# Involution Meets Sparse Attention:A Federated Learning Based Frontier in Medical Image Classification

Nafisa Khan Youkee
21301499
Azwad Aziz
21301027
Fariha Shams Ahmed
21301023
Sandia Tasnim
21301383

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
BRAC University
February 2025

# Declaration

It is hereby declared that

- The thesis submitted is our own original work while completing degree at BRAC University.
- The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
- The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
- We have acknowledged all main sources of help.

**Student's Full Name and Signature:**

| | |
|---|---|
| Nafisa Khan Youkee<br>21301499 | Azwad Aziz<br>21301027 |
| Fariha Shams Ahmed<br>21301023 | Sandia Tasnim<br>21301383 |

# Approval

The thesis titled "Involution Meets Sparse Attention:A Federated Learning Based Frontier in Medical Image Classification" submitted by

1. Nafisa Khan Youkee (21301499)
2. Azwad Aziz (21301027)
3. Fariha Shams Ahmed (21301023)
4. Sandia Tasnim (21301383)

Of Fall, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on February 03, 2025.

**Examining Committee:**

Supervisor:
(Member)

Amitabha Chakrabarty, PhD

Professor
Department of Computer Science and Engineering
BRAC University

Co-Supervisor:
(Member)

Rafeed Rahman

Senior Lecturer
Department of Computer Science and Engineering
BRAC University

Program Coordinator:
(Member)

Md Golam Rabiul Alam, PhD

Professor
Department of Computer Science and Engineering
BRAC University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD

Chairperson
Associate Professor
Department of Computer Science and Engineering
BRAC University

# Abstract

Lung diseases are among the most common medical conditions worldwide, affecting millions of people across all age groups. Thus, early detection of lung diseases is essential for improving a patient's health and outcomes. In recent years, deep learning models have demonstrated remarkable performance in classification of medical images. Despite the development of increasingly robust models with higher classification accuracy, existing hybrid models are not suitable to be deployed in laboratory computers due to the extensive number of parameters they require during the training phase. For example: attention based SWIN and depthwise convolution based MobileNet_V2_S have achieved 66.58% and 86.82% accuracy with 29M and 4.5M parameters respectively. Several hybrid models like LeViT, CVT and MobileViT_S offer 77.03%, 73.63% and 87.09% accuracies with 7.8M, 19M and 5.6M parameters respectively. These state-of-the-art models are not sufficient to work in a resource constrained environment of hospital laboratories. Moreover, these heavy models need large datasets to achieve generalization. However, acquiring such large-scale datasets in the medical domain remains challenging due to various constraints. For bridging these gaps in the SOTA models, in this research we introduce a light-weight deep learning based hybrid model, InvoSparseNet to compete with the best SOTA hybrid models in the tradeoff between accuracy and parameter count. To train our model, we collected a small, primary CXR dataset consisting of normal, pneumonia and abnormal classes. The aim of InvoSparseNet is to assist medical professionals, such as radiologists, in diagnosing lung diseases from X-ray images and it can be adapted in any medical hospital easily, especially in resource constrained environments. To address this objective,we propose a lightweight architecture based on Involution, Sparse Attention, and depth wise convolution for classifying lung diseases. The model is optimized for deployment on PC's as it can be trained on small datasets because it has less parameters, ensuring accessibility and ease of use. To address challenges in collecting and analyzing medical images due to privacy concerns and the risk of information leaks, we have incorporated federated learning. Federated learning allows multiple hospitals or institutions to collaboratively train the model without sharing raw patient data, preserving privacy while maintaining high performance and data security. With just 3.2 million parameters, our model achieves 86% accuracy on the private dataset, setting a new benchmark for lightweight and high-performance solutions.



**Keywords:** Involution; Sparse Attention; Depthwise-Convolution; Federated Learning; Vision Transformer; CNN; Image Processing; Machine Learning; Lungs; Chest X-Ray

# Acknowledgement

First of all, we would want to thank Almighty Allah for keeping us strong enough—both physically and mentally—to finish our thesis titled "Involution Meets Sparse Attention: A Federated Learning Based Frontier in Medical Image Classification" within the deadline. As well as for giving us the information, abilities, and willpower to finish our thesis.

We are thankful for our supervisor, Dr. Amitabha Chakrabarty, Professor of Computer Science and Engineering. His invaluable advice, support, supervision, motivation, and suggestions throughout the thesis-writing process was instrumental. We want to express that his unwavering support allowed us to fully process our work and ideas.

The results of this study was successful not only as a result of individual effort but also as a result of the combined efforts of numerous other people. This research would not have been possible without the overall assistance of our Department of Computer Science. We would want to thank our parents for never giving up on us and continuing to support us while we pursue our jobs since without them, we might not have been able to finish our thesis work.

# Table of Contents

# List of Figures

# List of Tables

# Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

$AI$ Artificial Intelligence

$ANN$ Artificial Neural Networks

$AP$ Average Precision

$AUC$ Area Under Curve

$CLAHE$ Contrast Limited Adaptive Histogram Equalization

$CNN$ Convolutional neural network

$ConvFFN$ Convolutional Feed-Forward Network

$COPD$ Chronic obstructive pulmonary disease

$CPE$ Conditional Positional Encoding

$CVT$ Convolutional Vision Transformer

$CXR$ Chest radiography

$DL$ Deep Learning

$FASA$ Fully Adaptive Self Attention

$FAT$ Fully Adaptive Transformer

$FFNN$ Feed Forward Neural Network

$GAN$ Generative Adversarial Networks

$GELU$ Gaussian Error Linear Unit

$GPU$ Graphics Processing Unit

$GradCAM$ Gradient-Weighted Class Activation Mapping

$IID$ Independent and Identically Distributed

$INN$ Involutional Neural Network

$LGL$ Local-Global-Local

$MHSA$ Multi-Headed Self Attention

$ML$ Machine Learning

$MLP$ Multi layer Perceptron

$NN$ Neural Networks

$PGGAN$ Progressive Generative Adversarial Networks

$ReLU$ Rectified Linear Unit

$ROC$ Receiver-Operating Characteristic Curve

$SOTA$ State of the Art

$ViT$ Vision Transformer

$XAI$ Explainable Artificial Intelligence

$YOLO$ You Only Look Once

# Chapter 1

# Introduction

With the rise of respiratory illnesses threatening millions worldwide, innovative technologies such as Deep Learning are revolutionizing the ways of diagnosing these life-altering diseases. Among the most affected organs is the lung, which has a vital role in facilitating the intake of oxygen and the expulsion of carbon dioxide [3]. However, the respiratory system is vulnerable to pathogens which are responsible for a range of chronic pulmonary diseases, such as pneumonia, lung cancer, severe cough and cold, and other serious illnesses. Although medical experts are capable of diagnosing these conditions with high efficiency from Chest X-rays, the growing number of patients poses challenges in the rapid analysis of chest X-Ray images [21]. Deep learning models have emerged as valuable tools which offer precise detection of disease to assist the radiologists. However, most of the hybrid deep learning models have a complex backbone which require extensive computational time. While large datasets are required to train these heavy models, in real world scenarios, smaller datasets are more available. Furthermore, to ensure the privacy of medical data, decentralized approaches such as Federated Learning have demonstrated significant utility. These models, made to be integrated well into medical workflows, enhance diagnostic efficiency, reliability, and patient outcomes.

## 1.1 Motivation

The respiration process, a fundamental biological mechanism, occurs in the alveoli which are small air sacs within the lungs that facilitate the exchange of oxygen and carbon dioxide [3]. However, this crucial mechanism is often hindered by a range of diseases, including pneumonia, chronic cough and lung cancer. Pneumonia causes inflammation in the alveoli of one or both lungs and is triggered by various pathogens such as bacteria, fungi and viruses. External factors such as age and serious health conditions like diabetes, cardiac disease, and asthma further exacerbate susceptibility to pneumonia [4], [5]. The impact of pneumonia is acute, claiming the lives of approximately 12,000 children annually [7]. The symptoms of pneumonia include fever, chest pain, shortness of breath, nausea, fatigue, and vomiting [1]. Correspondingly, lung cancer presents a severe health burden as abnormal cells proliferate uncontrollably in lung tissues. Symptoms such as persistent cough, chest pain, shortness of breath, and coughing up blood complicate diagnosis and treatment [2]. Early detection of these diseases is crucial for improving survival rates and ensuring a better quality of life. Chest X-rays are a quick and non-invasive diagnos-

tic way for the detection of lung-related diseases. In patients with pneumonia, X-ray images reveal white or hazy areas in one or both lungs, which is an indication of inflammation and fluid accumulation in the alveoli [1]. For lung cancer, the images may show abnormal growths or masses in the lungs [6]. However, the traditional process of diagnosing diseases using X-rays relies heavily on radiologists who analyze images themselves, prepare detailed reports, and relay them to physicians. Due to the high volume of patients and limited time, radiologists often face challenges in delivering timely and accurate diagnosis, potentially leading to misdiagnosis or delays in treatment. To address these challenges, artificial intelligence (AI) has emerged as a transformative tool in the medical sector. AI technologies have proven to harbour remarkable potential in assisting radiologists with medical image analysis and early disease detection. These technologies outperform traditional diagnostic methods in terms of accuracy, efficiency, and speed [8]. Among AI techniques, Machine Learning and Deep Learning are particularly significant.

ML, a subset of AI, enables computers to learn from large datasets and recognize patterns without explicit programming [10], [13]. By drawing inspiration from the mechanism of the human brain to detect patterns in images, sound and text, DL, a more advanced subset of ML, generates meaningful insights and predictions [12], [14]. Deep learning employs artificial neural networks (ANNs) to model complex data relationships, by leveraging multilayered architectures to extract features from images [11]. The rise in the popularity of DL can be attributed to its ability to process vast datasets and advancements in computational power, particularly through the use of GPUs [9], [14]. GPUs are built for parallel processing, making them ideal for the matrix operations required in training CNNs, such as convolutions and backpropagation. This parallelism significantly reduces training times which allows CNNs to handle large datasets and complex architectures efficiently. Neural networks are consisted of multiple neurons that help in classification and other prediction tasks [15]. CNNs, a specialized form of ANNs, leverage GPU acceleration to meet the computational demands of training, enabling them to revolutionize image and video processing by solving complex visual problems [16], [18]. In the medical domain, the advancements achieved by CNNs in analyzing medical images, including X-rays, CT scans, and MRIs have been significant [17]. CNN has the ability to automatically extract features from images and has been proven to be effective in tasks like image classification and object detection [18]. CNNs are widely utilized in analyzing chest X-rays, providing powerful tools for detecting and diagnosing various respiratory diseases such as pneumonia, tuberculosis, lung nodules, and pleural effusion. They outperform traditional image analysis techniques by automatically learning features from raw data, including edges, shapes, and textures, without requiring predefined rules. CNNs achieve this through small filters or kernels that extract local patterns, reduce feature map size, and preserve key information, making the model efficient and robust to input variations [18]. These learned features are then combined to make accurate predictions, such as classifying abnormalities. Depthwise Convolution is a specialized type of convolution used primarily in lightweight neural networks, making it suitable for mobile and embedded applications. It is a variation of the standard convolution operation that reduces the number of parameters and computational complexity, making it more efficient [78]. Despite challenges like dataset bias and interpretability, CNNs have significantly enhanced the speed

and accuracy of chest X-ray analysis, reinforcing their role as indispensable tools in modern radiology.

Along the same line, Involution Neural Network (INN) is an emerging concept in the field of Deep Learning, and has the potential to make significant contributions to the medical domain by providing an efficient and adaptive alternative to the traditional approaches of convolution. INNs employ the Involution operation, a mechanism that dynamically adapts transformations based on the input itself, allowing for localized and context-aware feature extraction [88]. This adaptability can be particularly beneficial in medical imaging tasks where subtle variations in features are crucial for accurate diagnosis.

CNNs and INNs both excel in image processing but are different in the ways they handle feature extraction. CNNs apply fixed filters (kernels) uniformly across the entire image. This allows CNNs to capture general patterns, such as textures or edges, effectively across all regions of the image. However, CNN applies specific kernels for extracting each feature. This approach can lead to higher computational cost as well as memory cost which, as a result, leads to CNNs having a higher number of parameters. INNs, on the other hand, create kernels dynamically to adapt to each region of the image instead of using the same filter everywhere. INN dynamically adapts their kernels based on input, which reduces parameters and computational cost. This increases the efficiency of INNs while also enabling them to capture finer and localized details in tasks like segmentation and high-resolution image processing. To summarise, CNNs perform well in extracting spatial reasoning but INNs show a more prominent performance in this case by adapting to local variations. Both CNN and INN have their own strengths and potentials which can lead to breakthroughs in image processing.

Vision transformers, a recent innovation in Deep Learning, also have demonstrated exceptional performance in image recognition tasks. These transformer-based architectures achieve comparable or superior results to CNNs by processing embedded image patches through a conventional transformer framework [19], [20]. Attention mechanisms are a fundamental component in models like Vision Transformers which allows them to focus on different parts of the input sequence when making predictions. Attention mechanisms such as Self-Attention and Scaled Dot-Product Attention are commonly used, where each token attends to every other token in the sequence. Sparse attention is a modification of the traditional attention mechanism, where the model attends to a subset of tokens, instead of attending to every token [84]. This selective attention reduces the computational cost while making the model more efficient.

Implementations of Deep Learning architectures require storage or large centralized data which is challenging in case of medical images due to privacy concerns. Hence, a system of decentralized learning was proposed in [51] where all the clients train the model locally using their own dataset and only send the weight updates to the server where all the updates from the clients are aggregated. In the medical sector, this decentralized approach has also shown significant impact [68], [69]. Over the recent years, improvements in the field of federated have addressed various challenges like

communication efficiency, privacy, Non-IID data sets etc and these issues continue to be actively researched.

## 1.2 Research Problem

State-of-the-art hybrid architectures use heavy backbone which require many parameters during training because more parameters generally improve accuracy in disease classification. Nevertheless, this makes them difficult to use in hospitals with limited resources, where there may not be enough computing power or infrastructure. These heavy architectures need large datasets for training. However, in the real world scenarios we often deal with small datasets and these smaller datasets bring unique challenges. As a result, many models are not efficiently designed to perform well with limited data. In addition, most of the studies we reviewed did not apply histogram equalization to enhance image contrast, which can significantly influence the model's capacity to achieve high accuracy. Additionally, many researchers have overlooked data security by relying on publicly available datasets. However, in the real world when handling private datasets it is important to ensure data security. For example, patient data is highly confidential, and improper handling can lead to privacy leakages and ethical concerns.

## 1.3 Research Objectives

In this study, we propose an efficient lightweight hybrid architecture of Involution, Depthwise Convolution and Sparse attention mechanism which can work effectively on small datasets for chest X-Ray image classification.

With the increasing number of patients along with rising demand for diagnostic images, radiologists are facing a difficult time to keep up with the sheer volume of chest X-Ray reports. Radiologists are required to skim through each of the reports to analyze subtle abnormalities and categorize them respectively. This increase of workload results in delay in diagnosis as well as human error due to fatigue. The pressure of maintaining accuracy as well as meeting the increasing demands highlight the need for a Deep Learning model that can assist radiologists in diagnosis. Medical data contains personal and confidential information about an individual's health and medical history which is why it is treated as sensitive. Breach of such data could result in potential harm such as identity theft and loss of trust in healthcare providers. Furthermore, medical data includes an individual's name, address and other important information. Ensuring privacy of such information is an ethical responsibility to protect the rights of patients as well as maintaining the integrity of the healthcare system. A Deep Learning model which can ensure the privacy of medical data by utilising a decentralized approach would be an ideal fit for such cases. A decentralized approach would achieve good accuracy with minimum communication rounds and as a result, it can support the processing of large amounts of data in a short period of time as well as assist doctors to make a more confident decision for disease detection from Chest X-Ray images.

Hospital computers are unlikely to GPUs suitable for heavy model training. That

is why, a lightweight model which has lower computational cost along with optimal memory usage would be ideal for hospital computers. For the model to be lightweight, it should have a lower number of parameters. It is often the case that a lower number of parameters results in a lower accuracy rate. But we propose such a model which will have a lower parameter while achieving high accuracy as it is of utmost importance for medical image analysis.

The objective of our study includes:

- Propose a lightweight hybrid model constituted with Deep Learning based architectures which can be deployed in the computers of the laboratory and will work as a helping hand for the radiologists.
- Achieve minimization of the number of parameters without compromising accuracy.
- Preserve privacy of institutions while training the model with the help of Federated Learning.
- Collect a primary dataset to meet the specific requirements of our study.

## 1.4 Research Contributions

In this section we demonstrate the main contributions of our paper.

- We mitigated the problem of training hybrid Deep Learning models with a large number of parameter without trading off the accuracy in our paper.
- New scopes of research on the effects of post-COVID conditions on respiratory systems is facilitated with our primary dataset.
- Preserve privacy of institutions while training the model with the help of Federated Learning.

Figure 1.1 shows the general approach that we are going to follow for implementing our proposed model.

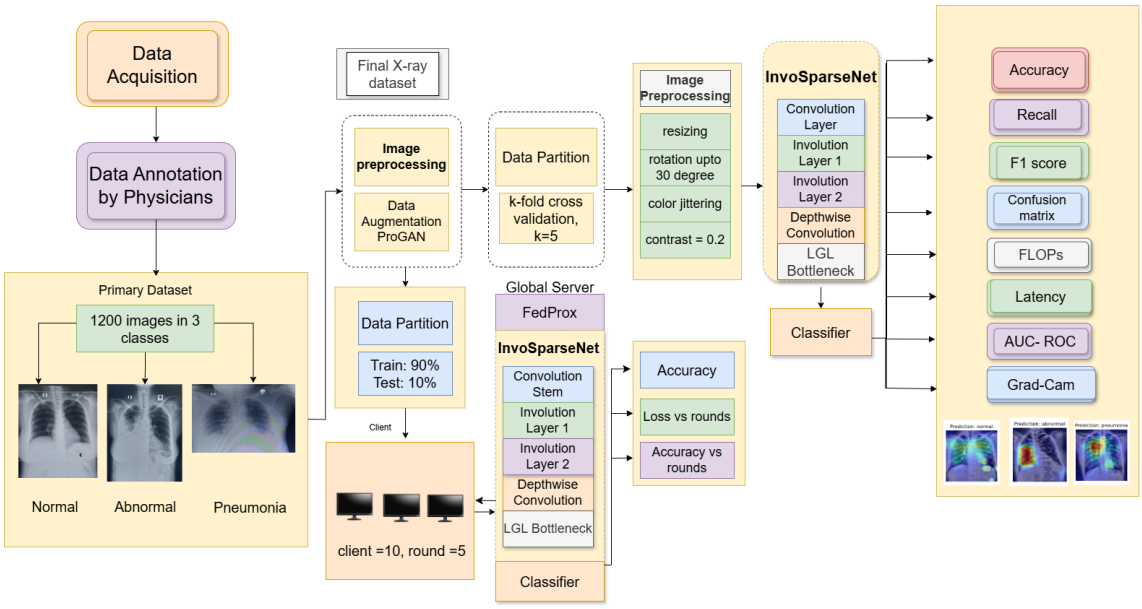


Figure 1.1: General Overview of the Workflow

## 1.5 Thesis Outline

In this section, we will provide the overall outline of the rest of our paper.

- Chapter 2:
  In this chapter, we extensively reviewed the existing work on medical image classification and federated learning. Moreover, we also talked about the research gaps of these existing literatures. Lastly, a thorough comparative analysis among the SOTA models was also provided in this section.

- Chapter 3:
  In this section, we delve into the description of our primary dataset. To begin with, we provided an overall overview of our dataset which included details about its data and classes. Furthermore, the entire methodology of dataset collection was also included. We also talked about different preliminary analyses that were conducted on the raw dataset, for example: class distribution, cropping and adjustments, histogram analysis and other techniques. Lastly, we provided a comprehensive study on the different preprocessing techniques we implemented in our paper, which were: RGB to Grayscale conversion method, CLAHE, Augmentation and Progressive GAN.

- Chapter 4:
  We proposed our model and provided an in-depth description of each fundamental block of our architecture in this section. The illustration of our model was also provided here.

- Chapter 5:
  Here, we describe the overall plan on the implementation of our proposed model. We begin with the entire flow of our work. An algorithm on the centralized implementation and the values of the tuned hyperparameters were

added next. Moreover, we talked about the federated system details. Furthermore, in order to establish the effectiveness of our model, we provided a thorough comparative analysis among most significant SOTA architecture and our model in terms of performance on benchmark dataset, performance on private dataset, experiments conducted with federated environment integration and heatmaps of GradCAM. To conclude this chapter, we discussed details about the ablation analysis that was conducted to propose the most appropriate design of our architecture.

