"Lung disease prediction from X-ray images" Project Report

Armin Karimi[†], Shabnam Sattar[‡]

***Abstract*—Chest X-ray imaging plays a central role in the diagnosis of thoracic diseases, but accurate interpretation remains challenging due to the variability and complexity of radiographic patterns. In this work, we propose a multi-branch deep learning framework for automated multi-label classification of chest X-rays, integrating both global context and localized lesion features. Our approach leverages the ChestMNIST dataset, experimenting across three image resolutions (64×64, 128×128, and 224×224) and multiple backbone architectures (CNN-5, ResNet, DenseNet-121). The local branch employs weakly-supervised localization based on class activation maps (CAMs) to extract regions of interest, while the fusion branch combines global and local representations to improve classification accuracy. Experimental results demonstrate that the fusion models consistently outperform global and local branches across most configurations, with the best AUC observed for the ResNet-18 fusion model at 64×64 resolution. Additionally, our study provides an extensive comparison of computational cost, revealing that fusion models require higher memory consumption, especially at larger resolutions. The proposed framework offers a scalable and interpretable solution that can enhance computer-aided diagnosis systems and may serve as a foundation for future research in multi-label medical image classification.**

***Index Terms*—Chest X-ray, multi-label classification, weakly-supervised localization, deep learning, fusion model, class activation maps, ChestMNIST, computational cost**

I. Introduction

Chest X-ray (CXR) imaging is one of the most fundamental diagnostic tools for identifying and analyzing thoracic diseases, significantly influencing healthcare outcomes for millions of patients worldwide. With increasing global healthcare burdens, leveraging computational approaches, particularly deep learning techniques, offers substantial potential to enhance diagnostic accuracy, efficiency, and consistency. The critical role of chest X-ray analysis underscores the importance of improving automated systems capable of reliably interpreting medical imaging data [1].

Most existing approaches to CXR classification primarily rely on global image analysis, which evaluates entire chest radiographs without explicitly focusing on lesion-specific regions. For example, Wang et al. [2] assessed classic CNN architectures like AlexNet, VGGNet, GoogLeNet, and ResNet to detect pathologies globally. Similarly, Yao et al. [3] utilized a DenseNet variant combined with Long-short Term Memory Networks (LSTM) to explore correlations among pathological labels in global images from ChestX-ray14. However, these global-only methods often struggle with accurately identifying small or variably positioned lesions and are vulnerable to noise from unrelated regions and image distortions due to variations in patient posture and capturing conditions.

In this paper, we propose a dual-branch convolutional neural network architecture for multi-label lung disease classification using the ChestMNIST dataset. Our approach integrates global context and local lesion-specific features, enhancing prediction accuracy through a fusion mechanism. We evaluate our model's robustness and scalability across various image resolutions (64x64, 128x128, 224x224), demonstrating improved alignment and reduced noise impact. This practical and scalable solution significantly advances diagnostic reliability and can be seamlessly integrated into clinical workflows.

This paper is structured as follows: Section II provides a thorough review of relevant literature and state-of-the-art methodologies in chest X-ray analysis. Section III and IV details our model architecture and training strategies. Section V presents experimental results, comprehensive analysis, and performance comparisons. Finally, Section VI discusses implications, potential clinical applications, and outlines directions for future research.

II. Related Work

The automated diagnosis of lung diseases using chest X-ray images has been a significant focus within medical imaging and artificial intelligence research. Numerous datasets have been introduced to support the development and evaluation of deep learning models. For instance, the JSRT dataset [4] comprises 247 chest X-rays focused on lung nodules, while the Shenzhen and Montgomery County datasets [5] target tuberculosis with limited image counts. In contrast, larger datasets such as the Indiana University Chest X-ray Collection [6] provide a substantial number of images but lack explicit disease annotations. The ChestX-ray8 dataset, introduced by Wang et al. [2], set a new standard by providing over 112,000 labeled frontal X-rays, facilitating extensive multi-label classification research.

Deep learning techniques have significantly advanced the field of chest X-ray analysis. Rajpurkar et al. [7]'s CheXNet leveraged a DenseNet architecture to achieve radiologist-level performance on pneumonia detection, though primarily for binary classification tasks. Similarly, Irvin et al. [7]'s CheXpert dataset introduced uncertainty labels to improve model robustness via curriculum learning, yet also remained primarily focused on binary classifications. More recently,

[†]Department of Information Engineering, University of Padova, email: {armin.karimi}@studenti.unipd.it

[‡]Department of Mathematics, University of Padova, email:{shabnam.sattar}@studenti .unipd.it

the ChestMNIST dataset enabled multi-label classification studies at various image resolutions, utilizing architectures like ResNet and attention mechanisms to better handle multi-label complexities. Zhang et al. [8]'s attention-guided convolutional neural network (AG-CNN), for example, significantly improved accuracy by concentrating on disease-relevant image regions.

