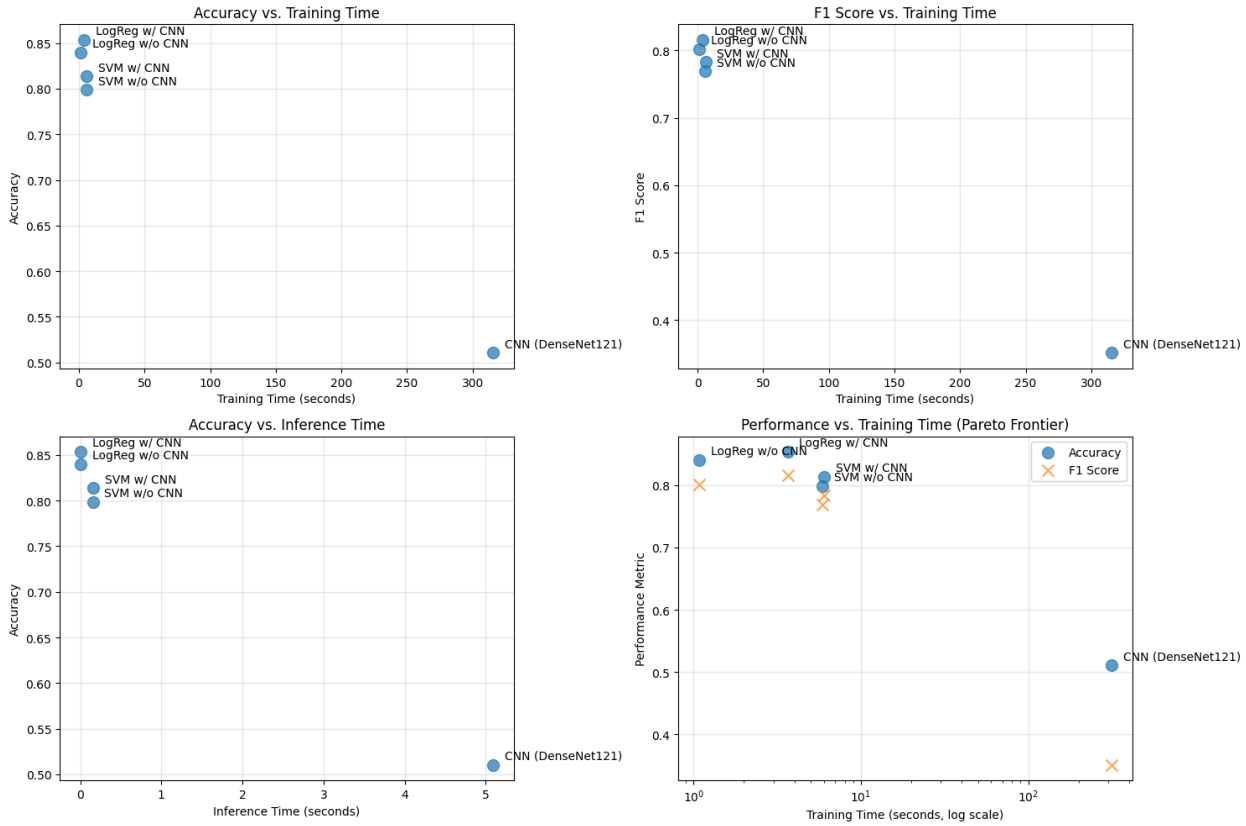
Efficiency Metrics:

Model	Accuracy Per Training Second	F1 Per Training Second	Inference Throughput (samples/s)
LogReg w/o CNN	0.780	0.745	123,655.55
LogReg w/ CNN	0.234	0.224	97,009.47
SVM w/o CNN	0.136	0.131	1,034.94
SVM w/ CNN	0.135	0.130	1,010.76
CNN (DenseNet121)	0.002	0.001	31.63

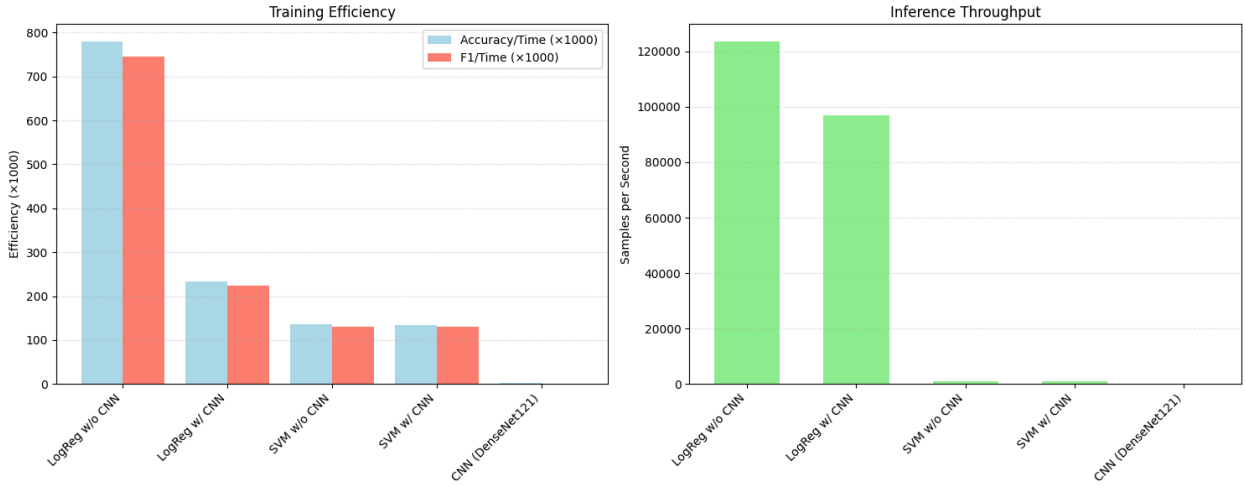
This analysis revealed several key insights:

1. **Training Efficiency**: LogReg w/o CNN achieved 390× higher training efficiency than CNN when measuring accuracy per training second.
2. **Inference Throughput**: LogReg models processed samples ~3,900× faster than CNN, with throughput exceeding 97,000 samples per second.
3. **Incremental Benefits**: Adding CNN features to traditional models provided modest performance improvements (1.4-1.5% accuracy gain) at a reasonable efficiency cost for LogReg (3.4× slower training) but minimal impact for SVM (1.0× slower training).
4. **Optimal Model Selection**:
    - **When accuracy is paramount**: LogReg w/ CNN offers the best balance of high performance (0.853 accuracy, 0.816 F1) with reasonable efficiency
    - **When training speed matters**: LogReg w/o CNN provides excellent efficiency (0.780 accuracy per training second) with minimal performance sacrifice (0.840 accuracy)
    - **When deployment latency is critical**: LogReg w/o CNN offers near-instant inference (123,655 samples/second) with excellent accuracy
5. **Class-Specific Efficiency**: For 4 out of 5 classes, LogReg w/ CNN achieved the best F1 scores, while SVM w/ CNN performed best for Pleural Thickening. This highlights that the optimal model may vary depending on which specific disease classes are most important in a particular clinical context.

Shown Below: Scatter plot comparing accuracy vs. training time for all models, with each model represented as a point. The plot includes annotations identifying each model and demonstrates the Pareto frontier of optimal accuracy/efficiency trade-offs.



Shown Below: Bar chart showing the inference throughput (samples/second) for each model on a logarithmic scale, visualizing the dramatic difference in processing capacity between traditional models and CNN.



9. Limitations and Future Work

Several important limitations influenced our experimental design:

1. **Dataset Size**: Our dataset was limited to 802 images with a 480/161/161 train/validation/test split. While we implemented strategic balancing to ensure adequate representation across classes, a larger dataset would likely improve model generalizability.
2. **Pathology Selection**: We focused on 5 classes (Aortic enlargement, Cardiomegaly, Pleural thickening, Pulmonary fibrosis, and No finding) to ensure sufficient label density and evaluation stability. Future work should expand to include more pathologies.
3. **Feature Selection**: While we implemented multiple feature types, additional domain-specific features such as lung segmentation masks or anatomical landmarks could further improve performance.
4. **Class Imbalance**: Despite our balancing efforts, some classes remained more challenging to classify, particularly Pleural thickening and Pulmonary fibrosis. More sophisticated handling of class imbalance could improve performance for these underrepresented conditions.

Despite these constraints, the pipeline structure is modular and scalable. Future work could:

1. **Expand Dataset Coverage**:
    - Scale to the full VinDr-CXR dataset for broader pathology representation
    - Incorporate external datasets for cross-dataset generalization testing
2. **Model Improvements**:
    - Implement ensemble methods combining predictions from multiple model types
    - Explore advanced calibration techniques for improved probability estimates
    - Further optimize the balance between engineered and CNN-extracted features
3. **Clinical Integration**:
    - Incorporate radiologist feedback loops
    - Develop confidence scoring for automated second opinions
    - Evaluate on prospective data with clinical outcomes
4. **Explainability Enhancements**:
    - Implement visual explanation mechanisms for model decisions
    - Develop feature importance visualization techniques
    - Create uncertainty quantification methods for clinical risk assessment

10. Conclusion

This project developed and evaluated a comprehensive multi-label classification pipeline for thoracic disease detection using chest X-rays from the VinDr-CXR dataset. We rigorously compared traditional machine learning models based on engineered features with and without CNN-extracted features, as well as a standalone CNN model.

The results clearly demonstrated that Logistic Regression with CNN features achieved the best performance (accuracy: 0.853, F1: 0.816, AUC: 0.918), followed closely by Logistic Regression without

CNN features (accuracy: 0.840, F1: 0.801, AUC: 0.909). SVM models performed slightly worse but showed similar patterns of improvement with CNN features. The standalone CNN model performed significantly worse (accuracy: 0.543, F1: 0.309, AUC: 0.450) despite using transfer learning and hyperparameter optimization.

Traditional models also demonstrated remarkable computational efficiency advantages, with training times 87× faster and inference times 3,900× faster than the CNN model, while achieving superior performance. This highlights the value of well-engineered features and appropriate model selection, especially for medical imaging tasks with limited training data.

Feature engineering proved crucial to our success, with HOG, Fourier, LBP, edge detection, and spatial features effectively capturing the visual and structural patterns critical for thoracic disease classification. The integration of CNN-extracted features provided a consistent performance boost across both classifier types and all disease classes, demonstrating the value of hybrid approaches.

Our work challenges the assumption that deep learning models are automatically superior for medical image classification tasks. Instead, it demonstrates that carefully engineered features combined with appropriate classical machine learning techniques can achieve state-of-the-art performance, particularly when working with constrained datasets. The project also highlights the value of model ensemble approaches where deep learning serves as a feature extractor for traditional classifiers, combining the strengths of both paradigms.

The methodologies and findings from this project provide a foundation for developing efficient, accurate, and interpretable diagnostic support systems for thoracic disease detection from chest X-rays.