- Chapter 6:
  In the final chapter, we summarize the key findings of our research and discuss our proposed model's contribution in classifying lung diseases accurately in a resource constrained environment. Lastly, we evaluate the limitations of the current approach and discuss the future application and improvement of our proposed architecture.

# Chapter 2

# Literature Review

In this section of the paper, we summarize the research motivation and describe the significant pre-trained and ensemble models for chest disease classification from CXR images.

## 2.1 Deep Learning Models Used in Lung Disease Classification

The study of Krizhevsky et al. [73] presented a novel approach to image classification by utilizing deep Convolutional Neural Networks (CNNs). The study used the large-scale ImageNet dataset, introduced a deep CNN architecture, as well as implemented efficient GPU processing to achieve state-of-the-art results in image classification tasks. The novelties of the study include the utilization of a large dataset, the deep CNN architecture for learning hierarchical features, and the efficiency of the GPU implementation for training large models. Methodologies such as using ReLUs for activation functions and employing dropout regularization were important for model performance and generalization. However, drawbacks include the computational complexity of training deep CNNs and the challenge of interpreting complex neural network decisions. Overall, the paper has shown improvement in image classification task by demonstrating the effectiveness of deep CNNs. Their model achieved top-1 and top-5 error rates of 37.5% and 17.0% on the ImageNet LSVRC-2010 dataset as well as winning the ILSVRC-2012 competition with top-5 test error rate of 15.3%.

In the paper Sanida et al. [22] proposed a lightweight Convolutional Neural Network (CNN) utilizing Chest X-ray (CXR) images that represent seven distinct classes. They had applied a five fold cross validation technique for training and testing the model to optimize robust results. The model had three convolutional blocks with batchNormalisation, ReLU and maxPooling2D for feature extraction, a flattened layer to convert the output of the previous layer. Lastly for classification, three fully connected layers with batchNormalisation, ReLU, dropout layer followed by an output layers were used. The study showed the proposed model outperforms six pretrained models with an accuracy of 98.56%. Since it was a lightweight architecture with parameters 5.35 times lower, it can be easily used in embedded devices or diagnostic tools without any negative effect on its diagnostic efficacy. The model was designed to provide computational efficiency in medical settings where there are

very limited resources. It could be trained in less than sixty minutes to classify the x-ray images.

Hoang et al. [80] examine the impact of placing and removing SE modules on the performance of EfficientNet-B0, the core architecture of the EfficientNets series, in their research study. EfficientNet-B0 is the primary model in the EfficientNet series which integrates several key components for enhancing their performance and minimizing computational overhead. The key components of the model is the mobile inverted bottleneck convolution (MBConv) module which shares ideas from depthwise convolution and inverted residual block. The mobile inverted bottleneck convolution starts with a $1 \times 1$ point-by-point convolution on the input, then adjusts the output channel dimension based on the expansion ratio.Then, performs a depthwise convolution of $k \times k$. Afterwards, it restores the original channel dimension after $1 \times 1$ point-by-point convolution. Finally, the input connection is dropped and skipped. When the SE module is connected, operations are executed after depthwise convolution $k \times k$. This design is the default for EfficientNet-B0 and the EfficientNet series. The baseline EfficientNet-B0 has a Top-1 accuracy rate of 77.2% and a Top-5 accuracy rate of 93.4%, with 5.29 million parameters.

In the paper, Ashraf et al. [76] focuses on designing a CNN architecture which has a lower parameter but can achieve better accuracy like other CNN architecture. Therefore they have proposed a new architecture called Squeezenet which fulfills the criteria. The building block of the Squeezenet architecture is the fire module which comprises two layers such as squeeze layer and expand layer. For faster computation squeeze layer uses $1 \times 1$ filters to decrease the number of input channels. After that it uses expand layer which is a mixture of $1 \times 1$ and $3 \times 3$ filters which expand the feature and extracts useful meaningful features. The number of filters in the squeeze layer is lesser than the total number of filters in the expand layer which makes the module more efficient as it decreases the input sent to $3 \times 3$ filters. This structure helps to minimize computing resources by first lowering and then extending data, while extracting important features using $1 \times 1$ and $3 \times 3$ filters. Squeezenet architecture starts with an $7 \times 7$ conv layer, followed by 8 fire module and end with a $1 \times 1$ conv layer. Max pooling is applied at the end of the network so that it can capture more information in the early layer and reduce it later for efficiency. SqueezeNet can achieve AlexNet-level accuracy on ImageNet with an accuracy of 57.5% on top-1 ImageNet and 80.3% on top-5 ImageNet using 50 times less parameters. For future work the author suggested to work on the design space of CNN architecture like they did while building squeezenet in a systematic way.

In another work, a dense CNN architecture was proposed by the authors of [23] which was used for Covid-19 detection and it achieved better results than other SOTA technologies. Two datasets were used in their work. Dataset1 from Kaggle had 15153 CXR images with 3 classes: Covid, Normal, Pneumonia. Dataset2 from github had 340 CXR images with two Covid and Non-Covid classes. The images were resized to a shape of $224 \times 224$. The architecture consisted of a total of 8 layers where 3 were convolutional and the rest were fully connected.Valid Padding was used with kernel sizes of $3 \times 3$. All 3 of them used ReLu activation to introduce non-linearity. Final output was achieved through a softmax function. There were 10 million trainable parameters and to prevent overfitting, techniques like data

augmentation, L1 and L2 regularizations, batch normalization, early stopping, and dropouts were employed. The most effective ones were dropout and batch normalization.Vanishing and exploding gradient was addressed through weight initialization techniques. Stochastic gradient descent optimizer was used and the accuracy achieved was 98.19% on Dataset1 and 99.8% on Dataset2.

Farhan et al. [24] had designed a new Hybrid Deep Learning architecture (HDLA) based on Machine Learning (ML) and Deep Learning (DL) techniques. They had implied a 2D CNN model utilizing a pretrained RESNet50 model to extract features from chest x-ray. The extracted features were inserted into various Machine Learning classifiers i.e, AdaBoost, Support Vector Machine (SVM), Random Forest (RM), Backpropagation Neural Network (BNN), and Deep Neural Network (DNN). For better performance of the model, each image was first preprocessed using filtering and contrast adjustment methods.They utilized the RESNet model for its fast processing speed and lowest memory needed in comparison to other CNN models. Moreover, it was known to be more powerful for feature extraction compared to other CNN models because the 50 deep layers of the model are trained on ImageNet dataset. In comparison with previous approaches, the proposed model reduced processing speed by 16.91%, increased lung disease classification accuracy by 3.1%, and consumed 15.6% less space.

In [65], the authors suggested a technique that utilised Bayesian Optimizer for maximising the accuracy. The authors achieved an accuracy of 95.79% with their proposed model. They achieved a higher accuracy in Classifying the Dataset into four classes: COVID19, Pneumonia, lung opacity and normal lung. To avoid overfitting issue, augmentation were done on the image. The images were resized to $224 \times 224$ pixels. The model chosen for the base architecture was VGG16. This model was fine-tuned for enhancing its performance. A sequential model was intimated by adding VGG16 layers. Dropout layer of 0.1 to 0.5 rate and dense layer with range of 32 to 256 were used. For the end layer of the tuning phase, soft max function was employed. Images from the training set were fed into the fine tuned VGG 16. The parameters are optimized by a Bayesian optimizer. The proposed model acquired better performance than models developed in previous studies. Future scope involves aggregating multiple architecture and optimising the hyperparameters through optimal selection of iterations and epoch count.

The study of Subashini et al. [25] was conducted on the classification of COVID-19 using Deep Learning models using images of chest X-ray. The focus of this study was the utilization of pretrained CNN architectures such as VG16, Inception Resnet and a custom CNN to differentiate between Covid pneumonia, viral pneumonia and normal chest X-ray images. Transfer learning was implemented on the pretrained models to leverage features learned from large datasets. Data preprocessing involves resizing images, normalization, and augmentation techniques to enhance model performance. This study highlights the importance of transfer learning as well as custom CNNs in the task of medical image classifications. Although Subashini et al. achieved promising results, with high accuracy rates of 96% for VGG16, 97% for Inception Resnet and 97% for the custom CNN, there remains some limitations such as the inability to determine the stage of Covid-19 pneumonia as well as potential overfitting.

To avoid overfitting, Abdullah et al. [26] used a dropout layer in the head model at a rate of 0.5 in their proposed hybrid Deep Learning CNN architecture. The hybrid Deep Learning CNN architecture was designed by a head model and a base model. Different activation functions and a dense layer of size three are added in the heading model. The base model was built using two pre-trained Deep Learning structures i.e, VGG16 and VGG19. In addition, different pooling layers were added to the base layer to minimize the feature dimensions obtained from the pretrained model. They assigned the head model at top of the base layer. They collected their dataset from COVID-19 Radiography database and their proposed model gained an accuracy of 92%.

Many researchers have used DL models for pulmonary disease detection. For instance, Sharma et al. [27] presented a Neural Network based VGG16 for detecting the symptom of Pneumonia with a benchmark accuracy of 95.4%. In their work, they had used two public datasets retrieved from Kaggle. The data were pre-processed through normalization and balancing. VGG-16 was capable of working with small filters instead of using a large number of hyperparameters. For this reason, VGG-16 was used for extracting features from the CXR images. This model contains a $3 \times 3$ filter and $2 \times 2$ filter for padding and max pooling. The pre-trained model utilized Adam optimizer with a learning rate of 0.0001 and as activation function, ReLU was chosen. In the output layer, two fully connected softmax layers were used. In order to evaluate the performance of the proposed model with state-of-the-art architectures such as: NB with VGG16, KNN with VGG16 model, SVM with VGG16 model and Random Forest with VGG16 model, comparison was done in terms of performance accuracy. Lastly, from the results it was evident that this model exhibited an accuracy of 92.15% and 95.4% for the first and second Datasets respectively which was higher than existing models.

In addition, Hussain et al. [28] employed collaborative ensemble architecture. Initially, the pixels of the image dataset were normalized to a range of [-1,1]. Moreover, data augmentation was used to avoid overfitting and for utilizing the small dataset. Multiple Deep Learning models were trained on the dataset. Later, top three models were chosen to formulate the ensemble model. EfficientNet-B0, VGG-16 and DenseNet201 were the selected models. The following equation 2.1 was used for deep ensemble learning:

$$f(Y) = \sum_{i=1}^{n} W_k f_i(Y) \tag{2.1}$$

In order to improve performance of the DL models, Transfer learning was used and certain layers were frozen during the fine-tuning process. Before feeding the images into models, they were resized into $224 \times 224 \times 3$ dimensions. For each DL model, three flattening layers are used. The ReLu activation function was sandwiched between those layers. 30% of neurons were dropped in the dropout layer of each model. Finally, the input was classified into normal, Pneumonia, and COVID-19 with the help of SoftMax Classifier.

Another prominent model which was constituted with the power of convolution is MobileNet. Howard et al. proposed this architecture for the first time in [92]. MobileNet demonstrated a remarkable performance in Imagenet classification compared

to other SOTA models. The core layers of the model are built upon the operation of depthwise separable convolution. They splitted the convolution into two layers: 1) filtering and 2) combining. This division has increased the efficiency significantly in comparison with convolution. The filtering is done with a depthwise convolution layer and combining is done with a pointwise convolution.

A Standard Convolutional has a cost of:

$$DK \times DK \times M \times N \times DF \times DF \tag{2.2}$$

where, DK is the spatial dimension of the kernel, DF is the spatial dimension of the feature map, M is the number of input channels and N is the number of output channels as shown in equation 2.2.

On the other hand, in equation 2.3,Depthwise convolution has a computational cost of:

$$DK \times DK \times M \times DF \times DF \tag{2.3}$$

In equation 4.3, the formulation of depthwise convolution is presented. The simple yet effective architecture in [92] was built on depth wise separable convolutions for all the layers. Exception lies in the first layer which utilizes the convolution. Moreover, ReLu and batch normalisation was applied in all the layers except for the final MLP layer layer. The later parts include an average pooling layer, a MLP layer and softmax function for classification. In terms of performance, MobiletNet achieved an accuracy of 70.6% with only 4.2 million parameters.

Introduced by Sandler et al. [78], MobileNetV2 is a neural network architecture tailored for mobile and resource-constrained environments. It builds upon the efficiency of its predecessor, MobileNetV1, by introducing a few architectural changes. The core components of MobileNetV2 include inverted residual blocks, which connect narrow bottleneck layers with fewer channels to improve gradient flow and reduce memory usage. In these bottleneck layers, linear bottlenecks avoid non-linear activations like ReLU to prevent information loss, preserving feature representations in compressed spaces. Additionally, depthwise separable convolutions replace standard convolutions by separating filtering and feature combination, significantly reducing parameters and computational cost. MobileNetV2 uses the ImageNet dataset, achieves 72.0% top-1 accuracy using 3.4M parameters and 300M Multiply-Adds (MAdds), outperforming MobileNetV1 and ShuffleNet in computational efficiency while maintaining competitive accuracy. Despite its efficiency, MobileNetV2 sacrifices some accuracy compared to heavier architectures like ResNet or NASNet. Achieving high performance for specific use cases may require fine-tuning hyperparameters, such as expansion ratios and input resolutions. The main innovation in MobileNetV2 is separating how much information the model can hold through bottleneck layers from how flexible it is in learning patterns through expanded layers. This design is ideal for mobile devices as it greatly lowers memory usage as well as creates a flexible structure that can handle different tasks without needing specialized hardware. MobileNetV2 excels in balancing accuracy, computational cost, and memory efficiency. However, its limitations highlight the ongoing trade-offs in designing lightweight architectures.

Similarly, another model based on Neural Network was proposed by Alshmrani et al. [29] recently. VGG19 and CNN techniques were implemented in their paper for a multiclass detection of Lung Opacity, Pneumonia, Lung Cancer, COVID-19 and TB. Various techniques such as resizing, normalization and converting the images into an array was applied for feeding the images into the model. Images of $224 \times 224 \times 3$ dimensions were used and the datasets were randomly splitted into 80% - 20% for training and validation sets respectively. Moreover, mechanisms of VGG 19 with three CNN blocks were employed for effective feature extraction. ReLu was used as an activation function for Convolution in each CNN block. Furthermore, batch normalization and max pooling layers were added before a dropout layer. In the output step, the final class was calculated by SoftMax. The proposed model implemented 0.000009 learning rate in the Adam optimizer function. As a result, 96.48% accuracy in performance was achieved.

Ibrahim et al. [30] observed performance of four proposed Deep Learning architectures such as VGG19-CNN, ResNet152V2, ResNet152V2 + Gated Recurrent Unit (GRU), and ResNet152V2 + Bidirectional GRU (Bi-GRU) for lung diseases detection and diagnosis and among which VGG19+CNN showed better performance and accuracy of 98.05% followed by ResNet152V2 + GRU with an accuracy of 96.09%. On the contrary, as compared to the other architectures, ResNet152V2 and ResNet152V2 + Bi - GRU had the lowest accuracy rates, with 95.31% and 93.36%, respectively.

The proposed model of Liu et al. [31] was a novel Deep Learning based method to predict various kinds of pneumonia types on chest X-ray images. In this study, along with CNN, a parallel pyramid MLP-mixer module which was self designed was utilized for the comprehensive feature analysis of multi-channel input image matrices, leading to the accuracy of the classification of pneumonia diseases. The study brings innovation by combining CNN features with the parallel pyramid MLP-Mixer module to enhance feature extraction and classification accuracy in diagnosing pneumonia from CXR images. The methodology involves using a Deep Learning framework that combines the YOLO-v4 network for thoracic region segmentation, a CNN for feature extraction, and parallel pyramid MLP-Mixer modules for comprehensive feature analysis. This approach aims to accurately classify the pneumonia cases of COVID-19 from chest X-ray images by extracting as well as fusing image features at different scales and channels. The localization step enhances the model's ability to extract meaningful features from the targeted thoracic region, contributing to the overall effectiveness of the diagnostic approach. Even with classification as well as localisation, the model that had been proposed had a high accuracy of 98.25%.

Explainable Artificial Intelligence, shortly known as XAI,is used to make Machine Learning models more transparent and understandable to humans. This was an important area of research as most of the Machine Learning model behaves like a "black box" where we receive an output without having a clear idea of how the model takes the decision. XAI provide an explanation based on the decision making process of ML models that helps AI system to gain more trust and acceptance by humans. Islam et al. [32] proposed a novel architecture DCNN-GRU which involved XAI and was implemented on lung cancer and Covid-19 datasets where it gained an accuracy of 98.97% and 99.30%. The new architecture applied Convolutional Neural Network

(DCNN) with a Gated Recurrent Unit (GRU) and Explainable AI strategies (XAI). The Deep Convolutional Neural Network part of the model extracted meaningful features from the lung images and passed to the Gated Recurrent Unit for classifying the images. The study used various XAI methods i.e, LIME, SHAP and Grad-CAM for recognizing important features related with the classification such as ground-glass opacities, consolidation patterns, nodules, masses, and tumor growth pattern that helped the model to make the decision and provide an effective communication between medical experts and patients. Furthermore, in comparison to previously used models, it claimed that the model required a shorter amount of training time.

Ravi et al. [33] proposed a model with multiple CNN architectures and a stacked ensemble classifier. Their model performed well and achieved an average accuracy of 98%. At first, normalization was done on the input image dataset to convert the pixel values between 0 and 1. Pretrained models were employed with the aim to increase the performance. For this reason, EfficientNetBo, EfficientNetB1 and EfficientNetB2 were trained on the Imagenet dataset. In the last layer of these models, in order to avoid overfitting, the Global Average Pooling(GAP) layer was used and use of fully connected layers was avoided.These models were used for feature extraction. The extracted features from all three models are fused together and fed into multiple fully connected layers. Among the multiple FC layers, dropout layer and batch normalization layers were also included. The final output was generated using stacking ensemble classifiers.The stacked ensemble learning classifier contained two stages which are random forest and SVM, and logistic regression respectively for lung disease detection. In comparison to previously studied models, the proposed multi-channel EfficientNet Deep Learning-based stacking approach achieved a more accurate performance for classification among different lung diseases. However, the authors did conduct study regarding the probable information loss that can happen during resizing of CXR image. Moreover, using a real-time dataset, thorough investigation and analysis of the proposed model can be done in the future. The robustness of the proposed method can be increased by fusing other features apart from using simple concatenation as the only fusion feature.

Hroub et al. [34] developed an advanced, reasonably priced explainable Deep Learning diagnostic tool to identify lung illness from medical imaging data. For developing this tool, firstly they had trained 10 conventional DL methods and 6 ViTs methods using both original and augmented data. This study suggested that data augmentation had shown higher accuracies performing this Deep Learning model. After examinations of these techniques, the study showed that some conventional Deep Learning models(Inception-V3, MobileNetV3, DenseNet, EfficientNet) performed better compared to ViTs (CCT, CvT, MobileViT, NesT, SepViT, and SimpleViT). Moreover, inceptionv3 outperforms all Deep Learning techniques, including ViTs, whether employing random or pretrained weights. The CAM algorithms, including GradCAM, GradCAM++, EigenGradCAM, AblationCAM, and RandomCAM, showed efficacy in recognizing the attention zone that comprises the presence of pneumonia and COVID-19.Their method gained accuracy of 94.55%. Lastly, they had created a unique online platform that employs a Gradio architecture and seven Deep Learning models together with explainable algorithms to categorize lung conditions into three phases (pneumonia, COVID-19, and normal).

Medical datasets of chest diseases are complex and large in number. Different Deep Learning models such as CNN and RNN do not show promising performance in analyzing this kind of complex data. Transformers solve this issue by utilizing the self-attention mechanism.In the paper [35], the authors replaced the recurrent layers which are common in encoder-decoder based models with self-attention layers. The encoder part of the transformer consists of multi headed self mechanism and feed forward layers. The main part of the model was the self attention layer. In this layer, the inputs are converted into a vector and represented as a sequence. Spatial relations of these sequences are computed in this layer. For this reason, transformers are capable of analyzing the local dependencies. Moreover, positional encoding was also included in this layer which enables the model to determine the position of each element in the input sequence. From the input vectors, (queries, keys and values) are generated. The dot product of queries and keys are computed and then it was divided by $\sqrt{d_k}$.Here, $Q$, $K$, $V$ mean the matrices of Queries, Keys and Values. Moreover, $d_k$ means the dimension of Queries. Finally, the softmax function was applied to obtain the entire weight on the values.
The decoder layer also consists of stacked layers like the encoder layer to generate the output. The encoder layer takes the input all at once. Whereas, the decoder layer generates output one by one relating each output with the previous one. This sequence continues until the entire output was generated.The authors have applied dot product attention which was computationally efficient because it can be implemented with optimized matrix multiplication code. The self attention mechanism allows to capture long range dependencies among the input elements. For this reason, attention based models are proven to showcase robust performance in data processing.