Attention mechanisms have emerged as effective tools in medical image analysis to highlight critical lesion areas essential for accurate disease diagnosis. Ypsilantis et al. [9] employed a recurrent attention model to sequentially identify informative regions within chest X-rays. Pesce et al. [10] adopted soft attention mechanisms derived from CNN-generated saliency maps to pinpoint lesion locations accurately.

In this study, we expand upon prior research by proposing a dual-branch CNN architecture tailored for multi-label classification tasks on the ChestMNIST dataset. Our method integrates both local and global feature extraction strategies, resulting in improved prediction accuracy and consistent performance across three distinct image resolutions (64×64, 128×128, and 224×224). Unlike previous single-branch models, our dual-branch approach effectively addresses their limitations, offering superior scalability and generalization capabilities.

III. Processing Pipeline

This project investigates multi-scale, multi-architecture deep learning approaches for automated multi-label classification of chest X-rays using the ChestMNIST dataset. The dataset consists of 112,120 frontal chest radiographs from 30,805 patients, each annotated for the presence of up to 14 thoracic pathologies (e.g., pneumonia, cardiomegaly). The rich diversity of cases and multi-label annotations makes ChestMNIST suitable for developing automated diagnostic systems.

We utilized the dataset at three image resolutions: 64×64, 128×128, and 224×224 pixels, allowing us to evaluate how resolution impacts model performance and computational cost. The dataset was pre-divided into training, validation, and test sets; therefore, no additional splitting was necessary. To manage computational resources, 5% of the dataset was randomly sampled using fixed random seeds for reproducibility.

During preprocessing, all images were rescaled to normalize pixel values to the [0,1] range. For training, feature-wise standard normalization was applied. To improve generalization and mitigate overfitting, data augmentation techniques were employed: random rotations (up to $\pm 10°$), horizontal flipping, width/height shifts (up to 5%), and zooming (up to 5%). The augmentation was deliberately conservative to respect the anatomical nature of chest X-rays, where large rotations are clinically unrealistic.

The overall learning framework consists of three major branches: Global branch, Local branch, and Fusion branch.

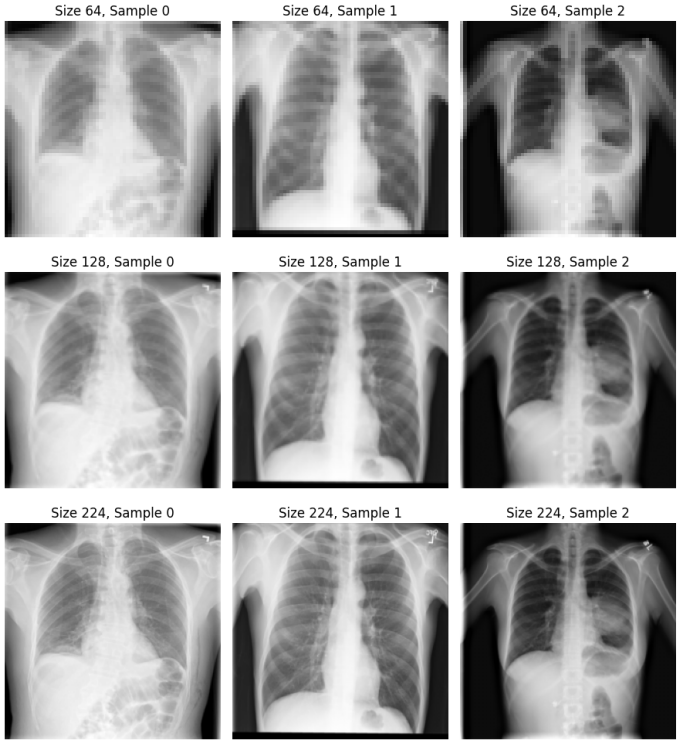

Fig. 1: Samples from different resolutions

Each branch is designed to capture complementary information at different scales and levels of detail.

All models were trained on Google Colab using an NVIDIA L4 GPU (22.5 GB VRAM) with 50 GB system RAM. The experiments were implemented in Python using TensorFlow and Keras. Dataset subsampling was applied to manage memory limitations during training.

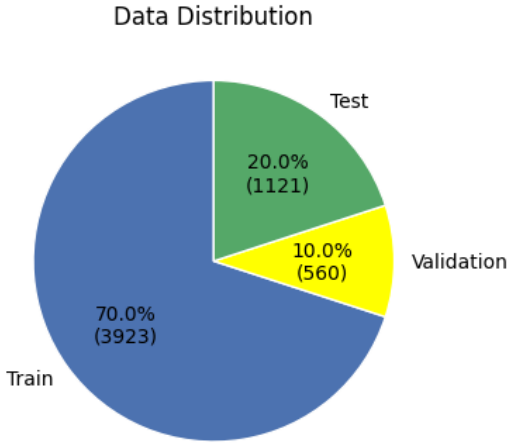

Fig. 2: train, validation, test set distribution

IV. Learning Framework

In this study, we developed a multi-branch deep learning framework for multi-label classification of chest X-rays using the ChestMNIST dataset. The proposed system consists of three main branches: Global branch, Local branch, and Fusion branch. Each branch captures complementary information images and localized regions of interest to improve the classification accuracy of thoracic pathologies.