Transformers were initially used in NLP oriented tasks. In [95], Dosovitskiy et al. applied the operation of transformers directly on images. Firstly, an image is fed into a transformer encoder as patches with their linear embedding. These patches are treated the same way as words in NLP. In the encoder, the input patches first pass through a normalisation layer. Then they are passed through Multi-headed self attention layers; the mathematical formulation of self attention operation is detailed in equation 4.5 of our paper. The final two layers are normalisation and MLP layers. The class token from the encoder is finally passed to the classifier. The authors visioned to apply a more scalable ViT in other computer vision related tasks.

Using the similar motivation, Singh et al. [36] implemented the self attention mechanism on the CXR dataset to detect pneumonia accurately with 97.61% accuracy. The authors employed patch embedding, positional encoding, multiple transformer layers, self attention mechanism, feed forward neural networks and classification head in their proposed architecture. The input images were reshaped into patches and embeddings were obtained from linear transformation. After that, positional embedding was integrated with the input images in order to retrieve the relative and absolute position of the patches. These patches were fed into multiple encoder layers. Finally, a MLP head classified the input data into Pneumonia and normal classes. The authors stated that ViT facilitates the detection of pneumonia by extracting local and global features of an image. Usually large and complex datasets are available in the medical sector. The proposed architecture was able to handle

such a dataset. However, the reasoning behind the decision making process of ViT was yet unknown. Further research can be conducted to increase the interpretability of the ViT architecture. The efficiency of the model can be increased by reducing its time complexity.

In their work, Srinivas et al. [85] present BoTNet, which combines Multi-Head Self-Attention (MHSA) with the ResNet architecture to improve computer vision tasks. While ResNet is efficient for local feature extraction, it struggles with long-range dependencies like most CNNs do. This is addressed by BoTNet by replacing the final three $3 \times 3$ convolutions in ResNet's bottleneck blocks with MHSA, which enables global context capture. BoTNet uses relative position encodings for better spatial awareness as well. BoTNet performs well on ImageNet benchmarks by reaching 84.7% top-1 accuracy on ImageNet, matching EfficientNet-B7 while being 1.64× faster on TPU hardware. The model is shown to be efficient, with fewer parameters than ResNet. However, BoTNet has limitations: the self-attention mechanism adds computational overhead, especially for large images. It also requires longer training or data augmentation for significant improvements. The paper mainly focuses on segmentation, detection, and classification, leaving other areas unexplored. BoTNet effectively combines the strengths of CNNs and self-attention, offering strong performance across vision tasks with an efficient design, setting a solid foundation for future research.

The Vision Transformer architecture went through various structural improvements and one of the major architecture was discussed In the paper [37], the authors proposed the Less Attention Vision Transformer (LiT) which showed improvements by usage of an MLP in the early stages and the use of a novel deformable token merging module. The entire architecture was divided into 4 stages where the first two stages were composed of MLP blocks and the others were standard Transformer blocks.The MLP block was made up of two FC layers with a non-linear activation function (GELU) and also layer normalization was applied. This initial use of MLP block helped to capture the local dependencies and also avoided huge computational costs because the use of MSA in initial layers could be expensive due to long token sequences.The last two blocks were standard transformers where each block had two MSA layers followed by an MLP layer with layer normalization applied in both cases. MSA was applied with sufficient head and it captured the long-range dependencies.Furthermore, in the 2nd, 3rd, and 4th stages a DTM (Deformable Token Merging) was added before. To merge patches in an adaptive manner and to learn a grid of offsets to adaptively sample more informative patches, a Densely Connected layer with Batch normalization and GELU was used. This implementation introduced negligible FLOPs and parameters. Compared with some other SOTA architectures like ResNet-18, DeiT, PVT and Swin proved that LiT produces best results (Reached Top-1% accuracy of 83.4% in the ImageNet dataset).

Concepts of LIT architecture was continued and the authors of the paper [38], proposed the LITv2 architecture which was based on the HiLo (High Low) attention mechanism with two segments, where one segment focused on High frequency and the other focused on low frequencies to achieve more efficiency. Furthermore, they used throughput to evaluate a ViT instead of FLOPs because throughput was a more direct metric for evaluation. At first, the images were passed to a $3 \times 3$ depth-

wise conv layer with zero-padding in each FFN. This approach helped to enlarge the receptive field in early stages and also served as a better substitute for the relative positional encoding (RPE) used in LITv1. Then feature maps were fed to two parallel segments. One segment for High Frequencies and the other one for Low Frequencies. However, in this approach, doubling the number of attention heads would increase the computation cost. As a result, a hyperparameter, $\alpha$, was used as a splitting ratio to split the attention heads between the low and high segment instead of naively going for a 50/50 splitting. The high-frequency segment helped to capture the high frequencies in an image, which were the rapid changes in pixel values. On the other hand, the feature maps were down-sampled with average pooling before being fed to the Low-frequency segment where the low frequencies in an image were learned, which were the smoother and more gradual changes in pixel values. Finally, they were concatenated. The LITv2 outperformed all SOTA architectures in terms of throughput, required less FLOPs and attained 84.7% (Top-1% accuracy).

Slika et al. [39] proposed a image augmentation scheme and neural network based model, Vision Transformer Regressor Infection Prediction (ViTReg-IP) to predict the severity of COVID-19 condition. At first, the image pixels were normalized and reformatted into $224 \times 224 \times 3$ dimensions. In addition, data augmentation methods like offline combined lung and score replacement were applied to increase the size of the dataset. These methods were also responsible for enhancing the performance of the model.The proposed model consists of ViT and a regression head. Deep neural networks were used and they were pretrained on the ImageNet dataset. At first, CXR images were reshaped by ViT and converted into flatten 2D patches. In order to obtain information related to positional dependence and spatial patch, position embedding was included with patch embedding. The Encoder layer of ViT included Multi Headed self attention layer and MLP blocks.The last part of the architecture contains a regression head. There are two fully connected layers and it generates the severity of diseases using CLS token. The model's output was the score which indicates severity of the disease. Absolute error loss and standard Gradient descend (SGD) were chosen as loss function and optimizer for the model respectively. The authors proposed a novel image augmentation scheme through which segmentation could be avoided in the preprocessing part. As a result, the model became computationally efficient. The proposed architecture was a neural network based model which includes the principles of Vision Transformers (ViT) with a small number of trainable parameters for classifying various pulmonary diseases including COVID-19. Moreover, severity of the disease was also quantified. This model can be used as a multi-task model since it can predict multiple scores concurrently. CT scans can also be used as input for the model in future. However, there was an insufficiency of annotated data for left and right lungs with individual score. This issue only affects the augmentation phase, and using an infection mask can aid in calculating the individual scores.

For application in resource constrained devices or in mobile devices, lightweight Deep Learning architecture was necessary. In the paper [40], the authors proposed a family of LightWeight Vision Transformers known as Fully Adaptive Transformers which used the novel Fully Adaptive Self attention mechanism that handled local and global dependencies along with bidirectional interaction between local and

global contexts. The FAT Architecture was made up of 3 modules: Convolutional Feed-Forward Network, FASA and Conditional Positional Encoding. CPE applied convolutional positional embeddings to the patches which were then fed to the FASA module. FASA had two parallel segments for Global and Local information capture with MHSA and Convolution, respectively. Then they were combined and sigmoid applied long bi-directional combinations with point-wise convolution to produce the output from FASA. ConvFFN was introduced to focus further onto local features. There were two types of parallel convolutions, one with stride of 1 and the other with stride of 2 to apply down-sampling which reduced dimensionality and also parameter count. The style in which features were aggregated there was known as Context Aware Feature Aggregation, was made up of Modulation and Aggregation. In the FASA method, the aggregation was improved compared to traditional Self-Attention as down-sampling was applied. Also, there were two feature aggregations as it was bidirectional and they were modulated with each other. Global Adaptive Aggregation was a fine-grained down-sampling strategy which minimized the loss of global information. It used a convolution of stride 2 to down-sample Key and Value vectors. Unlike MHSA, the last linear layer was omitted. Local Adaptive Aggregation was used to retain the local features where convolution was used with sigmoid activation. A cross-modulation technique was used to apply the bidirectional adaptive interaction. The proposed architecture achieved 77.6% accuracy on ImageNet which was better than many lightweight models with 4.5M parameters and 0.7G FLOPs. However, as it was computational resource constrained so it cannot be applied on heavy tasks like unsupervised learning, video processing, or visual language tasks due to its lightweight vision backbone. Also, it was not applied on Large Datasets like ImageNet-21k.

With some significant structural changes and use of a more direct metric to evaluate ViT architectures, the authors of the paper [41], proposed the Fast ViT architecture. They used a novel token mixer known as RepMixer to remove skip connections, applied linear train time overparameterization to improve accuracy and large convolutions in early stages to avoid self attention. The architecture was designed as such: it contained a stem at first in which the input image was fed. Here, large convolutions were applied to improve the receptive field. Then came 4 stages of the architecture which operated at different scales. After each stage, a patch embedding and stride of 2 was applied except in stage 4. This patch embedding was achieved through a series of convolutions, batch normalizations, activation functions, and concatenation. It improved the model's performance and robustness. In each stage, there was a RepMixer (token mixer component) followed by a ConvFFN block. RepMixer could be re-parameterized as a 3x3 conv during inference time to avoid the skip connections. However, in stage 4, conditional positional encoding was applied and then an Attention mechanism to retain global dependencies. Finally, the output from stage 4 was fed into a 3x3 depthwise convolution, average pooling and then a fully connected layer to receive outputs. Fast ViT had an 83.9% (Top-1% accuracy) on the ImageNet-1k which was better than other SOTA architectures with much lower latency. Furthermore, the lightweight version ( FastViT-T8 ) had 3.6M parameters and achieved 75.6 top 1% accuracy which was better compared to MobileOne (71.4%) with the same latency.

Touvron et al. [89] proposed the DeiT (Data-Efficient Image Transformer) model,

a Vision Transformer that can be trained effectively on a limited amount of data, addressing the data inefficiency challenges typically faced by standard Vision Transformers (ViTs). A key contribution of their work is the introduction of a distillation procedure, which adopts a teacher-student strategy. In this framework, a student model learns not only from labeled data but also from a teacher model via an attention mechanism, enhancing its performance and generalization.For the teacher model, they utilized ConvNets or and student models are image transformers, which were trained on large datasets. The innovation lies in the integration of distillation tokens, which are added alongside the traditional class tokens in the embedding layers. These distillation tokens interact with the input embeddings using self-attention mechanisms and facilitate the transfer of knowledge from the teacher to the student. This approach ensures that the student model benefits from both the dataset and the learned representations of the teacher model. The accuracy of the DeiT on ImageNet with and without external data are 85.2% and 83.1%.

Normal Transformers lacks the inductive bias. Furthermore, traditional CNN models fail to utilise the features properly for having a small receptive field and inability to capture long dependencies in an image. To solve these issues, Hao et al. [62] proposed DBM-ViT model to detect lung diseases. The authors worked with both CXR image and CT scan datasets to classify Covid-19, pneumonia and normal lung condition. At first, they used augmentation on image of $224 \times 224$ dimension to avoid overfitting and image pixels were normalized in range of 0 to 1. Transfer Learning was implemented to add robustness to the proposed model. VGG-19, DenseNet201 and Inception V3 models were selected to integrate the transfer learning network. The proposed DBM-ViT network had four major parts: DB-Conv module, DM-Conv module, ViT module, and MLP module.DB conv module was used to learn global features by expanding the receptive field. Moreover, pointwise Convolution in this module helped to project features into a new space and BN normalisation layer was added. After that, DM-Conv module extracted Pneumonia features again and downsampled the data. ViT module takes feature map along with position and classification vector as input. By stacking multiple transformer layers, local features were learnt. Lastly, classification was done with the help of MLP layer.The employed DBM-ViT model could efficiently detect lung diseases with an accuarcy of 97.8% along with capturing local and global dependencies and solving the problem of inductive bias.

Yang et al. [71] proposed a model name CovidViT that was typically based on a transformer block with the self attention mechanism to analyse chest x-ray images of covid-19, pneumonia, normal. It was a ViT based model consists of three crucial parts, i.e., image embedding, transformers encoder and full connection classifier. Initially the image embedding transforms images to vector sequence. Following this, the transformer encoder extracts features from input vector sequences. Finally, according to the extracted feature, full connection layers predict decisions. To reach better accuracy the used all the outputs of transformers whereas ViT only uses the first output of transformers encoder. They have also designed many CNNs to compare their performance with the proposed model. However, the study showed that Vision Transformer based CovidViT had outperformed different category of CNNs with an accuracy of 98.0%.

Ghali et al. [42] proposed five models such as ARSeg, TransM, Medical Transformer (MedT), TransUNet, and UNeXt to ensure the accurate lung detection and segmentation in healthy and unhealthy CXR images. Furthermore, three publicly available datasets and two loss functions: combo-loss and dice-loss were used to evaluate the models' performance. The experiments' results demonstrated that the suggested models had effectively identified important local and global characteristics in the input CXR images. Among the models, TransM acquired the highest F1 score, 97.47%. The f1 scores of MedT, TransUNet, UNeXt, and ARSeg were 97.40%, 96.80%, 96.26%, and 95.36%, respectively. They showed enormous promise in distinguishing lung shapes according to patient age, gender, and lung diseases such as TB and nodules, as well as in separating lung zones from the rib cage and clavicles.

There are other architectures which have just as much, if not more, promising results as pure CNN architectures. In the paper [43], the authors presented a comprehensive study of using DL models for detection of Covid-19 from CXR images. This study utilized CNN as well as ViT architectures and raised a comparison between the two. The CNN based models chosen for the classification task are ResNet50, MobileNet and EfficientNetB7. They used a mini-batch size of 64 images, dropout rate of 0.3, Adam optimizer with a learning rate of $10^{-3}$ and the models were trained for 300 epochs. Models were pre-trained on ImageNet dataset for 30 epochs. On the other hand, the ViT models used in the experiments included Twins, Swin, and Segformer with varying specifications. The study incorporated segmentation techniques, such as the use of a UNet model, and augmentation methods, including rotation at distinct angles, to preprocess CXR images. Due to the enhancement of quality and diversity of training data, the performance of the model improved. Among CNN based models, the highest accuracy rate was of EfficientNetB7, which was 99.82% and among the ViT based models it was SegFormer with the accuracy rate of 99.81%. It can be observed from the accuracy rates, both CNN and ViT based models accurately classified CXR images into classes including COVID-19, viral pneumonia and normal cases. The results suggested that both CNN and ViT models can be effective in aiding medical professionals in diagnosing respiratory system disorders, particularly during the COVID-19 pandemic.

Selvano et al. [44] measured the performance of CNN models that have already been trained (ResNet50), self-supervised learning (SSL) approaches, and ViT in the classification of four lung conditions using a publically accessible dataset with over 20,000 CXR images. Among all their proposed models DINO ViT-S16 showed the best scores. Three imbalanced, augmented, and undersampled datasets were generated in order to fully examine the differences between SSL models and supervised learning models and their performances on the same dataset with changed interclass data distributions. The robustness of SSL models in situations where there are significant variations in the quantity of data per class was assessed using the unbalanced dataset. The efficacy of SSL models to increase accuracy by appending altered images through data augmentation was evaluated using the augmented dataset. The potential of SSL models to minimise bias and increase accuracy when the number of images per class was equal was examined using the undersampled dataset. Lime was used for the visualisations of each class ( Lung Opacity, COVID-19,Viral Pneumonia, Normal).

Higher accuracy in performance of the models was obtained by combining the self attention mechanism of ViT and CNN. Fazal et al. [45] employed a Deep Learning based COViT model for Covid19 and Viral Pneumonia detection with higher accuracy. The proposed model utilized $3 \times 3$ filter for convolution step which was followed by alternate self-attention and MLP with hardswish function. Furthermore, an average pooling layer was employed in this model through a MLP head. For detecting the classes distinctively, fully connected layers with ReLu activation function and kernel L2 were used. This proposed model achieved a higher accuracy of 99.50% in performance than the existing CNN models.

Multiple label classification of images was more challenging then single label classification. In this paper, Wu et al. [46] introduces a novel parallel hybrid Deep Learning model called CTransCNN that comprises both CNN and transformer. The CNN and transformer branches consist of three main components such a multilabel multihead attention enhanced feature module (MMAEF), a multibranch residual module (MBR), and an information interaction module (IIM). The MMAEF component allows exploration of implicit correlation between labels whereas MBR provides model optimization. For feature exchange between two branches, the information interaction modules (IIM), specifically the C2T module and T2C module have been used. The study focuses on effectively utilising these components to discover the implicit correlation among labels and improve multi-label medical image classification.CTransCNN achieves the highest scores in both AUC and mAP metrics, with values of 84.56% and 67.67%, respectively.

Touvron et al. [72] proposed a method combining the strengths of Vision Transformers and CNN by integrating soft convolutional inductive biases into Vision Transformers. This study introduced Gated Positional Self-Attention (GPSA) layers, which enabled each attention head in adjusting a gating parameter hence balancing attention between position as well as information regarding content. The role of locality in learning by examining how it was naturally encouraged in standard self-attention layers and managed in GPSA layers can be observed in this study. The impact of locality on model performance and efficiency had been shown in this study. This architecture outperformed existing models such as DeiT on ImageNet with improved sample efficiency, representing a significant advancement in combining the strengths of architectures such as Vision Transformers and Convolutional Neural Networks. The model achieved a top-1 accuracy of 56.4%, as well as showing significant improvement in both top-1 and top-5 accuracy across different training dataset sizes when compared with the DeiT-S model on ImageNet-1k.

The paper by Wu et al. [74] introduced Convolutional vision Transformer (CvT), a novel architecture that combines Vision Transformers and Convolutional Neural Networks in the enhancement of the performance and efficiency in computer vision tasks. By incorporating convolutions into the Transformer architecture, CvT enabled efficient capture of local information while maintaining global context and dynamic attention mechanisms. The hierarchical multi-stage structure with Convolutional Token Embedding allowed for spatial downsampling and increased token feature dimensions, resembling CNN operations. Additionally, the Convolutional Transformer Block in CvT utilized a convolutional projection to improve local spatial context and reduce semantic ambiguity in the attention mechanism. These

innovations result in state-of-the-art performance on ImageNet-1k with fewer parameters and lower FLOPs in comparison to traditional ViTs and CNNs. CvT-W24, when pre-trained on ImageNet-22k, achieved a top-1 accuracy of 87.7% on the ImageNet-1k validation set.

The work of Pan et al. [84] introduces an innovative Local-Global-Local (LGL) bottleneck through their hybrid model EdgeViT to achieve efficient image recognition on mobile and edge devices. The overall algorithm of EdgeViT works in three stages: Local Aggregation for local context extraction, Global Sparse Attention to gain global information with reduced computation cost and Local Propagation to diffuse back the global information. EdgeViT employs a hierarchical architecture with progressive downsampling across four stages. Its Local-Global-Local bottleneck combines the mechanism of local aggression, global Sparse Attention, and transposed convolution. The model achieves superior performance on ImageNet-1k, surpassing models like MobileNet and PVT in classification as well as detection. EdgeViT is small enough to fit in mobile devices and is easy to deploy. The usual ViT models have a quadratic complexity, which makes them unsuitable for mobile devices, but the complexity of EdgeViT is significantly lower due to the implementation of the delegate token-based attention in the mechanism of the global Sparse Attention. EdgeViT can also work with limited resources, which is a limitation of usual ViT architectures. It emphasizes real-world efficiency with direct on-device measurements for latency and energy. However, its dependence on hardware-specific optimizations and the complexity of the LGL bottleneck are potential limitations. Notable features include the hybrid CNN-ViT design, delegate token-based attention, and multiple model variants (XXS, XS, S) to cater to diverse hardware constraints and computational budgets.

Guo et al. [67] addressed issues of transformers and CNN regarding performance and computational cost. To mitigate these issues they have proposed a new transformer based hybrid network CMT architecture for visual recognition which utilises transformers to pursue long-range dependencies and CNN to extract local information respectively. The architecture first sent images as input to the convolution stem (initial set of convolutional layers) for feature extraction and then fed to a stack of CMT blocks for representation learning. To extract multi-scale features, they applied four convolutional layers with stride 2, to decrease the resolution and increase the dimension flexibly in a stage-wise architecture manner which eventually helps to reduce computation burden generated by high resolution. The local perception unit (LPU) and inverted residual feed-forward network (IRFFN) in CMT was used to extract both global and local intermediate features. Lastly, the pooling layer was used for better classification results. Their experiments showed that even though being 14× and 2× less than the best current DeiT and EfficientNet, our CMT-S achieves 83.5% ImageNet top-1 with only 4.0B FLOPs. The model performed well in downstream vision tasks as well.

Ukwuoma et al. [70] proposed a novel hybrid explainable AI framework using Deep Learning both ensemble convolutional networks and the Transformer Encoder mechanism. An experiment was performed on six Deep Learning pre-trained models: DenseNet201, Xception, VGG16, GoogleNet, GoogleNet, InceptionResNetV2 and EfficientNetB7 and among them best models were selected and concatenated for

better feature extraction and accuracy. The selected Deep Learning models were used to build a ensemble learning backbone to extract meaningful images in two different cases ensemble A (i.e., DenseNet201, VGG16, and GoogleNet) and ensemble B (i.e., DenseNet201, InceptionResNetV2, and Xception). In addition, for diseases classification transformers with self-attention mechanism and multilayer perceptron (MLP) were applied for precise diseases classification. Explainable AI was used for localizing the most significant portion from images which was used during classification decisions. Two datasets namely Mendeley and Chest were used during model training, testing and validation phases.

For a generalized application of ViT in the medical domain, authors of the paper [47], proposed a hybrid architecture of ViT blocks and conv blocks to grab local and global dependencies. Also, Patch Momentum Changer (PMC) was proposed which was an augmentation technique to deal with adversarial attacks.They stacked a collection of repeatedly placed Efficient Convolution Blocks and Local Transformer Blocks. Efficient Convolution Block was composed of a Multi-Headed Self Attention and Locally Feed Forward Network. The MHSA and LFFN achieved multi-scale information. MCHA captured the global and LFFN captured the local dependencies.Local Transformer Block focused on global aspects as it contained the ESA (Efficient Self Attention) block which applied the multi-headed self attention mechanism. Further blocks of MHCA and LFFN were also involved in the LTB.Patch Momentum Changer: a unique augmentation technique which was proposed in this paper. It used the concept of Mixup Augmentation but not at the input image level rather at a feature level. The augmentation technique was applied on the feature maps which produced a more smoother decision boundary to make it robust to adversarial attacks.MedViT achieved the best results on most of the MNIST datasets with accuracy above 90% which exceeded many SOTA technologies.

According to Wang et al. [87] their proposed architecture Pyramid Vision Transformer (PVT) can function computer vision with lower computational cost unlike the ViT model. They claimed that their architecture can replace CNN based architecture for tasks requiring dense predictions as they leverage the benefit of both CNN and Transformer. Unlike ViT, it is designed for working on various sizes of input resolution or feature map, lower computation cost as it uses SRA (Spatial reduction attention mechanism). The pyramid architecture has four stages for extracting various feature scales and is heavily inspired by CNN like architecture. Each stage has a patch embedding layer and a transformer encoder.The MHSA mechanism is replaced by spatial-reduction attention in the transformer encoding layer which minimises the spatial scale of key and value before passing it to the attention layer. As a result, the computational cost and memory requirements significantly get reduced which helps to deal with larger input feature dimensions in resource limited settings. Moreover, PVT also differs from CNN backbone networks because it employs patch embedding layers to adjust the size of feature maps, where CNN uses various convolutional strides to minimize the scale of feature maps. Furthermore, the weight parameter of convolution is fixed for different input images whereas in the PVT the weight parameter of attention changes dynamically allowing a more flexible and expressive image classification. PVT-Small achieves 40.4 Average Precision which outperforms ResNet having 36.3 Average Precision and PVT-Large acquires 42.6 AP while ResNet has 4.1 AP.

Liu et al. [83] suggested a hierarchical transformer called Swin Transformer which substitute conventional Multi-Head Self Attention layer with shifted windows mechanism for vision related tasks with an accuracy of 87.3 top-1 accuracy on ImageNet-1K for classifying images, 58.7 box average precision and 51.1 mask average precision on COCO test dev for object detection. The architecture begins with first splitting the input RGB images into patches and passing the patches in the initial transformer blocks. The hierarchical structure is built using four swin transformer blocks which apply a shifted window approach in each block. In shifted window, the images are first split into non overlapping patches and at each patches self-attention is applied individually within each window.Afterwards, the window is shifted both horizontal and vertical for cross window interaction which allows information to pass between neighbouring windows.To maintain the hierarchical structure of CNN it uses patch merging layers at each transformer blocks to reduce the token size as it gets deeper in the model that leads to less computational costs.

Mehta et al. [82] presents the MobileViT architecture, a novel hybrid approach that merges the local inductive biases of Convolutional Neural Networks with the global attention capabilities of Vision Transformers. MobileViT introduces a unique MobileViT block that replaces the local processing step in convolutions with a global self-attention mechanism, thereby enabling efficient global context encoding while maintaining CNN-like properties. In the MobileViT architecture, MobileNetV2 blocks and MobileViT blocks are strategically arranged to handle different aspects of feature extraction and processing. This arrangement allows the MobileNetV2 blocks to handle lightweight, efficient local feature extraction, while the MobileViT blocks integrate critical global context at various levels, balancing computational efficiency and representation power. This architecture is applied across ImageNet-1k dataset, where it achieves a top-1 accuracy of 78.4% with only 5.6 million parameters, outperforming both CNN-based MobileNetv3 and ViT-based DeIT by 3.2% and 6.2%, respectively. MobileViT's limitations lie in its relatively higher computational cost on mobile devices compared to optimized CNNs, but it compensates with better generalization, scalability, and robustness to hyperparameter variations. The novelty of MobileViT lies in its efficient hybridization of CNN and ViT principles, achieving light-weight, general-purpose, and mobile-friendly performance.

Mehta et. al. [81] also introduces MobileViTv2 with separable self-attention that has less time complexity and is more efficient in resource constrained devices than mobileViT model. In MobileViT architecture they use multi-head self attention which have a complexity of $O(K^2)$ whereas in MobileViTv2 have a separable self-attention mechanism with linear complexity of O(K). Instead of comparing all input tokens with each other which happens in MHSA, separable self-attention selects a special token known as latent token. Then each token is compared with latent token and a context score is given to each token with respect to latent token. Afterwards using these score tokens are re-weighted and a context vector is generated to underline the global meaning from the input. At first the input image is divided into three parts input Query Q, Key K, Value V. Each token interacts with the latent node using a linear projection which helps to generate scalar distance.This step significantly minimizes the cost of computing attention score from $O(K^2)$ to O(K). The distances are normalized using softmax to compute context score. Then using the context score, the context vector is produced and after combining it with the token

representation it is passed to the transformer for the final output. The MobileViTv2 block differs from the MobileViTv block in two ways: (1) it substitutes multi-headed self-attention with separate self-attention to acquire global representations, and (2) it avoids the use of fusion blocks and skip-connections, which barely increase performance. The proposed architecture has 3 million parameters and acquires an accuracy of 75.6% in the ImageNet dataset.

Another lightweight ViT architecture was proposed by the authors of the paper [48], which was optimized for use in Federated Learning OnDev-LCT (On Device Lightweight Convolution Transformer). The input image was first fed into an LCT tokenizer instead of patch-based tokenization. LCT first converted the image into convolutional patches before sending them to the encoder. The $3 \times 3$ conv with ReLU activation and batch normalization helped to grasp the local dependencies, and then after reshaping, the outputs were forwarded to the encoder. The encoder contained the usual structure of MHSA with Query, Key and Value to apply the self-attention mechanism to retain the global dependencies. Dropouts and a dense layer were also added to the encoder, and before each operation, Layer Normalization was applied. The output from the encoder was passed to a pooling layer which was sent to a classifier (SeqPool) and then forwarded to a single layer FC to obtain the final output. LCT achieved 62.9% on CFAR-100 which was better than many lightweight cnn and vit architectures.

Ren et al. [49] presented RMT-Net, a novel model created by combining ResNet-50 with a transformer. While the transformer was used to gather data on long-distance features, RMT-Net uses ResNet-50 and depth-wise convolution to capture local features, minimize computation costs, and accelerate the detection process. The model had four stage blocks for feature extraction. The first three stages used global self attention techniques to extract important information of features and the last stage extracted the details of features using the residual clocks. A fully connected layer and a pooling layer with a global average were used for classification. The initial stage involved using ViT for early global feature inference, while the second stage involved using the Visual Transformer module to accomplish both simultaneous network parameter reduction and global feature modeling. Additionally, RMT-Net surpassed these architectures with a test accuracy of 97.65% on the X-ray image dataset and 99.12% on the CT image dataset when compared to ResNet-50, VGGNet-16, i-CapsNet, and MGMADS-3. Furthermore, the RMT-Net model could identify CT and X-ray images at a rate of 4.12 ms and 5.46 ms per image, respectively, while only having a 38.5 M size. As a result, they asserted that their suggested architecture can more precisely and efficiently categorize and detect COVID-19.

Where traditional computer vision techniques and CNNs have faced challenges in distinguishing pathological features in cases with subtle differences in lung textures, such as those with mild symptoms, the proposed model of Wang et al. [50] proves to be promising in such cases. Pneunet was a Deep Learning method based on ResNet18 for feature extraction and ViT for spatial feature detection in CXR images for COVID19 pneumonia diagnosis. Pneunet introduces a novel approach to pneumonia diagnosis by leveraging ViT technology, utilizing channel-based attention within CXR images to extract meaningful feature maps directly from the original images, enhancing the classification process. Experimental results demonstrate the

model's high accuracy rates in COVID-19 pneumonia diagnosis, outperforming existing Deep Learning methods. It had a test accuracy of 94.96% in three-category classification and 99.30% in binary classification, as well as promising performance in four-category classification, achieving an 86.94% prediction accuracy across COVID-19, none pneumonia, bacterial pneumonia, and viral pneumonia categories. However, the paper emphasizes the importance of dataset expansion to further improve the model's generalization capabilities and robustness. Overall, PneuNet shows promising results in accurate and efficient pneumonia diagnosis, and had the potential for enhancement through dataset diversification and clinical validation.

Graham et al. [79] proposed a novel architecture LeViT based on a hybrid neural network for image classification.The structure has been created based on the ViT model and DeiT training method. This is an hierarchical structure with three stages of transformer blocks each consisting of MHSA (Multi-Head Self-Attention), FFN(Feed-Forward Networks), normalization and downsampling token. Initially for patch embedding mechanism in transformer LeViT has used 4 layers of $3 \times 3$ convolution with stride 2 to the input images which helped to improve the model performance. Unlike ViTs they have replaced classification tokens with an average pooling layer for image embedding on the last activation map similar to CNN and pass to the transformer blocks. The parameters of LeViT-256 architecture are 18.9M. The architecture has achieved 80% accuracy top-1 accuracy in ImageNet. Furthermore, they claimed that LeViT is 1.5 to 5 times faster than existing neural networks for feature extraction.

Although the de facto convolution operation which is based upon spatially agnostic and channel specific filters have shown remarkable performance in image classification tasks, Li et al. proposed a novel framework with more efficiency in their paper [91]. They invented the basic operation of convolution and designed their feature extracting kernels to be spatially specific and channel agonistic. The core mathematical concept of Involution is presented in Equation 4.2 of this paper. Involution mitigated the problem of redundancy across channels, lowered the parametre count and overcame the difficulty of modeling long-range interactions. The authors followed the original structure of Resnet by stacking residual blocks after each layer. They replaced Involution for $3 \times 3$ convolution at all bottleneck positions in the stem (using $3 \times 3$ or $7 \times 7$ Involution for classification or dense prediction) and trunk (using $7 \times 7$ Involution) of ResNet, but retained all the pointwise or $1 \times 1$ convolution. They termed the new architecture as RedNet. Image classification on ImageNet dataset, Object Detection, Instance Segmentation and Semantic Segmentation were performed. In this paper, we heavily reviewed the performance achieved in the image classification task which gained a 79.3% accuracy with RedNet152. The authors closely studied the reason behind the popularity of attention mechanisms in recent times. They anticipated further advancement of neural architecture engineering focusing on simple yet effective models.

Chaudhary et al. [86] proposed an architecture called InvolutionNet based on Involution operation to detect melanoma skin cancer with an accuracy of 87%. Although CNN has shown immense potentiality in medical image processing, there are some drawbacks of CNN including computational complexity, inefficient long range dependency modeling and the necessity for large data augmentation.The dataset was

collected from different publicly available sources. To mitigate these issues they have proposed the InvolutionNet model. Unlike attention-based networks, they do not require large quantities of data for optimal performance. Three Involution blocks have been stacked together to extract features from the images. Initially the dimension of the input layer is $256 \times 256 \times 3$ which gradually reduced in each layer and at the end of the third layer the dimension becomes $32 \times 32 \times 3$. After the features have been extracted it flattened and passed onto the classification branch for binary classification. For future exploration the authors have been suggested to use attention mechanisms with InvolutionNet architecture and also transfer learning approach with InvolutionNet model.

In recent study, Pradhan et al. [90] addressed issues for detecting plant diseases and proposed a solution based on digital image of plant leaves analysis method for identifying and classifying plant diseases utilizing an Involution neural network and self-attention based model. Their proposed architecture is designed using one MobileNetV2 block, one MobileNet Involution block, and two basic blocks each composed of a $3 \times 3$ convolutional layers. The MobileNetV2 block consists a group of four basic blocks and the kernel sizes are $3 \times 3$, $1 \times 1$, $3 \times 3$, $1 \times 1$. The MobleNet Involution block has four convolution layers. Furthermore, Self-attention mechanism extracts critical features, improves classification accuracy by focusing on relevant image regions. They have collected a public dataset from PlantVillage to train the proposed model and gained an accuracy of 98.04% for 8 crops having 23 classes, presenting its ability to identify and recognize plant diseases.

Abuhayi et al. [88] proposed Inv-AlxVGGNets architecture which utilizes Involutional Neural Networks merged with two well-known neural network designs, AlexNet and VGGNet, along with Residual Networks. Traditional CNNs have more than 133 million parameters whereas the proposed architecture needs less than 8 million parameters, making it highly suitable for resource-constrained environments and limited datasets. Involution is specifically designed to capture long-range dependencies within feature maps and learn dynamic convolutional patterns, while significantly reducing the parameter count without compromising performance. Additionally, the architecture incorporates a VGG-inspired branch, which leverages the detailed and extensive feature extraction capabilities of VGG, enhanced with residual layers to preserve gradients and improve learning. Both outputs of AlexNet and VGG are merged and then passed to fully connected layers for final classification. By merging these two models serves great benefits such as input features enhancement as it mixes the features resulting from both models, improving performance as it enables the model to leverage the strength of both models, increasing the model capacity to learn more complex features. Inv-AlxVGGNets obtains 98.73% accuracy on testing and 99.78% on training data. The primary goal of the study was to address the high parameter requirements of CNNs by replacing them with Involution, which significantly reduces parameters while maintaining high performance.

## 2.2 Federated Learning Approach

Centralized implementation of Deep Learning provided great accuracy in classification tasks for disease diagnosis in various sectors. However, implementation of these

models would be impractical in real world scenarios as institutions are unlikely to share data due to privacy concerns. Hence, a system of decentralized training known as Federated Learning came into the equation. Figure 2.1 shows an overview of the Federated Learning system. The authors in paper [51] proposed the first solution to this problem in their research with the novel Federated Averaging Algorithm which was expressed in the equation 2.4.

$$W_{t+1} \leftarrow W_t - \eta \sum_{k=1}^{k} \frac{n_k}{n} g_k \tag{2.4}$$

Where $\eta$ was the learning rate, $k$ was the client number ,$n$ was the size of total dataset and $n_k$ was the dataset size of client $k$, $w_t$ was the current weight, $g_k$ was the average gradient on a local client and $w_{t+1}$ was the updated weight. They ran experiments on the CFAR-10 and MNIST datasets which ensured privacy but the final accuracy achieved was 96%. Decentralized system adds various problems such as the communication cost, non iid datasets and accuracy concerns etc. New and impactful research was carried out to address these problems.In [52], the authors dealt with communication efficiency by proposing new ideas: Sketched updated and Quantization. These techniques simplified the local updates of the clients but at an expense of lower accuracy of 85%. The problem of privacy preserving was handled by the authors of the paper [53] in their research, where they implemented a novel DSSGD algorithm on MNIST dataset which converges slower than centralized SGD but eventually reaches similar accuracy upto 99%. Authors of the paper [54], found one of the key factors was the use of Batch Normalization which significantly affected the accuracy in FL systems. Instead, using Layer Normalization or Group Normalization provided remarkable improvements in accuracy.To address the problem of Non-IID datasets, many solutions were proposed by various authors. In one of the recent researches, the authors of the paper [55] proposed their Aorta architecture where they handled the problem with augmentation. It helped to resolve latency in updates from faster and slower clients. In another research, the authors of the [56] employed knowledge distillation technique in their FedTweet architecture which also handles adversarial attacks. Authors of another paper [57] used representation augmentation to handle the non-IID datasets. Recent works on communication efficiency showed great improvements to boost the system performance. One such implementation was proposed by the authors of the paper [58] where they used cosine similarity between the global model and the local clients to determine a coefficient of client's contribution. The inclusion of the coefficient helped to handle the client contribution and improved the overall accuracy. In another research, the authors of the paper [59] proposed a selective aggregation approach for client's updates. They introduced the idea of dropped gradients which was a similar effect like Dropout. In dropped gradients, clients which contributed were selected with a probability and this change helped to address bottleneck issues and achieved higher accuracy. Authors of another paper [60], proposed a Federated Learning approach in the medical domain for CXR image classification to detect Covid-19. They used RestNet50 and VGG16 architecture and compared that centralized implementation verses the decentralized implementation had very minimal effect on the accuracy. Overall, complete large system designs were also proposed by the authors of the paper [61]. They proposed a well defined implementation of the server,local client systems and the process of round updates to train the global model.

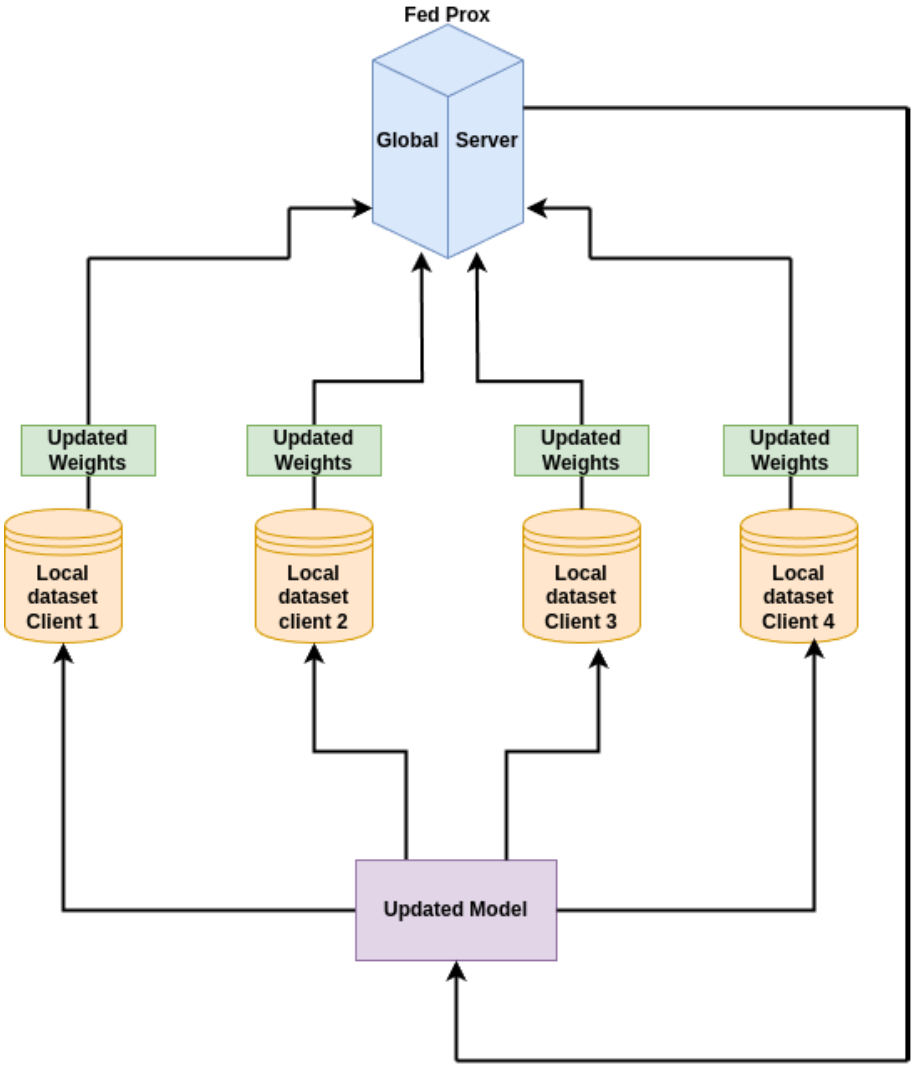


Figure 2.1: Federated Learning Architecture

In table 2.1 we summarise findings from the related works that we reviewed.