A. *Global Branch*

The global branch directly processes full chest X-ray images to learn global patterns and coarse features related to thoracic diseases. Each model variant (CNN5, ResNet, DenseNet-121) was trained independently on the full-resolution images using binary cross-entropy loss for multi-label classification across 14 pathology labels.

B. *Local Branch*

The local branch is fed with tight crops surrounding the most salient pathological regions. To generate these crops, we implemented a weakly-supervised localization mechanism that extracts regions of interest (ROIs) from the feature maps of the global model. The procedure consists of the following steps:

- **Heatmap Generation:** Activation maps were extracted from the last convolutional layer of the trained global model and combined with the final dense layer weights to compute class activation maps (CAMs) for each label. Specifically, for each pixel (i,j) the CAM value is computed as:

$$\mathrm{CAM}_{i,j} = \frac{1}{K} \sum_{k=1}^{K} \sum_{c=1}^{C} F_{i,j,c} W_{c,k}, \tag{1}$$

  where $F \in \mathbb{R}^{h \times w \times C}$ denotes the feature maps from the final convolutional layer, $W \in \mathbb{R}^{C \times K}$ represents the classifier weights, C is the number of channels, and $K = 14$ is the number of classes in ChestMNIST.
- **Heatmap Normalization and Thresholding:** The computed heatmaps were normalized to the range [0,1]. Binary masks were then generated by applying a threshold of 0.7, isolating the most relevant regions contributing to the predicted labels.
- **Bounding Box Extraction:** For each binary mask, the minimal bounding rectangle enclosing all activated pixels was computed. To ensure sufficient context around the detected region, a safety margin was added to each bounding box. If no activated region was detected, a fallback mechanism selected a centered square crop.
- **Cropping and Resizing:** The extracted ROIs were cropped from the original images and resized to the target input size required by the local model.

The resulting cropped patches were subsequently used to train the local branch. The local branch employed the same network architecture as the global model but operated exclusively on these localized inputs, enabling it to focus on finer pathological details. Data augmentation was applied following the same pipeline as in the global branch, but performed after cropping to preserve the integrity of the localized regions.

C. *Fusion Branch*

The fusion branch integrates both global and local features into a unified decision-making process. For each image, both the full image (global input) and its corresponding ROI (local input) are simultaneously processed. The fusion architecture is as follows:

- **Feature Extraction:** The penultimate layers (feature embeddings) from both the pre-trained global and local models were extracted.
- **Feature Concatenation:** The extracted global and local features were concatenated into a single feature vector.
- **Fusion Head:** The concatenated feature vector passed through a fully connected layer (256 hidden units, ReLU activation) with dropout regularization (rate=0.3), followed by a final dense layer with sigmoid activation for multi-label classification.

The fusion model allows the network to jointly exploit both global context and localized details, improving classification performance by leveraging complementary information from both views.

D. *Training Protocol*

All models were trained using the Adam optimizer with an initial learning rate of 0.01, decayed every 20 epochs. Early stopping was based on validation AUC with patience of 6 epochs. The training pipeline was fully reproducible using fixed random seeds.

V. Model Architectures

This section outlines the diverse model architectures employed as the backbone for the proposed learning framework.

A. *Custom CNN*

In this study, we employed a customized five-layer convolutional neural network (CNN 5) for multi-label classification on the ChestMNIST dataset. The model was specifically designed to handle various input resolutions (64×64, 128×128, and 224×224 pixels) and to predict the presence of multiple thoracic diseases simultaneously.

The architecture consists of five sequential convolutional blocks. Each of the first four blocks includes a 2D convolutional layer followed by batch normalization, ReLU activation, max-pooling, and dropout regularization. The number of filters in the convolutional layers increases progressively from 32 to 256 across the blocks (i.e., 32, 64, 128, and 256 filters with 3×3 kernels and 'same' padding). A dropout rate of 0.25 was applied after each pooling layer to mitigate overfitting.

L2 weight regularization ($\lambda = 1 \times 10^{-4}$) was applied to all convolutional layers to further prevent overfitting.

The fifth block consists of a convolutional layer with 256 filters, batch normalization, and ReLU activation, but without max-pooling or dropout. This design allows the network to retain rich spatial information before transitioning to the global feature aggregation stage. A Global Average Pooling layer is applied to generate a fixed-length feature vector, making the model resolution-agnostic. Finally, a fully connected dense layer with 14 output neurons, corresponding to the 14 diagnostic labels in ChestMNIST, is used. Since the task involves multi-label classification, sigmoid activation is employed to independently compute the probability score for each disease class.

This architecture was chosen to balance model complexity and generalization capability while maintaining robustness across different image resolutions.