| Ref No | Dataset | Models | Accuracy |
|---|---|---|---|
| [28] | Public | Ensemble Model: EfficientNet-B0, VGG-16 and DenseNet201 | 97% |
| [33] | ImageNet | EfficientNet-B0, EfficientNet-B1 and EfficientNet-B2 ensembled | 98% |
| [36] | Public | Vision Transformer | 97% |
| [39] | Public | Vision Transformer Regressor Infection Prediction (ViTReg-IP) | 88% |
| [62] | Public | DBM-ViT | 97% |
| [63] | ImageNet | ViT and VDSNet | ViT: 70%; VDSNet: 69% |
| [45] | Public | CoViT | 99% |

| Ref No | Dataset | Models | Accuracy |
|---|---|---|---|
| [64] | Public | Neural Network architecture composed of two parts: 1) Inception architecture 2) Inception modules and a Vision Transformer network | 73% |
| [27] | Public (from Kaggle) | VGG-16 and Neural Network | 94% |
| [29] | Public datasets of varying size | VGG-19 + CNN | 96% |
| [65] | Public | Fine tuned VGG-16 | 95% |
| [66] | C19RD, CXIP (from Kaggle) | RNN-LSTM, SVM, ANN, KNN and ensemble classifier | C19RD: 95%,88%,91%,86%,88%; CXIP: 94%,87%,86%,83%,84% |
| [40] | ImageNet-1k | Fully Adaptive Transformer (FAT) | 77% |
| [46] | MNIST | MedViT | Above 90% |
| [37] | ImageNet-1k | LITv1 | Top 1% : (83%) |
| [38] | ImageNet-1k | LITv2 | Top 1% : (84%) |
| [41] | ImageNet-1k | FastViT | 84% |
| [48] | CFAR-100 | OnDev-LCT | 63% |
| [22] | Public | Lightweight CNN | 98% |
| [24] | Public | 2D ResNet50 | RF-97%, SVM-96%, BNN-96%, Adaboost-97%, DNN-99% |
| [26] | Public | VGG19+VGG16 | 92% |
| [30] | Public | VGG19+CNN model | 98% |
| [32] | Public | CNN +GRU+XAI | Covid-19: 99%, Lung cancer: 99% |
| [42] | Public | ARSeg, TransM, Medical Transformer (MedT), TransUNet, and UNeXt | 97%,98%,97%,96%,95% |
| [44] | Public | DINO ViT | 94% |
| [44] | ImageNet | CNN + Transformer | 83% |
| [49] | Public | Model: RMT-Net (ResNet50 +ViT+VT) | Xray_dataset: 97%, CT Scan_dataset: 99% |

| Ref No | Dataset | Models | Accuracy |
|---|---|---|---|
| [70] | Public dataset | EnsembleA (DenseNet201, VGG16, and GoogleNet)+tranformer + XAI, Ensemble B (DenseNet201, InceptionResNetV2, and Xception) +tranformer+XAI | 97%, 96% |
| [71] | Public dataset | CovidViT | 98% |
| [73] | ImageNet Dataset | Deep CNN | top-1% : (37.5%), top-5% : (17%) |
| [72] | ImageNet-1k | DeiT (Data efficient Image Transformer),ConViT | 79%,81% |
| [74] | ImageNet Dataset | CvT-13, CvT-21, CvT-W24 | 83%, 85%, 88% |
| [23] | Public Dataset 1 from Kaggle and Dataset 2 from Github | Custom-CNN | Dataset 1: (98%) Dataset 2: (99%) |
| [88] | Private Dataset | Inv-AlxVGGNets | 98.73% |
| [76] | ImageNet | SqueezeNet | top-1 : 57.5%, top-5 : 80.3% |
| [81] | ImageNet | MobileViTv2 | 75.6% |
| [79] | ImageNet | LeViT | top-1 : 80% |
| [89] | ImageNet | DeiT | 85.2% |
| [90] | Public | INN | 98.04% |
| [80] | Public | EfficientNet-B0 | Top-1:77.2%, Top-5: 93.4% |
| [86] | Public | InvolutionNet | 87% |
| [83] | ImageNet-1K | Swin Transformer | 87.3% |
| [87] | COCO dataset | PVT small, PVT large | 40.4 AP, 42.6 AP |
| [85] | ImageNet | BoTNet | 84.7% |
| [84] | ImageNet-1k | EdgeViT-S, EdgeViT-XS, EdgeViT-XXS | 81%, 77.5%, 74.4% |
| [82] | ImageNet-1k | MobileViT | 78.4% |
| [78] | ImageNet | MobileNetV2 | 72% |
| [91] | ImageNet | RedNet-152 | 79% |

| Ref No | Dataset | Models | Accuracy |
|---|---|---|---|
| [95] | ImageNet, ImageNet-ReaL and CIFAR-100 | ViT-H/14 | 90.07% on ImageNet-ReaL, 94.5% on CIFAR-100 and 89% on ImageNet |
| [92] | ImageNet | MobileNet-224 | 70.6% |

Table 2.1: Summary of Related Works

## 2.3 Research Gap

In the previous section, we thoroughly studied state of the art models. The gaps identified in these studies are outlined below.

- Most research have not worked with small datasets. In a real-world context, availability of large medical datasets is very much difficult due to privacy concerns of the institutions.
- Pre-processing techniques such as advanced histogram equalization can be an efficient solution to produce higher accuracies which is also not employed in most of the previous works.
- Recent studies did not ensure a proper trade-off between accuracy and number of parameters in their work.However, light weight models are better suited for resource constrained environments like hospitals.
- Recent studies fail to propose a complete decentralized system that can ensure the data-privacy of institutions.

# Chapter 3

# Dataset

This section of the paper summarizes the purpose, pre-processing methodology, and significance of the dataset.

## 3.1 Description of the Dataset

We have collected a primary dataset of 1186 Chest X-Rays for our research. Chest X-ray is used to evaluate the condition of lungs and helps to diagnose and monitor treatment for a variety of pulmonary diseases. By reviewing different existing papers, we observed that most of them have utilized existing datasets focusing on conventional classes like Covid, Pneumonia and Tuberculosis. Although this kind of classification was proved to be beneficial in providing health insights, it may not be fully relevant to capture the pattern of post-covid respiratory complications. For bridging this gap in the study, we collected a primary dataset composed of the latest image data of 2024. Our dataset captures the contemporary medical scenarios which will provide the opportunity to study the emergent effects of the COVID-19 pandemic. Moreover, we have classified our data into three labels that cover a broad range of general pulmonary diseases. Hence it will ensure greater applicability to daily-life diagnosis.

The data collection process enabled us to observe and understand the real-life diagnosis conducted in a radiology unit. We talked with different medical professionals to understand how diseases are detected from x-rays, and what kind of adjustments are made in the display for making the quality better. This kind of practical learning allowed us to acknowledge the real life situations faced by clinicians and deepened our overall knowledge about our research methodology.

Furthermore, by collecting an up-to-date and fresh dataset we could capture the health information about the current population. Our dataset also unfolds the underlying disease trend among them. This can be a valuable resource for conducting region specific research and providing aid to diagnosis of the evolving health complications in the future. For adding more scopes in the future work, each image of our dataset was encoded with gender and age of the patient. The entire encoding process was done by keeping the privacy concerns in mind. As a result, our primary dataset containing fresh and never-seen-before data will open scopes for future analysis and findings.

## 3.2 Overview of the Dataset

Our primary dataset contains 1186 validated Chest X-Ray images.All the images were classified into 3 classes, which are:

1. Normal
2. Pneumonia
3. Abnormal

**Description of the Data:**
We collected RGB images of chest x-ray. The dimension of images was $1024 \times 1024$.

**Description of the Classes:**
Normal class includes the X-rays which are not infected with any kind of disease. Except the aforementioned class, all other images with any health issues were grouped into the rest two classes. The images in which Pneumonia was prevalent were grouped in Class-Pneumonia. Moreover, there are other serious pulmonary diseases. All of those were included in Class-Abnormal. The Figure 3.1 below shows the sample images of the depicted classes from our dataset.

(a)Normal (b)Abnormal (c)Pneumonia

Figure 3.1: Original Images from our Dataset

## 3.3 Data Collection Methodology

The images of our primary dataset consist of Chest x-rays of patients starting from 1 day to 85 years old. We collected the data from the radiology and imaging department of Japan Bangladesh Friendship Hospital. We collected the images within the span of four days. The timeline of our data collection is given below:

1. 02/02/2024, starting time: 11:34 am and ending time: 1:17 pm
2. 09/02/2024, starting time: 03:14 pm and ending time: 06:15 pm
3. 10/06/2024, starting time: 02:30 pm and ending time: 02:57 pm
4. 05/10/2024, starting time: 10:51 am and ending time: 12:06 pm

(a) Computer Display

(b) X-Ray System

Figure 3.2: (a)Computer Display and (b)X-Ray System Machine

The X-ray films were displayed in the laboratory computer by the Laboratory Assistants. We intensively discussed the x-ray films and the pattern of diseases observed in each of the images of our dataset with them. They gave us insights about different pulmonary complications, how labeling is done by radiologists and other clinical situations. For capturing the x-rays, Fujifilm FDR Smart-F X-ray system machine as mentioned in Figure 3.2 (b) was used. And the X-rays were viewed on Eizo and DELL computer displays. Moreover, we used the camera of our smartphone(Samsung-A52) to capture the images. While capturing the image, we selected the ones with a vivid view of the Lungs. Any hazy or distorted films were not considered.

## 3.4 Preliminary Data Analysis

Once the collection of the dataset is complete, the first and foremost step is always to analyze the data before starting to take any further steps. As a result, carrying out a detailed analysis is essential. We analyzed our CXR image dataset through various techniques like statistical methods, diagrams, plots and also through manual inspection. Our key findings are listed below:

1. Class Distribution: Table 3.2 shows the distribution of images in each class. It is evident from the statistics that there is a class imbalance as the proportion of normal class is much higher than the other two classes. Hence this disparity needs to be adjusted to ensure a fair evaluation of analysis of the Deep Learning models.

2. Missing Files: All the collected data has been checked thoroughly for corrupted files by running Python scripts to ensure that all the files are perfectly ready to be fed into the architectures. No corrupted files were found.

3. Cropping and Adjustments: Images needed cropping to remove the unwanted regions so that the models can focus mainly on the necessary parts. So, unnecessary noise were removed and all the images were resized to a shape of $1024 \times 1024$.

4. Histogram Analysis: A histogram of pixel intensity values is necessary to gain more insights about the distribution of pixel values hence we used "matplotlib" library from Python to generate the histogram of a randomly selected image. The histogram is shown in Figure 3.3. It is evident from the histogram that the pixel values are more skewed towards the brighter end of the spectrum and hence a Contrast Limited Adaptive Histogram Equalization technique would be necessary to make the distribution more uniform and improve the contrast and clarity of the image.

5. Labeling Consistency: For ensuring the accuracy and reliability of the dataset, labeling was conducted by a medical professional. The doctor has an MBBS degree and is currently posted as an Assistant Surgeon in a government hospital. Each of the images was annotated with great expertise and hands on experience of the doctor which enhanced the quality of our dataset and made it qualified enough to use in real life scenarios.

6. Outliers check: Mean and Standard deviation were calculated across all the images of our dataset and the values were found to be:
   Mean= 132.86
   Standard Deviation: 52.08
   A threshold of 3 was set for outlier calculation where lower and upper bounds are defined as equation 3.1 and 3.2:

$$lower\_bound = mean - threshold \times standard\_deviation \tag{3.1}$$

$$upper\_bound = mean + threshold \times standard\_deviation \tag{3.2}$$

   Any value outside this range was considered to be an outlier and our test showed that there were no outliers proving that there is sufficient uniformity in the collected images.

7. Other Consideration: Another important observation that was found through visual inspection is the presence of CXR of both adults and children who are less than 8 years old. Generally the size of the chest and lungs is much larger for an adult compared to a child so there is a lot of variation in the image

structure, even though there are no outliers. As a result, it can be difficult for the model to learn the features of a child's CXR to identify normal, abnormal or pneumonia classes. This problem can be dealt with by resizing the children's CXR to a larger shape but it can degrade the quality of an image. Another approach can be used which is to load the images of children more frequently than the ones of adults, so that the model learns more of the children's CXR structures but it can lead to overfitting as the quantity of images of children is significantly less than that of adults. Table 3.1 shows the distribution among adults and children:

| **Group** | **Image Count** |
|---|---|
| Children below 8 | 105 |
| Adult | 1081 |

Table 3.1: Image Distribution In Terms of Age Group

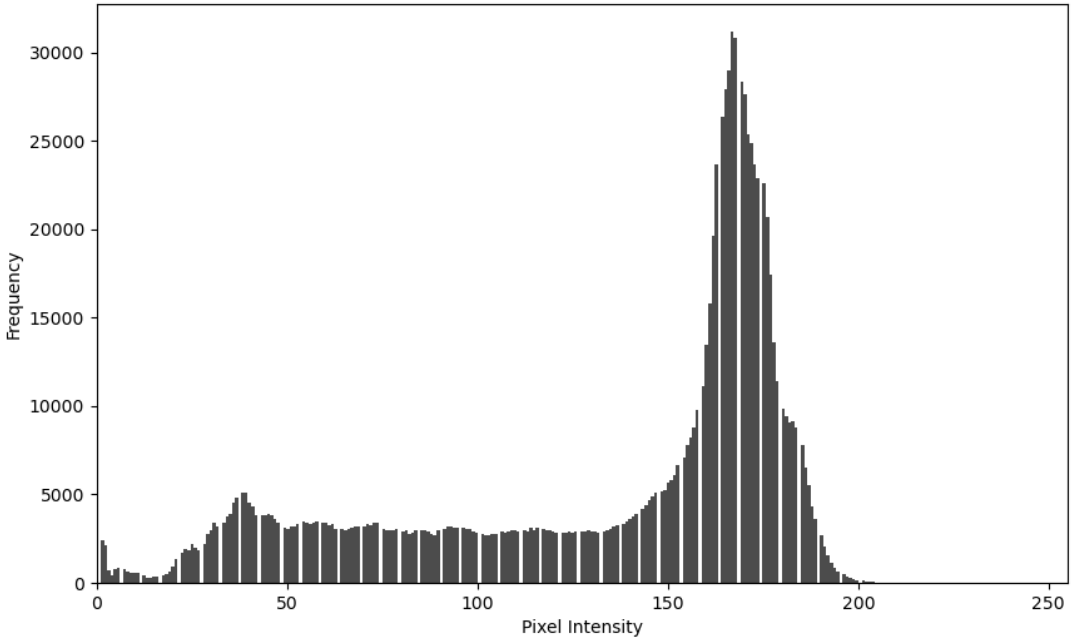


Figure 3.3: Histogram of Pixel Intensity Values of a CXR Image

## 3.5 Preprocessing

In the field of Machine Learning, preprocessing plays an important role as the success of architecture depends heavily on the quality of input data. Shortage of data, presence of noise and outliers can significantly affect the outcome of a system.In order to mitigate these challenges and to further boost the performance of the models, researchers have relied on the idea of preprocessing. It enables to elevate the standard of raw data so that the task becomes more efficient and reliable for the architectures to provide more accurate results. In the field of Deep Learning, preprocessing plays a vital role as the spatial information of an image is the core component to achieve any form of detection, classification and segmentation. In our

context the fine details of CXR images are more important as the specific spatial patterns help to classify different diseases. Also, feature extraction is dependent on spatial hierarchies as it enables preserving more details that can help to achieve higher accuracy. Furthermore, shortage of data can lead to insufficient training. To overcome these problems, we have taken the following steps to achieve better results:

1. Converting from RGB to Grayscale
2. Applying ( CLAHE : Contrast Limited Adaptive Histogram Equalization)
3. Augmentation

### 3.5.1 RGB to Grayscale Conversion

The raw images in the dataset are RBG .However, their conversion to grayscale can help to reduce the number of channels which reduces the amount of processing and also enhances the focus on image features to detect abnormalities for classification. As a result color noise is removed which is unnecessary in this classification context. We used the built-in "transforms" library from pytorch where the conversion to grayscale is implemented using the equation 3.3. Here R,G and B are the Red, Blue and Green channel pixel intensity values respectively.

$$Gray = 0.2989 \times R + 0.5870 \times G + 0.1140 \times B \tag{3.3}$$

### 3.5.2 CLAHE

It is a contrast enhancement technique which utilizes Histogram Equalization to improve the image quality. Histogram Equalization is a technique where the histogram of an image is equalized to produce a more uniform distribution that improves the contrast of an image. This method is improved in CLAHE as it applies this technique in certain local regions instead of the entire image and it also keeps a threshold known as clip limit which redistributes any pixel that exceeds the threshold. The local region is defined with the help of a fixed kernel size. In our implementation, we have kept kernel size at $8 \times 8$ and clip limit=2. This improvement in quality helps the model to extract more spatial details from a CXR image.
The outcome of applying this technique is illustrated in the Figure 3.4.

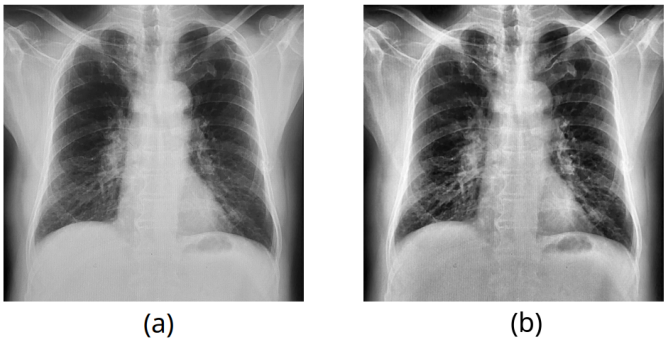


Figure 3.4: (a) Original Image and (b) Processed Image(CLAHE)

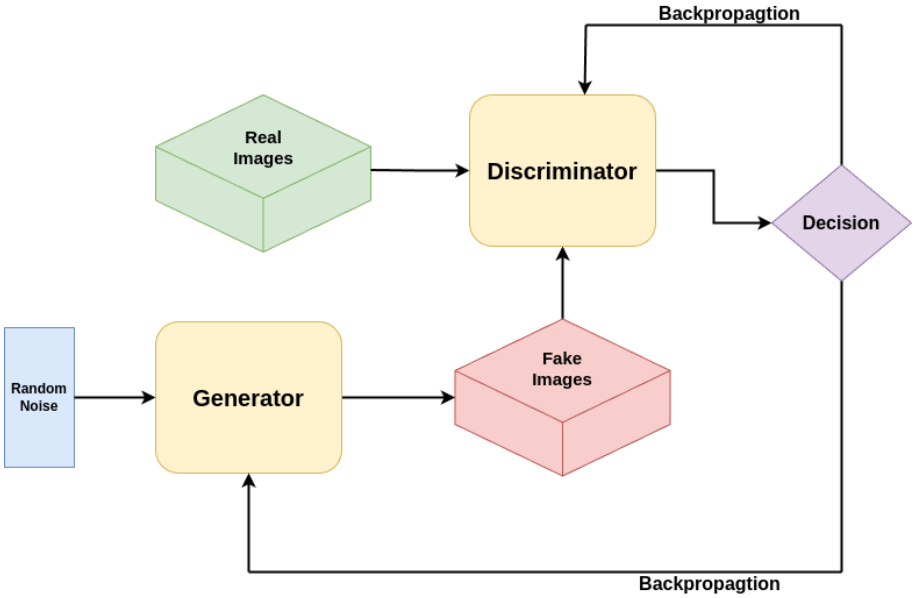


Figure 3.5: Overview of GAN Architecture

### 3.5.3 Augmentation

Augmentation is the most widely used method to increase the size of a dataset by generating more data. It leads to a more efficient training for Deep Learning models. Augmentation can be applied by using various methods that include Random Rotation, Horizontal/Vertical Flips etc. However, in real-life applications it is unusual to test a Deep Learning model with randomly rotated or flipped images. Generally CXR images are created and evaluated in a straight and upright structure. As a result, in our research we planned to implement the state of the art techniques known as GAN (Generative Adversarial Networks) [75] to generate fake images that can significantly help to increase the quantity of CXR images. GANs work with the idea of adversarial training where there are two models known as Discriminator and Generator and they participate in a similar situation like a Min Max game. The Generator tries to fool the Discriminator by generating fake images and the Discriminator tries to detect the fake images correctly. As a result the Generator tries to maximize the loss but the Discriminator tries to minimize the loss. The process runs for several epochs until the Discriminator assigns a probability of 0.5 to the fake images which indicates that the Discriminator is not certain about the image being real or fake. A general overview of GAN architectures is shown in Figure 3.5. The loss function is expressed in equation 3.4 where D,G are the Discriminator and Generator functions, $x$ is an input image and $z$ is a random noise .However to ensure good image quality and consistency we have implemented a PGGAN (Progressive GAN ) [77] which is a well designed architecture for stable training and generating a decent quality of image.Three images of Normal, Abnormal and Pneumonia classes which are generated by Progressive GAN architecture is shown in Figure 3.6. Class distribution of images generated by Augmentation process is represented by table 3.2.