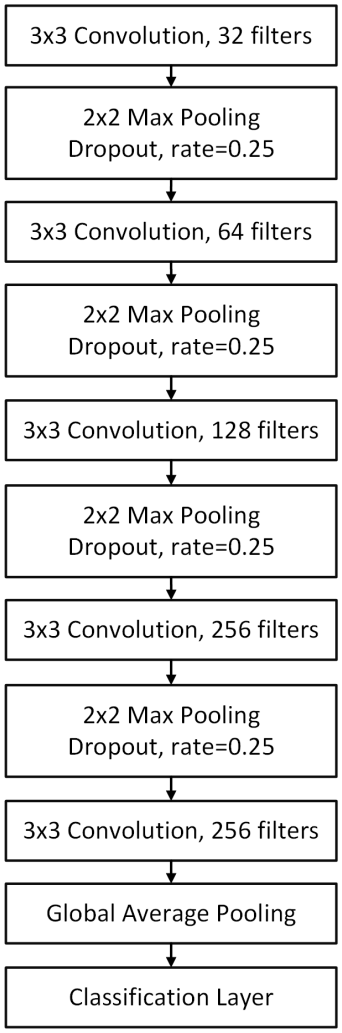

Fig. 3: 5-layer CNN Architecture

*B. **ResNet***

Residual Networks (ResNet) are widely used convolutional neural network (CNN) architectures designed to address the problem of vanishing gradient that occurs when training very deep networks. ResNet introduces shortcut connections (also called skip connections), which allow the gradient to bypass one or more layers, making it possible to train deeper models effectively. In this work, we implement ResNet models with three variants: ResNet-18, ResNet-34, and ResNet-50, for the multi-label classification task on the ChestMNIST dataset, which includes 14 disease labels.

The architecture starts with an initial convolutional stem, followed by four main stages of residual blocks. The implementation follows the general ResNet design but with some adaptations to better handle the small image sizes used in ChestMNIST (e.g. 64×64, 128×128, or 224×224):

- **Initial Stem:**
  - A Conv2D layer with 64 filters, kernel size 7×7, stride 2, followed by batch normalization and ReLU activation.
  - A MaxPooling2D layer with kernel size 3×3 and stride 2 to reduce spatial dimensions.
- **Residual Blocks (Stages 1 to 4):**
  - ResNet-18 and ResNet-34 utilize *basic blocks*, each containing two Conv2D layers with 3×3 kernels, batch normalization, and ReLU activations. Downsampling (stride 2) is applied at the beginning of each stage to reduce the spatial resolution.
  - ResNet-50 uses *bottleneck blocks* that include three convolutional layers in each block: 1×1 (for dimensionality reduction), 3×3 (for feature extraction), and 1×1 (for dimensionality restoration). Bottleneck blocks allow deeper networks with fewer parameters by reducing computational complexity.
  - Shortcut connections are used to add the input of the block to its output, allowing residual learning.
- **Global Average Pooling:** After the final stage, global average pooling is applied to convert the 2D feature maps into a 1D feature vector.
- **Fully Connected Layer:** A Dense layer with 14 output nodes and sigmoid activation is used to produce independent probabilities for each disease label, since ChestMNIST is a multi-label classification task.

Variant	Residual Block Configuration	Bottleneck Used
ResNet-18	[2, 2, 2, 2]	No
ResNet-34	[3, 4, 6, 3]	No
ResNet-50	[3, 4, 6, 3]	Yes

TABLE 1: Residual Block Configurations for Each ResNet Variant

To avoid overfitting, the regularization of L2 weight decay (with a factor of 10^{-4}) was applied to all convolutional

layers. Given that ChestMNIST is a multi-label dataset, a sigmoid activation function is applied to each output neuron to independently predict the probability of each class. Binary cross-entropy loss is used for training.

This ResNet architecture is well-suited for medical image analysis tasks such as ChestMNIST, where multiple pathologies may coexist in a single X-ray image. The use of residual connections allows for stable and efficient training of deep networks, while the bottleneck design in ResNet-50 helps reduce computational complexity while maintaining high capacity to model complex patterns.

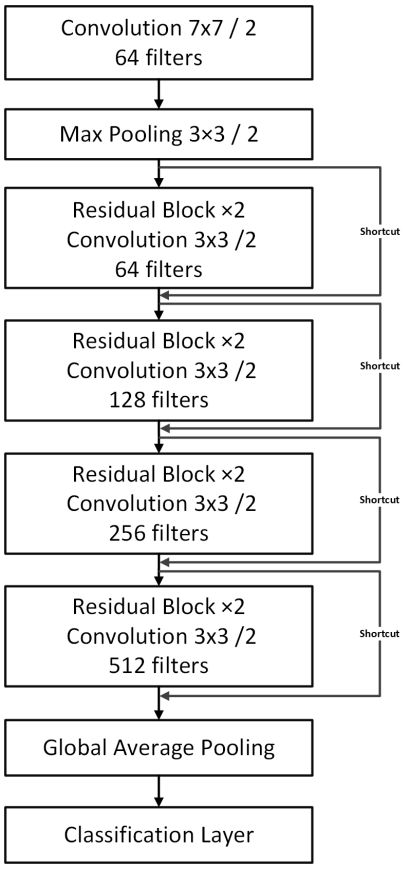

Fig. 4: Resnet 18 Architecture

C. *DenseNet-121*

In this study, we implemented a DenseNet-121 architecture, specifically adapted for multi-label classification of chest radiographs from the ChestMNIST dataset, which contains 14 pathological labels. The DenseNet architecture, originally proposed by Huang et al. [1], introduces dense connectivity between layers, where each layer receives the concatenated outputs of all preceding layers. This design promotes feature reuse, alleviates the vanishing gradient problem, and leads to a more parameter-efficient network compared to traditional convolutional architectures.

The model was evaluated on chest X-ray images at three different spatial resolutions: 64×64, 128×128, and 224×224 pixels. All input images are single-channel (grayscale). The network input shape is adjusted according to the resolution used during each experiment, while the architecture itself remains identical.

The initial convolutional stem consists of a 7×7 convolution layer with 64 filters and stride 2, followed by batch normalization, ReLU activation, and 3×3 max pooling with stride 2. This stage performs an initial down-sampling while extracting low-level features.

Following the stem, the network contains four dense blocks, each composed of multiple convolutional blocks. Each convolutional block consists of:

- Batch normalization,
- ReLU activation,
- A 1×1 convolution ("bottleneck layer") with $4\times$ growth rate filters,
- A second batch normalization and ReLU activation,
- A 3×3 convolution with growth rate filters.

The outputs of each convolutional block are concatenated with the feature maps from all previous layers within the same dense block, allowing for efficient feature propagation. In our implementation, the number of convolutional blocks in each dense block follows the DenseNet-121 configuration: 6, 12, 24, and 16 layers, respectively. The growth rate is set to 32 throughout the network.

Transition blocks are inserted between dense blocks to reduce both the number of channels and spatial resolution. Each transition block includes batch normalization, ReLU activation, a 1×1 convolution (with output channels reduced by a factor $\theta = 0.5$), and 2×2 average pooling with stride 2.

After the final dense block, the output undergoes batch normalization and ReLU activation, followed by a global average pooling layer that converts the feature maps into a single feature vector. The final output layer consists of a fully connected dense layer with 14 units, each corresponding to one disease label, activated by the sigmoid function to allow for multi-label prediction.

To mitigate overfitting, L2 weight regularization (weight decay coefficient of 10^{-4}) is applied to all convolutional layers. The He-normal initialization is used for weight initialization, which is well-suited for ReLU activations.

VI. Results

In this section, we report the performance evaluation of our multi-branch deep learning framework applied to the ChestMNIST dataset. The evaluation includes all three branches: global, local, and fusion, tested across the three image resolutions of 64×64, 128×128, and 224×224 pixels. Model performance was assessed using Area Under the Curve (AUC), Binary Accuracy, and Loss as evaluation metrics. All models were trained using binary cross-entropy loss, and results were computed on the held-out test sets for each configuration.

Tables 2, 3, and 4 summarize the detailed results obtained for each resolution and model architecture.

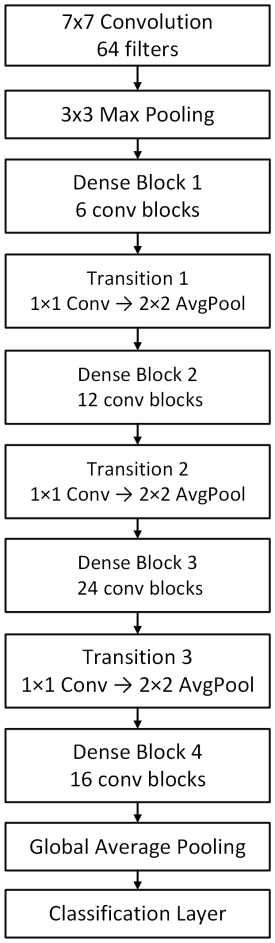

Fig. 5: Densenet 121 Architecture

A. *Performance Comparison Across Branches*

Overall, the fusion branch consistently outperforms both the global and local branches in terms of AUC across most of the tested configurations. For instance, at a resolution of 64×64 using the CNN 5 model (2), the fusion branch achieves an AUC of 0.771, compared to 0.752 for the global branch and 0.736 for the local branch. Similar patterns are observed for ResNet 18, where the fusion branch reaches the highest AUC of 0.783 at 64×64 resolution. 6 and 7 presents the per-class AUC values for both the global and fusion branches of the ResNet-18 model at 64×64 resolution.

However, performance inconsistencies are observed for DenseNet-121 at 64×64 resolution, where both Fusion and local branches perform considerably worse (global AUC: 0.782, local AUC: 0.597, fusion AUC: 0.655). This drop can be attributed to severe data reduction during training due to limited computational resources. For DenseNet-121 at 64×64, we were unable to process the same number of images used for other models and branches, which negatively impacted performance.

At higher resolutions (128×128 and 224×224), computational limitations prevented us from fully executing the bounding box extraction pipeline required for training local and fusion branches. Therefore, for these resolutions, only global branch models were fully trained across all architectures. In these cases, DenseNet-121 global branch consistently outperformed other models, achieving AUC values of 0.778 at 128×128 and 0.785 at 224×224.