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))] \tag{3.4}$$

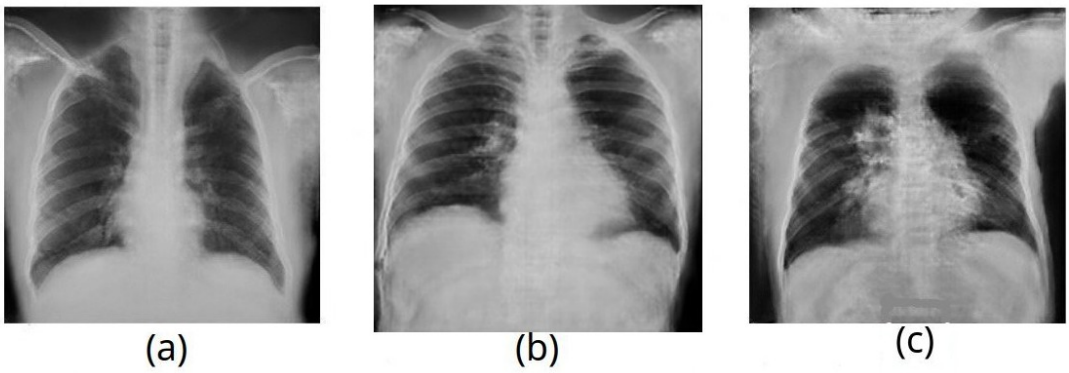


Figure 3.6: Images of Class (a) Normal, (b) Abnormal and (c) Pneumonia, Generated by Progressive GAN Architecture.

| | Normal | Pneumonia | Abnormal | Total |
|---|---|---|---|---|
| Raw | 622 | 219 | 345 | 1186 |
| GAN | 946 | 1134 | 749 | 2829 |
| Total | 1568 | 1353 | 1094 | 4015 |

Table 3.2: Augmented and Original Images Class Distribution

### 3.5.4 Progressive GAN Architecture

The core idea of PGGAN is developed around this central concept of Progression from lower size to higher sizes that helps to achieve a great quality. Initially the architecture starts to learn to produce small sized images starting from $4 \times 4$ and once it is successful in this particular size it grows to a larger size $8 \times 8$. Then it moves to $16 \times 16$ and so on. This is a fundamental concept in progressive growing. As the architecture tries to produce larger size images, its layers and parameters are also increased to learn more features. A key concept must be highlighted that when the architecture is working at a larger size ($128 \times 128$) for example, then the previous layers are not frozen but set to train. There are 4 key concepts that establish the success of this architecture:

1. **Progressive Growth:** Starting with low resolution and improving to higher.
2. **Mini Batch Standard Deviation:** To increase variation in the training data a standard deviation is calculated over the minibatch and it is used to create one feature map that is inserted towards the end of the discriminator.
3. **Pixel Normalization:** For every feature vector each pixel value is normalized across the channels by taking a squared average.
4. **Equalized learning rate:** During runtime, He's initialization it used to retrieve per layer.Normalization constant and the weights are scaled by that constant to ensure they are scaled as per their deviation from optimal values.

The generator and discriminator are mirror reflections of each other and they grow synchronously. If the discriminator moves from $16 \times 16$ to $8 \times 8$ to $4 \times 4$. Then the generator architecture will move from $4 \times 4$ to $8 \times 8$ to $16 \times 16$. Then a new layer is being added due to new resolution it is faded in gradually which is maintained with by a hyper-parameter $\alpha$ which varies between 0 and 1. This controls how much of the new layers will be faded in. As training progresses value of $\alpha$ increases towards 1. Both Generator and Discriminator uses $3 \times 3$ Conv layers and uses Leaky ReLu activation function with 0.2 leakiness. The detailed information of the layers is provided in the Figure 3.7. Upsampling and downsampling are carried out with $2 \times 2$ element replication and average pooling. The loss function used here is the WGAN-GP loss function.

| | Normal | Pneumonia | Abnormal | Total |
|---|---|---|---|---|
| Train | 1472 | 1232 | 966 | 3670 |
| Test | 96 | 121 | 128 | 345 |
| Total | 1568 | 1353 | 1094 | 4015 |

Table 3.3: Train and Test Distributions

In our implementation we trained the PGGAN initially on our benchmark dataset to ensure that the model learns the general structure of CXR images which serves as a good starting point for weight initialization. Then the architecture is trained on our dataset separately on respective classes for further 20 epochs to generate fake images. Samples from our PGGAN for different classes are given in Figure 3.7. We divided our dataset into training set and test set in a 90% to 10% ratio which is shown in table 3.3.

| **Generator** | Act. | Output shape | Params |
|---|---|---|---|
| Latent vector | – | 512 × 1 × 1 | – |
| Conv 4 × 4 | LReLU | 512 × 4 × 4 | 4.2M |
| Conv 3 × 3 | LReLU | 512 × 4 × 4 | 2.4M |
| Upsample | – | 512 × 8 × 8 | – |
| Conv 3 × 3 | LReLU | 512 × 8 × 8 | 2.4M |
| Conv 3 × 3 | LReLU | 512 × 8 × 8 | 2.4M |
| Upsample | – | 512 × 16 × 16 | – |
| Conv 3 × 3 | LReLU | 512 × 16 × 16 | 2.4M |
| Conv 3 × 3 | LReLU | 512 × 16 × 16 | 2.4M |
| Upsample | – | 512 × 32 × 32 | – |
| Conv 3 × 3 | LReLU | 512 × 32 × 32 | 2.4M |
| Conv 3 × 3 | LReLU | 512 × 32 × 32 | 2.4M |
| Upsample | – | 512 × 64 × 64 | – |
| Conv 3 × 3 | LReLU | 256 × 64 × 64 | 1.2M |
| Conv 3 × 3 | LReLU | 256 × 64 × 64 | 590k |
| Upsample | – | 256 × 128 × 128 | – |
| Conv 3 × 3 | LReLU | 128 × 128 × 128 | 295k |
| Conv 3 × 3 | LReLU | 128 × 128 × 128 | 148k |
| Upsample | – | 128 × 256 × 256 | – |
| Conv 3 × 3 | LReLU | 64 × 256 × 256 | 74k |
| Conv 3 × 3 | LReLU | 64 × 256 × 256 | 37k |
| Upsample | – | 64 × 512 × 512 | – |
| Conv 3 × 3 | LReLU | 32 × 512 × 512 | 18k |
| Conv 3 × 3 | LReLU | 32 × 512 × 512 | 9.2k |
| Upsample | – | 32 × 1024 × 1024 | – |
| Conv 3 × 3 | LReLU | 16 × 1024 × 1024 | 4.6k |
| Conv 3 × 3 | LReLU | 16 × 1024 × 1024 | 2.3k |
| Conv 1 × 1 | linear | 3 × 1024 × 1024 | 51 |
| Total trainable parameters | | | **23.1M** |

| **Discriminator** | Act. | Output shape | Params |
|---|---|---|---|
| Input image | – | 3 × 1024 × 1024 | – |
| Conv 1 × 1 | LReLU | 16 × 1024 × 1024 | 64 |
| Conv 3 × 3 | LReLU | 16 × 1024 × 1024 | 2.3k |
| Conv 3 × 3 | LReLU | 32 × 1024 × 1024 | 4.6k |
| Downsample | – | 32 × 512 × 512 | – |
| Conv 3 × 3 | LReLU | 32 × 512 × 512 | 9.2k |
| Conv 3 × 3 | LReLU | 64 × 512 × 512 | 18k |
| Downsample | – | 64 × 256 × 256 | – |
| Conv 3 × 3 | LReLU | 64 × 256 × 256 | 37k |
| Conv 3 × 3 | LReLU | 128 × 256 × 256 | 74k |
| Downsample | – | 128 × 128 × 128 | – |
| Conv 3 × 3 | LReLU | 128 × 128 × 128 | 148k |
| Conv 3 × 3 | LReLU | 256 × 128 × 128 | 295k |
| Downsample | – | 256 × 64 × 64 | – |
| Conv 3 × 3 | LReLU | 256 × 64 × 64 | 590k |
| Conv 3 × 3 | LReLU | 512 × 64 × 64 | 1.2M |
| Downsample | – | 512 × 32 × 32 | – |
| Conv 3 × 3 | LReLU | 512 × 32 × 32 | 2.4M |
| Conv 3 × 3 | LReLU | 512 × 32 × 32 | 2.4M |
| Downsample | – | 512 × 16 × 16 | – |
| Conv 3 × 3 | LReLU | 512 × 16 × 16 | 2.4M |
| Conv 3 × 3 | LReLU | 512 × 16 × 16 | 2.4M |
| Downsample | – | 512 × 8 × 8 | – |
| Conv 3 × 3 | LReLU | 512 × 8 × 8 | 2.4M |
| Conv 3 × 3 | LReLU | 512 × 8 × 8 | 2.4M |
| Downsample | – | 512 × 4 × 4 | – |
| Minibatch stddev | – | 513 × 4 × 4 | – |
| Conv 3 × 3 | LReLU | 512 × 4 × 4 | 2.4M |
| Conv 4 × 4 | LReLU | 512 × 1 × 1 | 4.2M |
| Fully-connected | linear | 1 × 1 × 1 | 513 |
| Total trainable parameters | | | **23.1M** |

Figure 3.7: Architecture Details Retrieved from PGGAN Paper

# Chapter 4

# Methodology

First of all, we begin by collecting primary CXR image data from local hospitals. All the collected CXR images are then labelled from a doctor specialized in the field of radiography and imaging. The images are classified into Normal, Pneumonia and Abnormal classes. Cases like effusion, cancer, chronic cough etc are all included in the Abnormal class. Next, to increase the size of dataset we use a ProGAN architecture which provides us sufficient images suitable for training Deep Learning models. All the images were then preprocessed from training by applying augmentation techniques like random rotation, color jitter and CLAHE. Then the InvoSparseNet architecture was built with proper adjustment of layers and sufficient fine tuning. All the settings used for the architecture is provided in table 4.1. For training and comparison 3 different approaches were taken. Firstly, we trained our model and also 5 other SOTA architectures and compared various metric to provide an in-depth analysis. Confusion-Matrix and AUC-ROC curve were also generated. Then we also tested all the models on a benchmark dataset [93] from Kaggle which was also used by other authors [94]. This step is taken to establish the credibility of our dataset and also justify the usability of our proposed model. Lastly, all the models were tested in Federated Learning environment that gives a representation of a real-world scenario and to analyze the behavior of the models in that context. Loss vs Rounds and Accuracy vs Rounds curves are shown in 5.3 which shows the convergence of the model over 10 rounds of communication. Finally, GradCAM was also used to add further explainability, which helped to visualize the underlying decision making procedure of our proposed model.

## 4.1 Proposed Architecture Description

Our study proposes a novel methodology for lung disease classification by leveraging the output features from depthwise convolution, Involution instead of traditional CNN, and global dependency capturing Sparse Attention mechanism. Residual connections are used in each layer for tackling the problem of vanishing gradient. These features are then concatenated to serve as input to fully-connected layers for classification. Our aim is to utilise the combined strengths of the aforementioned methods to achieve superior performance in medical image classification. Figure 4.1 provides a visualization of the proposed model. Furthermore, we will provide a detailed overview of the architectural components of our proposed model, InvoSparseNet in the following section:

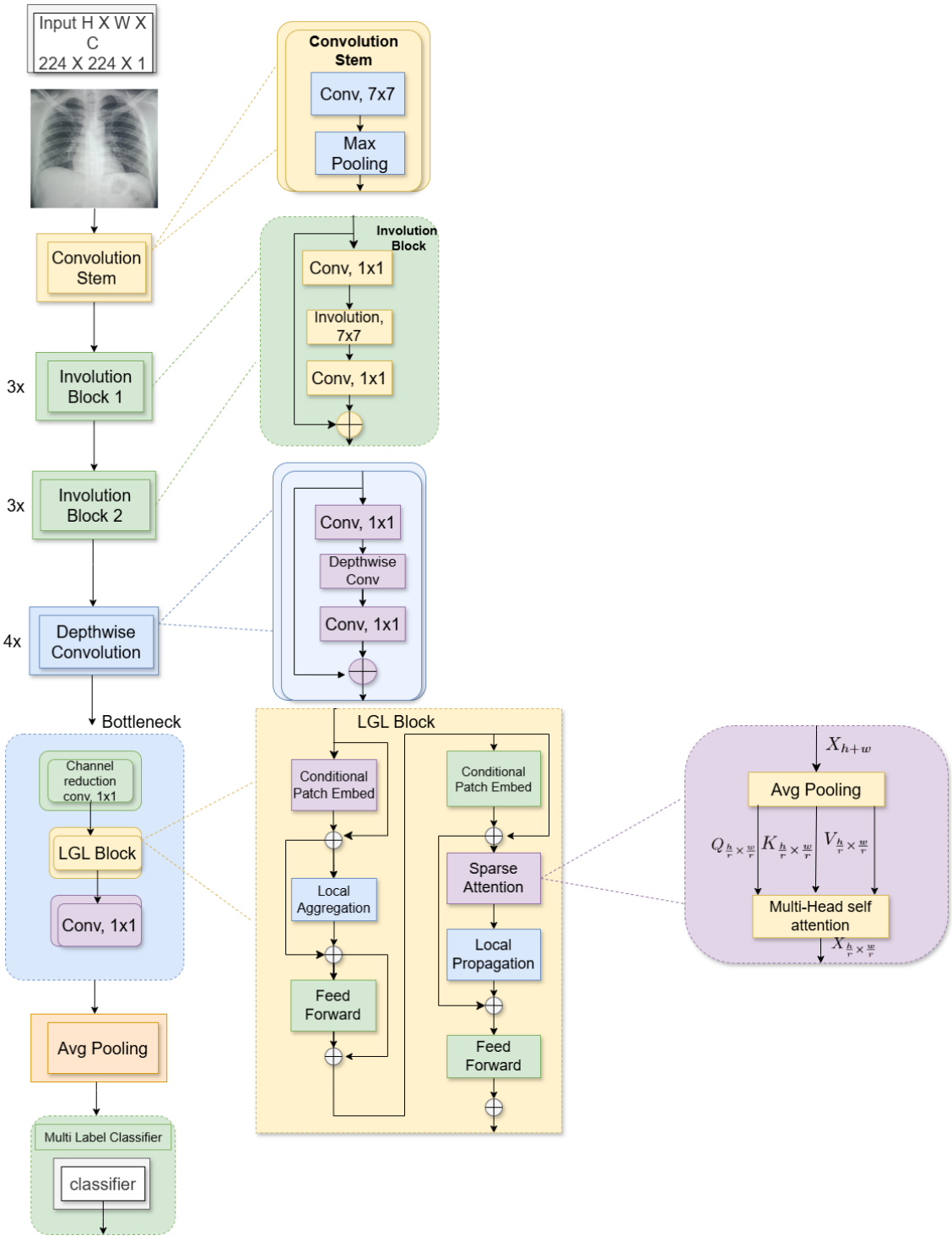


Figure 4.1: Proposed Architecture

### 4.1.1 Convolution Stem

The architecture flow starts with an input of chest x-ray images which were pre processed at the initial layers. The dimension of these input images are $224 \times 224 \times 1$, depicting Height, Width and number of total channels in the x-ray films. These images are now propagated to a convolution stem. Convolutional Neural Networks have shown groundbreaking performance in image classification tasks for their excep-

tional capability of 1) capturing spatial reasoning, 2) feature extraction with variable filters for each channel and 3) providing inductive bias i.e translation equivalence. In this stage we use a convolution of $7 \times 7$. GELU activation function is used in the Convolution layer for introducing non linearity in the weights. Lastly, dimensions of the feature maps are reduced by a max-pooling layer which is the last layer of the convolution stem.

We employed a convolution stem at the beginning for reducing the spatial dimension of the input tensor and extracting meaningful patterns. We demonstrate the mathematical operation of the convolution mechanism in equation 4.1 below:

$$\text{Conv}(x) = Y_{i,j,k} = \sum_{c=1}^{C_i} \sum_{(u,v)\in\Delta_K} A_{k,c,u+\frac{K}{2},v+\frac{K}{2}} X_{i+u,j+v,c} \tag{4.1}$$

where,

$$\begin{aligned} Y &= \text{output feature map} \\ X &= \text{input feature map} \\ A &= \text{Filter} \\ i &\rightarrow \text{row number} \\ j &\rightarrow \text{column number} \\ u, v &\rightarrow \text{row, column offset} \\ c &\rightarrow \text{channel number} \\ C_i &\rightarrow \text{total channels of input} \\ k &\rightarrow \text{filter number} \\ \Delta_K \in \mathbb{Z}^2 &\rightarrow \text{set of offsets in the neighborhood of the kernel} \end{aligned}$$

### 4.1.2 Involution Block 1 and 2

We leverage the strengths of both convolution and Involution in our model. Involution is a novel approach for feature extraction through spatial specific and channel agnostic kernels [91].The spatially specific Involution kernels can effectively extract features by prioritizing pixels with the most underlying information. For example, the pixels surrounding lung area in our x-ray films would carry more details about the disease than any pixel on the borderline of the x-ray. Moreover, Involution enables our model to capture long range spatial dependencies with fewer iterations. Contributing to the efficiency, the channel agnostic nature of Involution helps to overcome the inner-channel redundancy of CNN [91]. These dynamic features make the Involution mechanism suitable to integrate in our proposed model. In this paper, the Involution operation is defined in equation 4.2:

$$\text{Inv}(X) = Y_{i,j,k} = \sum_{(u,v)\in\Delta_K} H_{i,j,u+\frac{K}{2},v+\frac{K}{2},\frac{kG}{C}} X_{i+u,j+v,k} \tag{4.2}$$

where,

$$
\begin{aligned}
Y &= \text{output feature map} \\
X &= \text{input feature map} \\
k &\rightarrow \text{channel number} \\
G &\rightarrow \text{number of groups} \\
C &\rightarrow \text{total channels of input} \\
\Delta_K \in \mathbb{Z}^2 &\rightarrow \text{set of offsets in the neighborhood of the kernel}
\end{aligned}
$$

We have incorporated two blocks of Involution in our proposed architecture. In each block, 3x3 Involution is implemented in a bottleneck structure.With these Involution layers, we ensure that the number of parameters are kept optimal while extracting features; this enables our model to work in a resource-constrained environment.

### 4.1.3 Depthwise Convolution Layer

The output from the Involution blocks are fed into depthwise-convolution. We implemented depthwise-convolution in this layer for bringing efficiency in our computation. The core motivation of depthwise-convolution is defined in equation4.3. Moreover, depthwise convolution can generalize better with a small number of samples. To place the depthwise block into a bottleneck structure, a 1x1 convolution as both channel reduction layer and channel expansion layer was introduced at the starting and ending of this block respectively. This block contributed to our vision of extracting meaningful spatial patterns while keeping the parameter count in check.

$$
\text{DWConv}(X) = Y_{i,j,k} = \sum_{(u,v)\in\Delta_K} G_{k,u+\frac{K}{2},v+\frac{K}{2}} X_{i+u,j+v,k} \tag{4.3}
$$

where,

$$
\begin{aligned}
Y &= \text{output feature map} \\
X &= \text{input feature map} \\
G &\rightarrow \text{kernel} \\
k &\rightarrow \text{channel number} \\
\Delta_K \in \mathbb{Z}^2 &\rightarrow \text{set of offsets in the neighborhood of the kernel}
\end{aligned}
$$

### 4.1.4 Local-Global-Local (LGL) Block

A cost-efficient LGL block as introduced in [84] was implemented to place the Sparse Attention in our model. We take the motivation from [84] to describe different parts of the LGL Block below:

- Conditional Patch Embedding: Conditional patch embedding was used instead of the usual embedding module. This can be implemented using 2D depth-wise convolutions with a residual connection[84].