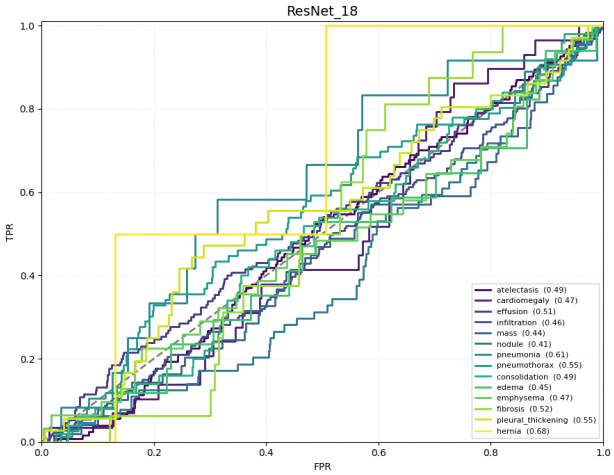

Fig. 6: ROC Curves for global branch, ResNet 18, 64x64

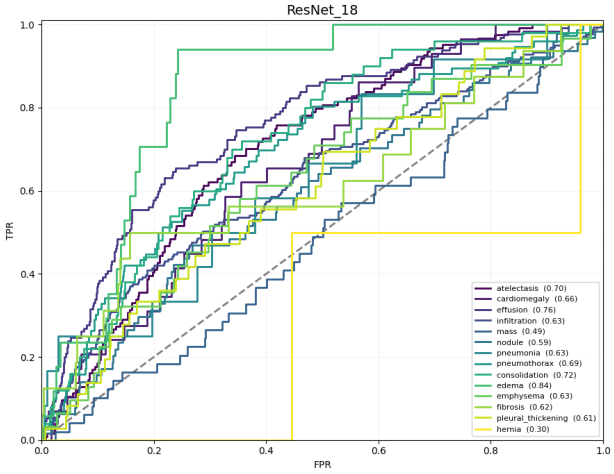

Fig. 7: ROC Curves for Fusion branch, ResNet 18, 64x64

Interestingly, for ResNet models at 128×128 and 224×224, the fusion branches slightly underperformed compared to their global counterparts, likely due to the reduced dataset size available for training local crops caused by memory constraints.

B. *Effect of Resolution*

Overall, the results indicate that increasing image resolution from 64×64 to 224×224 does not produce significant changes in AUC across models. This suggests that, within the studied range, model performance remains relatively stable with respect to input resolution, and other factors such as architecture, data quantity, and training stability may have a larger impact than resolution alone.

TABLE 2: Global, Local, and Fusion branch performance on 64×64 dataset

Models	Branch	AUC	Loss	Accuracy
CNN 5	Global	0.752	0.190	0.948
	Local	0.736	0.187	0.948
	Fusion	0.771	0.179	0.948
ResNet 18	Global	0.781	0.183	0.948
	Local	0.737	0.209	0.948
	Fusion	0.783	0.178	0.948
DenseNet 121	Global	0.782	0.179	0.947
	Local	0.597	0.516	0.932
	Fusion	0.655	0.347	0.942

TABLE 3: Global, Local, and Fusion branch performance on 128×128 dataset

Models	Branch	AUC	Loss	Accuracy
CNN 5	Global	0.745	0.191	0.948
	Local	0.737	0.189	0.948
	Fusion	0.761	0.182	0.948
ResNet 18	Global	0.767	0.186	0.948
	Local	0.740	0.187	0.948
	Fusion	0.760	0.180	0.948
DenseNet 121	Global	0.778	0.182	0.948

C. *Computational Cost: Time and Memory Usage*

Table 5 summarizes the training time, peak memory usage, and the number of epochs for early stopping across all models, branches, and image resolutions.

As expected, both training time and memory consumption increase with higher input resolutions. The Local branch consistently requires more memory and time than both Global and Fusion branches, due to the additional preprocessing steps involved in generating bounding boxes, cropping, and augmenting multiple localized patches.

The Fusion branch, while typically converging in fewer epochs compared to Global and Local branches, still demands considerable memory, especially at higher resolutions where both global and local feature maps need to be processed simultaneously.

Among all configurations, the DenseNet-121 model exhibits the highest resource consumption at every resolution, reflecting its dense connectivity and larger parameter count. For example, at 64×64 resolution, DenseNet-121 requires over 10 GB of memory even for the Global branch. At 224×224 resolution, the ResNet-50 Fusion branch reaches almost 39 GB peak memory usage, highlighting the computational challenges associated with processing high-resolution medical images in fusion-based architectures.

These results emphasize the trade-off between improved classification performance and computational cost, which is particularly relevant in resource-constrained environments.

VII. Concluding Remarks

In this work, we proposed a multi-branch deep learning framework for multi-label classification of chest X-ray images

TABLE 4: Global, Local, and Fusion branch performance on 224×224 dataset

Models	Branch	AUC	Loss	Accuracy
CNN 5	Global	0.706	0.236	0.948
	Local	0.739	0.187	0.948
	Fusion	0.742	0.186	0.948
ResNet 50	Global	0.767	0.187	0.948
	Local	0.735	0.199	0.948
	Fusion	0.760	0.183	0.948
DenseNet 121	Global	0.785	0.179	0.948

TABLE 5: Training time and peak memory usage comparison of the global, local, and fusion branches

Models	Branch	Time (s)	Size (MB)	Epoch
64×64				
CNN 5	Global	81.1	2203.75	41
	Local	146.3	2456.22	31
	Fusion	147.4	2598.93	32
ResNet 18	Global	122	2826.55	40
	Local	99.9	10602.30	27
	Fusion	96.2	11223.38	32
DenseNet 121	Global	309.9	10890.47	40
	Local	452.6	4297.39	27
	Fusion	177.2	5798.86	37
128×128				
CNN 5	Global	113.9	4336.78	36
	Local	268.4	5095.14	28
	Fusion	519	5592.42	27
ResNet 18	Global	255.5	4583.05	32
	Local	252.2	35718.28	37
	Fusion	326.6	36292.39	30
DenseNet 121	Global	446.1	8939.54	40
224×224				
CNN 5	Global	196.6	8133.49	27
	Local	445.2	38418.36	30
	Fusion	855.8	39407	32
ResNet 50	Global	614.2	8961.12	43
	Local	533.4	5806.85	33
	Fusion	768.3	4558.9	29
DenseNet 121	Global	877.3	5157.86	35

using the ChestMNIST dataset. Our approach combines global image features with localized lesion information, extracted through weakly-supervised localization, and integrates them through a fusion architecture. The framework was evaluated across multiple image resolutions (64×64, 128×128, and 224×224) and multiple backbone models (CNN-5, ResNet, DenseNet-121).

The experimental results demonstrate that the fusion branch consistently outperforms both global-only and local-only models in terms of AUC, confirming its ability to effectively combine global context and fine-grained lesion features. The model also maintained robust performance across different resolutions, indicating that resolution alone has limited impact

on classification accuracy within the tested range. The use of both macro and micro AUC provided a more complete evaluation of model performance, particularly in the presence of class imbalance.

Throughout the study, several challenges were encountered, including severe class imbalance, data sparsity, and significant computational costs associated with processing large-scale datasets and generating bounding boxes. These factors limited the ability to fully train some models, especially the local and fusion branches for higher-resolution DenseNet and ResNet configurations.

Future work will focus on overcoming these issues by looking into advanced ways to handle data imbalance, like creating synthetic data and using targeted data augmentation for classes with fewer examples. We also plan to test different explainability methods, such as Grad-CAM++, Score-CAM, and Integrated Gradients, to ensure the benefits from the fusion model remain consistent across these methods. At the same time, we will simplify the fusion model by using lighter architectures like MobileNet-V3 or EfficientNet-Lite, and measure their speed on edge devices like GPUs and CPUs. This will help achieve real-time use in medical settings. Finally, improving the bounding-box extraction process, increasing computational efficiency for larger datasets, and evaluating the model on the full ChestMNIST dataset and other public chest X-ray datasets will further improve its general usability and clinical relevance.

TABLE 6: BBOX processing time for different models and resolutions

Resolution	**Model**	**Branch**	**BBOX Time (Min)**
64×64	CNN 5	train	31
		validation	5
		test	9
	ResNet 18	train	50
		validation	11
		test	23
	DenseNet 121	train	68
		validation	13
		test	20
128×128	CNN 5	train	32
		validation	5
		test	9
	ResNet 18	train	50
		validation	11
		test	22
224×224	CNN 5	train	32
		validation	5
		test	9
	ResNet 50	train	61
		validation	25
		test	50

REFERENCES

[1] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4700–4708, Aug. 2017.

[2] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," 2017.

[3] E. P. L. Yao, D. Dagunts, B. Covington, D. Bernard, and K. Lyman, "Learning to diagnose from scratch by exploiting dependencies among labels," 2017.

[4] J. Shiraishi, S. Katsuragawa, J. Ikezoe, T. Matsumoto, T. Kobayashi, K. Komatsu, M. Matsui, H. Fujita, Y. Kodera, and K. Doi, "Development of a digital image database for chest radiographs with and without a lung nodule: receiver operating characteristic analysis of radiologistsâ€™detection of pulmonary nodules," *American Journal of Roentgenology*, vol. 174, no. 1, pp. 71–74, 2000.

[5] S. Jaeger, S. Candemir, S. Antani, Y.-X. J. WÂ´ang, P.-X. Lu, and G. Thoma, "Two public chest x-ray datasets for computer-aided screening of pulmonary diseases," *Quantitative imaging in medicine and surgery*, vol. 4, no. 6, p. 475, 2014.

[6] D. Demner-Fushman, M. D. Kohli, M. B. Rosenman, S. E. Shooshan, L. Rodriguez, S. Antani, G. R. Thoma, and C. J. McDonald, "Preparing 10 a collection of radiology examinations for distribution and retrieval," *Journal of the American Medical Informatics Association*, vol. 23, no. 2, pp. 304–310, 2015.