- Local Aggregation: In this stage, delegate tokens learn spatial information about their local area through depthwise convolution. A batch normalization layer was added at the beginning, followed by a pointwise convolution and then a depthwise convolution layer. At the later parts, another batch normalization layer was added. Local aggregation ended with a pointwise convolution. The operation of the depthwise convolution and pointwise convolution is introduced and mathematically defined in equation 4.3 and equation 4.4 respectively.

$$\text{PWConv}(X) = Y_{i,j,k} = \sum_{c=1}^{C_i} F_{k,c} X_{i,j,c} \tag{4.4}$$

where,

$$\begin{aligned} Y &= \text{output feature map} \\ X &= \text{input feature map} \\ F &= \text{Filter} \\ k &= \text{Filter number} \\ C &\rightarrow \text{channel number} \\ C_i &\rightarrow \text{Total number of channel in input} \end{aligned}$$

- Sparse Attention: Instead of considering the entire input tensor for feeding into MHSA, self-attention as demonstrated in equation 4.5, was applied to a sparse set of delegate tokens. The delegate tokens were computed using average-pooling. These tokens were passed into the self attention layer. 4 heads of self attention were used.

$$\text{Self-Attention}(Q, K, V) = \left(\text{Softmax}\left(\frac{Q^T K}{\sqrt{d_k}}\right)\right)^T \times V \tag{4.5}$$

where,

$$\begin{aligned} Q &= \text{Query} \\ K &= \text{Key} \\ V &= \text{Value} \\ d_x &= \text{Dimension of each head} \end{aligned}$$

- Local Propagation: Delegate tokens share the features extracted in the attention layer with the non-delegate tokens using transpose convolution. The mathematical formulation of the transposed convolution mechanism is detailed in equation 4.6.

$$\text{TransConv}(X) = Y_{i+u,j+v,k} = \sum_{(u,v)\in\Delta_K} A_{u+\frac{K}{2},v+\frac{K}{2},k} X_{i,j,k} \tag{4.6}$$

where,

$$\begin{aligned} Y &= \text{output feature map} \\ X &= \text{input feature map} \\ A &\rightarrow \text{kernel} \\ k &\rightarrow \text{channel} \\ \Delta_K \in \mathbb{Z}^2 &\rightarrow \text{set of offsets in the neighborhood of the kernel} \end{aligned}$$

- Feed Forward Neural Network: We used GELU in feed forward layers. This was crucial for introducing non-linearity to the weights. Equation 4.7 in this paper provides the mathematical formulation of the GELU activation function.

$$\text{GELU}(x) = x\emptyset(x) = \frac{x}{2}(1 + erf(\frac{x}{\sqrt{2}}) \tag{4.7}$$

where,

$$\begin{aligned} X &= \text{input} \\ \emptyset(x) &\rightarrow \text{Gaussian cumulative distribution function} \\ erf &\rightarrow \text{error function} \end{aligned}$$

### 4.1.5 Bottleneck Structure

We strictly employed the fundamental components of our proposed architecture:Depthwise convolution, Involution and LGL block in between a bottleneck structure to reduce the computational cost of our model in Figure 4.1. A Bottleneck acts like a powerful backbone which first contracts the number of channels and is followed by core operation like attention. At the end, the size of the feature map is expanded again. It enables the core operations of our model to only prioritize the most essential patterns by reducing the number of channels, and makes them computationally efficient.

### 4.1.6 Multilabel Classifier

The features extracted from the preceding layers are now propagated into a feed forward network. Finally classification is done into normal, abnormal and pneumonic categories.

We combined the strengths of the aforementioned core operations as described in [91], [84] and [92]. Our hybrid model InvoSparseNet as demonstrated in Figure 4.1

is robust in classification tasks and performs well in a resources constrained environment because 1) it captures spatial reasoning efficiently with depthwise convolution kernels, 2)can prioritize more information rich areas in the chest x-ray and can work with fewer parameters by having Involution do the feature extraction and 3) Global dependency is captured using Sparse Attention, placed at the deeper layers, which only operates on the information rich feature maps from the top layers.

| **Layer Name** | **Output Size** | **Layer Components** |
|---|---|---|
| Conv_Stem | 56x56x64 | 7×7, 64, stride 2 / Conv<br>3×3, stride 2 / Maxpool |
| Inv_Layer_1 | 28x28x256 | 1×1×64 / PWConv<br>7×7, padding=3, group=16 / Invo<br>1×1×256 / PWConv |
| Inv_Layer_2 | 14x14x512 | 1×1×128 / PWConv<br>7×7, padding=3, group=16 / Invo<br>1×1×512 / PWConv |
| DWConv_Layer | 14x14x1024 | 1×1×256 / PWConv<br>3×3, padding=1, stride=2, groups=256 / DWConv<br>1×1×1024 / PWConv |
| Channel Reduction | 14x14x64 | 1×1×64 / PWConv |
| LGL Block | 14x14x64 | 3×3×64, padding=1, groups=64 / CPE<br>1×1×64 / LocalAgg<br>3×3×64, padding=1, groups=64 / LocalAgg<br>1×1×64 / LocalAgg<br>1×1×256 / MLP<br>1×1×64 / MLP<br>1×1, stride=2 / AvgPool<br>256×16×4 (dim, patches, heads) / Q, K, V<br>2×2×64, stride=2 / TransConv |
| Channel Expansion | 14x14x1024 | 1×1×1024 / PWConv |
| Average Pool | 1x1x1024 | 14×14 / AvgPool |
| Linear | 3 | 1024×3 / NN |

Table 4.1: Overview of the Proposed Model Layers

# Chapter 5

# Implementation

## 5.1 Workflow Details

We tested our model on our primary dataset and also on another benchmark CXR dataset from Kaggle. Furthermore, 5 different SOTA architectures were also trained to compare the performances and draw conclusions. Hyperparameters for all of them were kept the same which is listed in table 5.1. Cross Entropy loss was selected as the loss function for our implementation and we used Adam optimizer with L2 regularization. All these models were also trained in Federated Learning settings with our private dataset where we selected 50% of the clients at each round randomly. The entire dataset was distributed unevenly among all the 10 clients. We have assumed an IID-distribution of the dataset among all the participants. Each one of the selected clients trained their local model with their local dataset and the updates were sent back to the server. Local training of each client was performed using the FedProx algorithm shown in equation 5.1. The server then computed a weighted average of the updates from the clients as per their dataset size to ensure fairness while updating the global model. Pytorch library of Python was used to carry out the entire implementation and training process. CUDA was utilized to train the models in an NVIDIA RTX 3060 Ti GPU. Lastly, to add further explainability, GradCAM was used to show how the model identifies each class of the CXR dataset. Algorithm 1 shows a pseudocode of our centralized implementation using K-fold cross validation with 5 folds.

$$\text{Loss} = F(w) + \frac{\mu}{2}\|w - w^t\|^2 \tag{5.1}$$

where,

$$\begin{aligned} w &= \text{local model weights} \\ w^t &= \text{global model weights} \\ F(w) &\rightarrow \text{local model's cross entropy loss} \\ \mu &\rightarrow \text{regularization parameter} \end{aligned}$$

**Algorithm 1** Centralized Implementation

**Dataset Distribution:**
$p \leftarrow$ path to dataset
$d \leftarrow$ dataloader($p$)
**Augmentations:**
CLAHE($d$)
Rotation($d$)
Color Jitter($d$)
**Model Definition:**
$x \leftarrow$ Conv_stem($x$) ▷ x = Input Tensors
$x \leftarrow$ Inv2(Inv1($x$))
$x \leftarrow$ DWConv($x$)
$x \leftarrow$ Sparse_Attn($x$)
$x \leftarrow$ Classifier($x$)
**Training:**
$folds \leftarrow$ K_fold($d, k = 5$) ▷ Distributing dataset in 5 folds
**for** $i \leftarrow 1$ to $folds$ **do**
    $e \leftarrow 5$ ▷ Epochs per fold
    **for** $j \leftarrow 1$ to $e$ **do**
        $scores \leftarrow$ model($fold\_data$)
        $loss()$
        $backward\_propagate()$
    **end for**
    $acc, recall, prec, cm \leftarrow$ evaluate($remaining\_fold\_data$)
**end for**
**Evaluation:**
$Accuracy \leftarrow$ average($acc$)
$Recall \leftarrow$ average($recall$)
$Precesion \leftarrow$ average($prec$)
$Confusion\_Matrix \leftarrow$ sum($cm$)

| **Hyperparameter** | **Values** |
|---|---|
| in_channels | 1 |
| num_classes | 3 |
| learning_rate | 0.0001 |
| batch_size | 16 |
| num_epochs_per_fold | 4 |
| num_folds | 5 |
| Optimizer | Adam |
| Activation | GELU |
| Loss | Cross Entropy |

Table 5.1: Parameters of the Proposed Model

## 5.2 Federated System Details

Centralized implementation of Deep Learning models have shown great success in terms of achieving high accuracy for tasks like classification, segmentation, detection etc. However, in the medical domain, collection of large datasets is highly unlikely due to the privacy concerns of the institutions. As a result, to keep our approach relevant to real-world contexts, we employed the decentralized method known as Federated Learning to train our model. Figure 2.1 shows the workflow of a federated system where the clients can be assumed as the individual medical institutions who train the model locally using their own dataset and send the updated weights to the global server. We further integrated Federated Proximal algorithm which is shown in equation 5.1 to minimize the uneven distribution of data among different clients. Other factors like communication cost, non-iid datasets and further privacy preserving techniques also affects the accuracy but has not been addressed and are left for future work. Algorithm 2 shows a pseudocode of our Federated implementation using 10 clients with 0.5 selection probability for 5 rounds.

**Algorithm 2** Federated Learning

**Dataset Distribution:**
$p \leftarrow$ path to dataset
$d \leftarrow$ dataloader($p$)
$client\_num \leftarrow 10$
$client\_datasets \leftarrow$ distribute_dataset($d$)
**Augmentations:**
CLAHE($client\_datasets$)
Rotation($client\_datasets$)
Color Jitter($client\_datasets$)
**Model Definition:**
$x \leftarrow$ Conv_stem($x$) ▷ x = Input Tensors
$x \leftarrow$ Inv2(Inv1($x$))
$x \leftarrow$ DWConv($x$)
$x \leftarrow$ Sparse_Attn($x$)
$x \leftarrow$ Classifier($x$)
**Federated Training:**
$n \leftarrow 5$ ▷ Number of rounds
**for** $i \leftarrow 1$ to $n$ **do**
    **Global Server Broadcast:**
    $client\_models \leftarrow$ send_global_model($client\_num$)
    $pr \leftarrow 0.5$ ▷ Client selection probability
    $e \leftarrow 5$ ▷ Epochs per client
    $selected\_clients \leftarrow$ sample($client\_models, pr$)
    **for** each client in $selected\_clients$ **do**
        **for** $j \leftarrow 1$ to $e$ **do**
            $client\_model(data)$
            $loss \leftarrow$ CE Loss + Proximal Term ▷ FedProx
            $backward\_propagate()$
        **end for**
        send_update_to_server(local_weights)
    **end for**
    **Server Side Aggregation:**
    $global\_weights \leftarrow$ weighted_average($local\_weights$) ▷ FedAVG
**end for**
**Evaluation:**
$Accuracy, Loss, Num\_Rounds \leftarrow$ generate_graphs($global\_model$)

## 5.3 Comparative Analysis

Our comprehensive experiments are evaluated with various metrics and necessary graphs to provide an in-depth analysis.

### 5.3.1 Private Dataset Experiments

Training on the private dataset was carried out in K-Fold cross validation style and the results shown in table 5.2 provided some great insights into the classification task. Swin, being a fully ViT based architecture and the hybrid LeViT and CVT architectures had relatively poor performance on the small CXR private dataset. It can be explained by an inherent property of ViT architectures that they require a significant amount of training examples to reach the optimal level and show decent performance. Furthermore, among the hybrid architectures only MobileViT V2 achieved the highest performance as it had less reliance on the Transformer section and more dominance in the DepthWise Convolution blocks. However, MobileNet V2 being a fully depthwise convolution based architecture had only 4.5M parameters and produced 86.8% accuracy which is very close to the performance of MobileViT V2. This is again due to the nature of the classification task as lung based diseases are more likely to be localized in short regions which makes the convolution operation gain an edge in this regard. Finally, our architecture produces 85.9% accuracy which is very close to the best performing SOTA models while standing with the lowest parameter count in this entire group. The confusion matrix is generated with randomly selected 421 images from the dataset and is shown in Figure 5.1, also reflects this performance where it is evident that the model was able to handle all the separate classes with great accuracy. To provide further clarity in this aspect, AUC-ROC is computed using the entire dataset and the curves are provided in Figure 5.2, which shows consistent performance across all the 3 classes.This proves that the model is reliable in terms of clearly distinguishing among the different classes.

| Model | Parameters | Precision | Recall | F1 score | Accuracy | Latency | GFLOPs |
|---|---|---|---|---|---|---|---|
| Swin | 29M | 0.6338 | 0.6338 | 0.6343 | 66.58% | 66.7ms | 4.5 |
| LeViT | 7.8M | 0.7148 | 0.8703 | 0.7849 | 77.03% | 19.5ms | 0.6 |
| CVT | 19M | 0.7771 | 0.7363 | 0.7562 | 73.63% | 72.3ms | 4.53 |
| MobileViT_S | 5.6M | 0.8779 | 0.8689 | 0.8734 | 87.09% | 42.2ms | 0.7 |
| MobileNet_V2_S | 4.5M | 0.872 | 0.8682 | 0.8701 | 86.82% | 37.2ms | 0.3 |
| InvoSparseNet | 3.2M | 0.8642 | 0.8594 | 0.8618 | 85.94% | 47.4ms | 0.3 |

Table 5.2: Performance Metrics of The State of The Art Architectures

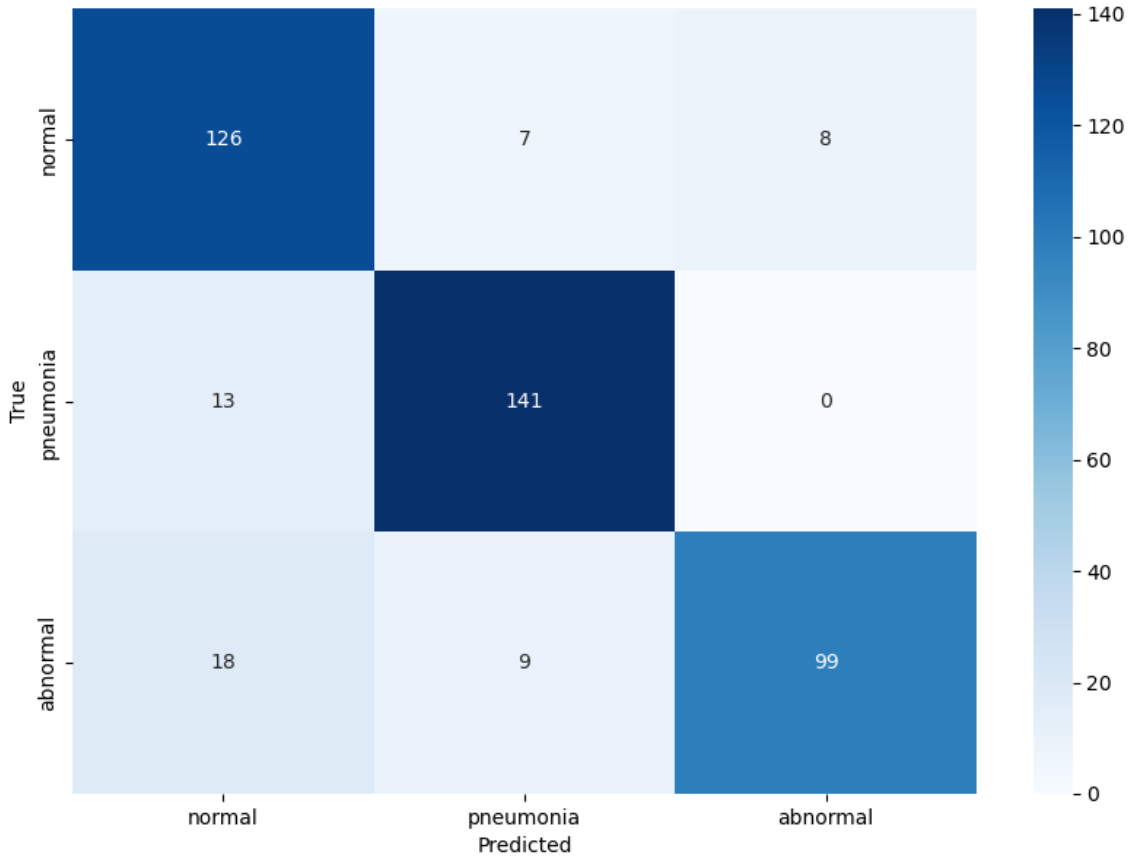


Figure 5.1: Confusion Matrix of InvoSparseNet on Private Dataset

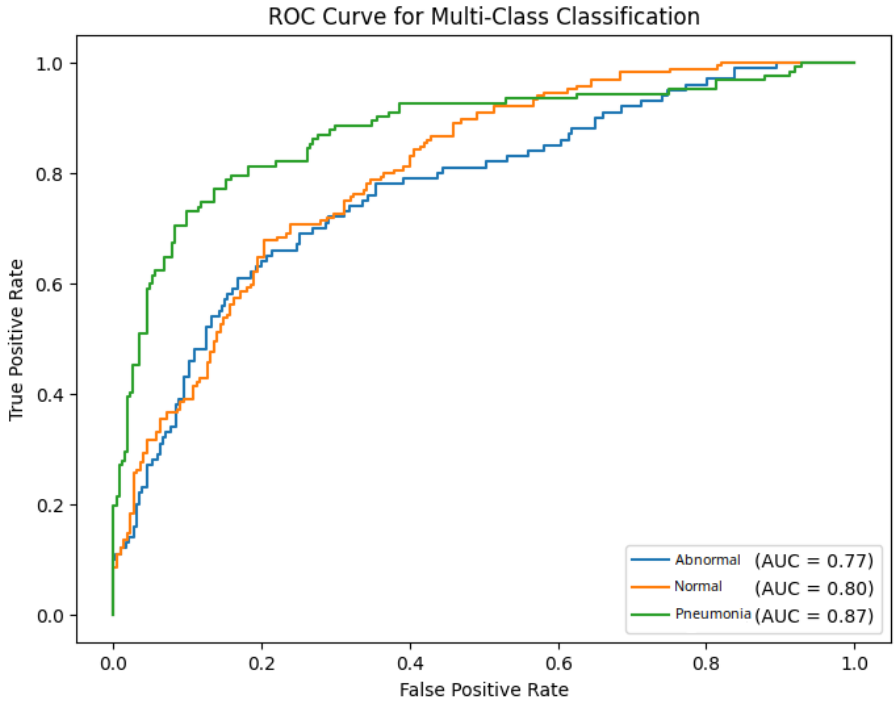


Figure 5.2: AUC-ROC Curve of InvoSparseNet on Private Dataset

### 5.3.2 Benchmark Dataset Experiments

All the models were also tested on the benchmark dataset from Kaggle. This is done primarily to justify the fact that the results produced by models are consistent across different datasets. Also, this further clarifies the credibility of the private dataset as the results shown in table 5.3 are in much coherence with the results from private dataset. One noticeable difference is that the accuracy scores are above 90% and the heavy ViT models have performed significantly better. This is also a predictable and reliable outcome for this scenario as the training samples in the benchmark dataset is much higher and this shows an overall improvement in the testing accuracy.

| Model | Parameters | Accuracy |
|---|---|---|
| Swin | 29M | 79.26% |
| LeViT | 7.8M | 77.03% |
| CVT | 19M | 88.11% |
| MobileViT_S | 5.6M | 96.01% |
| MobileNet_V2_S | 4.5M | 95.68% |
| InvoSparseNet | 3.2M | 95.06% |

Table 5.3: Performance Metrics on Benchmark Dataset

### 5.3.3 Federated Environment Experiments

Testing in the federated setup is a crucial aspect for this classification task due to the data scarcity and privacy concerns of medical institutions. Hence, in a realistic situation, Federated Learning must be employed to train the model to reach the optimal state. However, this approach adds some more factors like network connectivity, communication latency, non-IID datasets etc which can hinder the training performance. In our implementation, we mainly restricted our approach to focus on the performance of training assuming that these aforementioned factors have negligible impact. The results in table 5.4 shows that the accuracy achieved in the Federated Learning environment is very much similar to the centralized one. This is mainly due to the fact that we assumed an IID distribution and other aspects of Federated Learning was not taken into consideration. Our model has once again, shown its competitiveness with the best performing SOTA architectures. This further clarifies that our model is suitable and does now show any hindrance while working in the decentralized environment.Also accuracy and loss were monitored over 10 rounds while training the InvoSparseNet in the federated setup which in shown in Figure 5.3.