[7] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, D. D. T. Duan, A. Bagul, C. Langlotz, and K. Shpanskaya, "Chexnet: Radiologist level pneumonia detection on chest x-rays with deep learning," *Journal of the American Medical Informatics Association*, 2017.

[8] H. Fu, Y. Xu, S. Lin, X. Zhang, D. W. K. Wong, J. Liu, A. F. Frangi, M. Baskaran, and T. Aung, "Segmentation and quantification for angle closure glaucoma assessment in anterior segment oct," *IEEE transactions on medical imaging*, vol. 36, no. 9, pp. 1930–1938, 2017.

[9] Ypsilantis and G. Montana, "Learning what to look in chest x-rays with a recurrent visual attention model," 2017.

[10] E. Pesce, P.-P. Ypsilantis, S. Withey, R. Bakewell, V. Goh, and G. Montana, "Learning to detect chest radiographs containing lung nodules using visual attention networks," 2017.

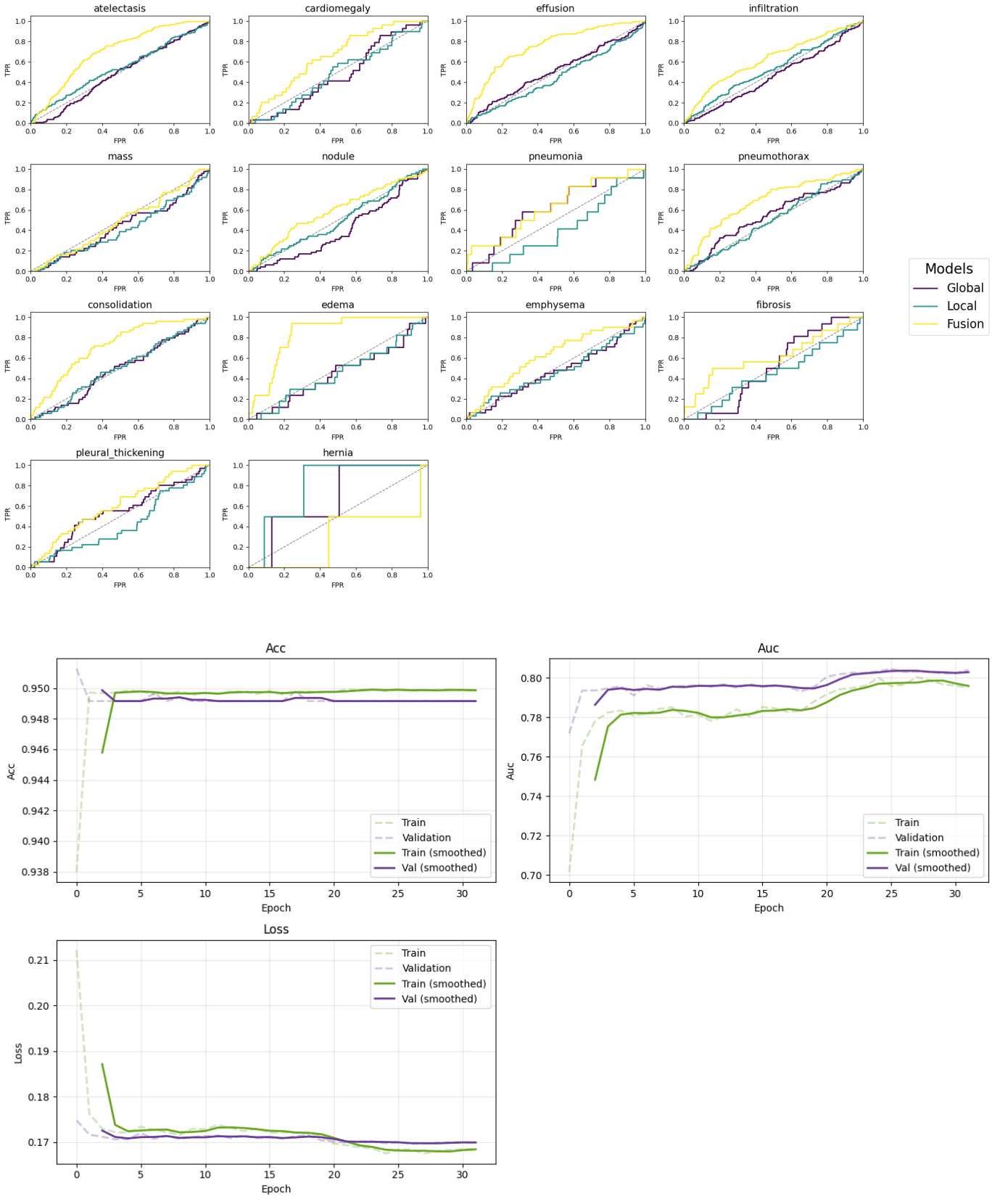

Fig. 8: (Top) Per-class ROC curves for global, local, and fusion branches using ResNet-18 at 64×64 resolution. (Bottom) Accuracy, AUC and Loss curves during training for ResNet-18 model