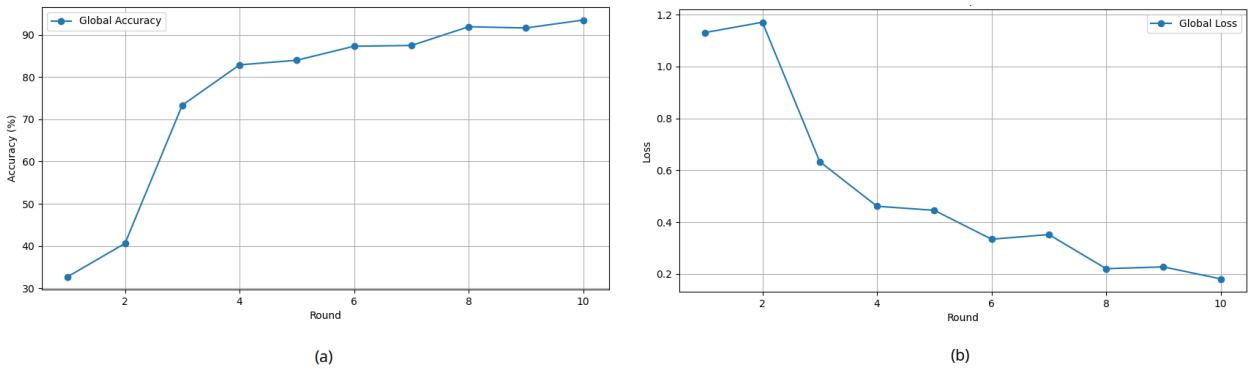


Figure 5.3: (a) Accuracy vs Loss and (b) Loss vs Round Curves

| Model | Parameters | Accuracy |
|---|---|---|
| Swin | 29M | 65.52% |
| LeViT | 7.8M | 73.36% |
| CVT | 19M | 70.02% |
| MobileViT_S | 5.6M | 87.07% |
| MobileNet_V2_S | 4.5M | 87.05% |
| InvoSparseNet | 3.2M | 86.43% |

Table 5.4: Performance Metrics on Federated Environment

### 5.3.4 GradCAM

Gradient Weighted Class Activation Mapping is used to visualize the specific parts of the input image where the model focuses when making a prediction. The generated heat-maps highlight the important regions which made the most contribution for a model's decision. GradCAM results of the three different classes are shown in Figure 5.4. Regions highlighted in case of the pneumonia class are mostly around the lung border shown with red to signify high activation. For the "abnormal" class, the regions which depict the abnormality have also been focused in red. In the case of "normal" class the activation regions are very low in strength which is shown by light green colors in the heatmap. With the help of these results, it is clearly evident that the model has learned relevant pathological features and not relying on non-diagnostic cues to correctly classify the images.

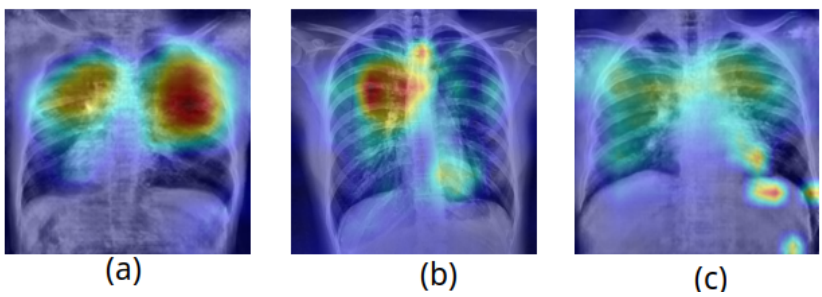


Figure 5.4: GradCAM Heatmaps of (a) Abnormal, (b) Pneumonia and (c) Normal Classes

### 5.3.5 Ablation Analysis

In this section we present the ablation studies which further helps to explain the key contributions of each of the components of our architecture.
To find an optimal arrangement for stacking the layers of our architecture, we analyzed various combinations of layers by shifting their relative positions and varying the sizes. As depicted by the 3rd and 4th rows of table 5.5 using only Depthwise Convolution or Involution alone does not provide the maximum achievable accuracy. Using all Involutions layers provide heavy focus on spatial features but they need many samples to learn which is not sufficient in this scenario, hence the accuracy is not maximized. On the other hand, using all depthwise convolution layers would help to learn sufficiently with few images but spatial focus will not be powerful.Hence, a combination of both of these provide better performance. Moreover, the most optimal performance with a balanced tradeoff between accuracy and parameter count is achieved when the Depthwise Convolution layers are stacked after the initial Involution layers. This can be further explained by the fact that Involution helps in early stages to capture low-level details like the edges or textures with great precision due to its varying kernel sizes and focus on important regions. As this information is passed to the deeper layers, they combine this information in a cost efficient way with the help of depthwise convolution and learn the pattern quickly even with few images. However, if we use depthwise convolution in early stages followed by Involution layers, this would lose efficiency because spatial size reduces as we go into deeper layers .With such reduced sizes, Involution kernels will not be able to extract much spatial information which it can in the early layers.Furthermore, using only 6 layers (4 Involution and 2 depthwise convolution) reduces parameter count significantly but the accuracy drops by a large amount as the model is unable to learn features effectively due to its short capacity. On the other hand, using more than 10 layers would increase the parameter count excessively that will not be suitable for a general purpose PC based system. Overall, keeping the Involution layers at early stages and depthwise convolution with a total of 10 layers ensures that the parameter count is kept low but not to an extent that might significantly harm the accuracy.

| Layer Structures | Parameters | Accuracy |
|---|---|---|
| 2INV - 2INV - 2DWConv | 2.4M | 76.21% |
| 3DWConv - 4INV - 3INV | 3.1M | 82.52% |
| 10DWConv | 3.6M | 81.57% |
| 10INV | 2.9M | 79.24% |
| 3INV - 3INV - 4DWConv | 3.1M | 84.54% |

Table 5.5: Impact of Layers Rearrangement

### 5.3.6 Attention Mechanism

Next, we analyzed the impact of different attention mechanisms in the final layer of our architecture. As shown in the table 5.6, Sparse Attention outperforms the general multi headed self attention by 1.7% with 0.6M less parameters. Sparse attention is applied over a selected section of tokens making it computationally

efficient. However, the improvement in accuracy is more related to the context of the classification task we attempted. Sparse attention focuses more on the local surrounding regions and the medical CXR disease abnormalities are also localized in small regions. This helps to learn those irregular patterns more effectively. This is not found in case of regular multi-headed self attention mechanisms as they provide attention to all tokens which is unnecessary in this context. Hence, Sparse attention extends the capabilities of the model with a slight increase in parameter count and a substantial benefit in terms of learning patterns and increasing the overall accuracy.

| Attention Mechanism | Parameters | Accuracy |
|---|---|---|
| MHSA | 4M | 84.21% |
| Sparse Attention | 3.2M | 85.94% |

Table 5.6: Impact of Using Different Attention Mechanisms

### 5.3.7 Federated Algorithm

Afterwards, we also tested various algorithms for the Federated Learning environment. FedAvg shows a decent performance as it computed a weighted average during server side aggregation as shown in the table 5.7. It is mainly because the dataset set is unevenly distributed and FedAvg can deal with it effectively as updates from small dataset clients are less prioritized. However FedProx performed best in this scenario as it added another proximal term in the loss function during client side training of the local models. The proximal term helped to keep the local models close to the global one and it ensured a better accuracy. FedMax was the least effective among all 3. It adjusts the loss function by adding the loss for the class that was given the highest probability. However, its complex process makes it difficult to generalize in a small dataset environment which led to its degraded performance. Overall FedProx was able to show optimal performance with its simplified approach and proper client side training.

| Fed Algorithm | Accuracy |
|---|---|
| FedMax | 81% |
| FedProx | 86% |
| FedAvg | 84% |

Table 5.7: Impact of Using Different Federated Algorithms

### 5.3.8 Hyperparameters

Lastly, we tuned the hyperparameters to find the optimal choices which leads to maximum accuracy. Table 5.8 shows the different combinations of hyperparameters that were tested. The lowest learning rate of 0.0001 was the most effective as it ensures that the model learns and updates its parameter values gradually. This is effective because a faster learning rate can make the learning process very quick but the parameter values may get updated aggressively which may surpass the minimum point and the model will not be able to reach its optimal state. Using a smaller

batch size of 8 reduces the accuracy because the gradients are not computed with a large number of samples which can lead to inaccurate values. However using a batch size of 16 increases accuracy as it leads to better calculation of gradients. Furthermore higher batch sizes like 32 and 64 were avoided because it would require a lot of memory for computation. Among the 3 optimizers, RMSProp performs better than SGD as it uses a scaling learning rate which changes for different parameters as per their gradient values whereas SGD uses only a fixed learning rate. However, Adam performs best as it is an updated version of RMSProp which introduces momentum along with scaling learning rates.Momentum helps to accumulate information of previous updates that makes the changes in parameter values more gradual and smooth. GELU activation was more effective than SILU as it applies a probabilistic approach and is proven to be most effective in hybrid architectures whereas SILU would be more effective in simpler cases. L2 regularization helps to avoid overfitting by adding a term in the loss function which depends on the value of the weights. Larger weights can lead to overfitting and this issue is penalized with L2 regularization by increasing the loss so that the model is discouraged from reaching large weight values. All these configurations ensure that the model stays within the hardware limitations with negligible compromise in accuracy.

| Learning_Rate | Batch_Size | Optimizer | Activation Function | L2-Regularization | Accuracy |
|---|---|---|---|---|---|
| 0.0001 | 16 | Adam | GELU | FALSE | 85.94% |
| 0.0003 | 16 | Adam | GELU | FALSE | 85.84% |
| 0.001 | 16 | Adam | GELU | FALSE | 79.12% |
| 0.0001 | 8 | Adam | GELU | FALSE | 84.31% |
| 0.0001 | 16 | SGD | GELU | FALSE | 69.72% |
| 0.0001 | 16 | RMSProp | GELU | FALSE | 85.41% |
| 0.0001 | 16 | Adam | SILU | FALSE | 84.36% |
| 0.0001 | 16 | Adam | GELU | TRUE | 85.47% |

Table 5.8: Hyper-Parameter Tuning Results

# Chapter 6

# Conclusion

## 6.1 Conclusion

Our research presents an innovative approach to lung disease image classification by introducing an Involution, Depthwise convolution, and Sparse Attention-Based architecture, InvoSparseNet within a Federated Learning framework. Our hybrid Deep Learning model addresses the critical challenge of misdiagnosis, which poses significant risks to patient safety, by acting as a reliable assistant to medical professionals, improving diagnostic accuracy.Unlike traditional models that often require high-end computational resources, our lightweight model is designed to operate efficiently on standard PCs, making InvoSparseNet suitable for widespread use in hospital settings. The use of Involutions reduces parameter requirements by employing position-specific kernels shared across channels, while Sparse Attention enhances computational efficiency by focusing on key data points. These design choices ensure resource efficiency without compromising accuracy. Moreover, by integrating Federated Learning, the model ensures that sensitive data remains secure within local devices or institutions, enabling collaborative model training while maintaining strict privacy safeguards. With its high accuracy, computational efficiency, and strong privacy protections, our model provides a practical and reliable solution for improving lung disease diagnosis in diverse medical environments. We evaluated the model using a private dataset collected from a medical hospital, alongside benchmark datasets.The data analysis of our study demonstrates the robustness of InvoSparseNet as our model achieved 95.06% and 86% accuracy on benchmark and private dataset respectively with a lower parameter count of 3.2 million.

## 6.2 Future Work

We have explored the application of InvoSparseNet in case of image classification tasks. However, our model can be implemented in other computer vision related tasks as well. We have combined the future work of InvoSparseNet below:

- Our model can work with diverse datasets with multiple classes (beyond the current limit of three), to further enhance the model's generalization and robustness.

- Given the limited time available during the development phase, additional time will be required for fine-tuning and comprehensive testing. Therefore, we plan to implement a pruning strategy to further reduce the model's complexity and improve efficiency.
- While Federated Learning was employed to ensure data privacy, future work will need to address challenges such as communication costs, privacy concerns, and the handling of non-IID datasets.
- In addition to image classification, future work will focus on extending our model to perform image segmentation, disease detection and localization.
- Further research can be conducted to observe the post-Covid effects on the respiratory system of specific gender and age group by utilizing our primary dataset.

# Bibliography


[1] Ascend Imaging Center, *Pneumonia on chest x-ray: Diagnosis and symptoms*, Accessed: 2025-02-06, 2025. [Online]. Available: https://ascendimagingcenter.com/blogs/pneumonia-on-chest-x-ray/#:~:text=Why%20Are%20Chest%20X%2DRays,pus%20accumulation%20in%20the%20lungs..

[2] W. H. Organization, *Lung cancer who.int*, 2023. [Online]. Available: https://shorturl.at/3JG8p.

[3] N. C. Institute, *Nci dictionary of cancer terms cancer.gov*, 2024. [Online]. Available: https://www.cancer.gov/publications/dictionaries/cancer-terms/def/alveoli.

[4] M. Clinic, *Pneumonia - symptoms and causes mayoclinic.org*, 2020. [Online]. Available: https://shorturl.at/5fHG1.

[5] J. Hopkins, *Pneumonia hopkinsmedicine.org*. [Online]. Available: https://www.hopkinsmedicine.org/health/conditions-and-diseases/pneumonia.

[6] N. H. S. UK, *Lung cancer - diagnosis nhs.uk*, 2022. [Online]. Available: https://shorturl.at/6cSL4.

[7] ICDRR, *Pneumonia and other respiratory diseases icddrb.org*. [Online]. Available: https://www.icddrb.org/news-and-events/press-corner/media-resources/pneumonia-and-other-respiratory-diseases.

[8] O. Ali, W. Abdelbaki, A. Shrestha, E. Elbasi, M. A. A. Alryalat, and Y. K. Dwivedi, "A systematic literature review of artificial intelligence in the healthcare sector: Benefits, challenges, methodologies, and functionalities," *Journal of Innovation Knowledge*, vol. 8, no. 1, p. 100 333, 2023, ISSN: 2444-569X. DOI: https://doi.org/10.1016/j.jik.2023.100333. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2444569X2300029X.

[9] M. M. Taye, "Understanding of machine learning with deep learning: Architectures, workflow, applications and future directions," *Computers*, vol. 12, no. 5, 2023, ISSN: 2073-431X. DOI: 10.3390/computers12050091. [Online]. Available: https://www.mdpi.com/2073-431X/12/5/91.

[10] G. M. Learning, *Ai vs. ml : How do they differ? — google cloud cloud.google.com*. [Online]. Available: https://cloud.google.com/learn/artificial-intelligence-vs-machine-learning.

[11] M. Rana and M. Bhushan, "Machine learning and deep learning approach for medical image analysis: Diagnosis to detection," en, *Multimedia Tools and Applications*, vol. 82, no. 17, pp. 26 731–26 769, 2023, ISSN: 1380-7501, 1573-7721. DOI: 10.1007/s11042-022-14305-w. [Online]. Available: https://link.springer.com/10.1007/s11042-022-14305-w.

[12] A. W. Deep, *What is deep learning? - deep learning explained - aws aws.amazon.com.* [Online]. Available: https://shorturl.at/nSMNC.

[13] GeeksforGeeks, *Difference between machine learning and artificial intelligence - geeksforgeeks — geeksforgeeks.org*, 2024. [Online]. Available: https://www.geeksforgeeks.org/difference-between-machine-learning-and-artificial-intelligence/.

[14] GeeksforGeeks, *Introduction to deep learning - geeksforgeeks geeksforgeeks.org*, 2024. [Online]. Available: https://www.geeksforgeeks.org/introduction-deep-learning/.

[15] Simplilearn, *How to build powerful keras image classification models — simplilearn simplilearn.com*, 2024. [Online]. Available: https://www.geeksforgeeks.org/introduction-deep-learning/.

[16] J.-G. Lee, S. Jun, Y.-W. Cho, *et al.*, "Deep learning in medical imaging: General overview," en, *Korean Journal of Radiology*, vol. 18, no. 4, p. 570, 2017, ISSN: 1229-6929, 2005-8330. DOI: 10.3348/kjr.2017.18.4.570. [Online]. Available: https://www.kjronline.org/DOIx.php?id=10.3348/kjr.2017.18.4.570.

[17] D. Abdelhafiz, C. Yang, R. Ammar, and S. Nabavi, "Deep convolutional neural networks for mammography: Advances, challenges and applications," en, *BMC Bioinformatics*, vol. 20, no. S11, p. 281, 2019, ISSN: 1471-2105. DOI: 10.1186/s12859-019-2823-4. [Online]. Available: https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-019-2823-4.

[18] S. Indolia, A. K. Goswami, S. Mishra, and P. Asopa, "Conceptual understanding of convolutional neural network- a deep learning approach," *Procedia Computer Science*, vol. 132, pp. 679–688, 2018, International Conference on Computational Intelligence and Data Science, ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2018.05.069. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050918308019.

[19] S. Bhojanapalli, A. Chakrabarti, D. Glasner, D. Li, T. Unterthiner, and A. Veit, "Understanding robustness of transformers for image classification," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 10 211–10 221. DOI: 10.1109/ICCV48922.2021.01007.

[20] C.-F. ( Chen, Q. Fan, and R. Panda, "Crossvit: Cross-attention multi-scale vision transformer for image classification," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2021, pp. 357–366.

[21] K. Batko and A. Ślezak, "The use of big data analytics in healthcare," en, *Journal of Big Data*, vol. 9, no. 1, p. 3, 2022, ISSN: 2196-1115. DOI: 10.1186/s40537-021-00553-4. [Online]. Available: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00553-4.

[22] T. Sanida and M. Dasygenis, "A novel lightweight cnn for chest x-ray-based lung disease identification on heterogeneous embedded system," en, vol. 54, pp. 4756–4780, Mar. 2024, ISSN: 0924-669X, 1573-7497. DOI: 10.1007/s10489-024-05420-2. [Online]. Available: https://link.springer.com/10.1007/s10489-024-05420-2.

[23] A. M. Hussein, A. G. Sharifai, O. M. Alia, *et al.*, "Auto-detection of the coronavirus disease by using deep convolutional neural networks and x-ray photographs," en, *Scientific Reports*, vol. 14, no. 1, p. 534, Jan. 2024, ISSN: 2045-2322. DOI: 10.1038/s41598-023-47038-3. [Online]. Available: https://www.nature.com/articles/s41598-023-47038-3.

[24] A. M. Q. Farhan and S. Yang, "Automatic lung disease classification from the chest x-ray images using hybrid deep learning algorithm," en, *Multimedia Tools and Applications*, vol. 82, no. 25, pp. 38 561–38 587, Oct. 2023, ISSN: 1380-7501, 1573-7721. DOI: 10.1007/s11042-023-15047-z. [Online]. Available: https://link.springer.com/10.1007/s11042-023-15047-z.

[25] G. Mohan, M. M. Subashini, S. Balan, and S. Singh, "A multiclass deep learning algorithm for healthy lung, covid-19 and pneumonia disease detection from chest x-ray images," en, vol. 4, p. 20, Mar. 2024, ISSN: 2731-0809. DOI: 10.1007/s44163-024-00110-x. [Online]. Available: https://link.springer.com/10.1007/s44163-024-00110-x.

[26] M. Abdullah, F. B. Abrha, B. Kedir, and T. Tamirat Tagesse, "A hybrid deep learning cnn model for covid-19 detection from chest x-rays," en, *Heliyon*, vol. 10, no. 5, e26938, Mar. 2024, ISSN: 24058440. DOI: 10.1016/j.heliyon.2024.e26938. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2405844024029694.

[27] S. Sharma and K. Guleria, "A deep learning based model for the detection of pneumonia from chest x-ray images using vgg-16 and neural networks," *Procedia Computer Science*, vol. 218, pp. 357–366, 2023, International Conference on Machine Learning and Data Engineering, ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2023.01.018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050923000182.

[28] A. Hussain, S. U. Amin, H. Lee, A. Khan, N. F. Khan, and S. Seo, "An automated chest x-ray image analysis for covid-19 and pneumonia diagnosis using deep ensemble strategy," *IEEE Access*, vol. 11, pp. 97 207–97 220, 2023. DOI: 10.1109/ACCESS.2023.3312533.

[29] G. M. M. Alshmrani, Q. Ni, R. Jiang, H. Pervaiz, and N. M. Elshennawy, "A deep learning architecture for multi-class lung diseases classification using chest x-ray (cxr) images," *Alexandria Engineering Journal*, vol. 64, pp. 923–935, 2023, ISSN: 1110-0168. DOI: https://doi.org/10.1016/j.aej.2022.10.053. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1110016822007104.

[30] D. M. Ibrahim, N. M. Elshennawy, and A. M. Sarhan, "Deep-chest: Multi-classification deep learning model for diagnosing covid-19, pneumonia, and lung cancer chest diseases," *Computers in Biology and Medicine*, vol. 132, p. 104 348, 2021, ISSN: 0010-4825. DOI: https://doi.org/10.1016/j.compbiomed.2021.104348. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010482521001426.

[31] Y. Liu, W. Xing, M. Zhao, and M. Lin, "A new classification method for diagnosing covid-19 pneumonia based on joint cnn features of chest x-ray images and parallel pyramid mlp-mixer module," en, *Neural Computing and Applications*, vol. 35, no. 23, pp. 17 187–17 199, Aug. 2023, ISSN: 0941-0643, 1433-3058. DOI: 10.1007/s00521-023-08604-y. [Online]. Available: https://link.springer.com/10.1007/s00521-023-08604-y.

[32] M. K. Islam, M. M. Rahman, M. S. Ali, S. Mahim, and M. S. Miah, "Enhancing lung abnormalities detection and classification using a deep convolutional neural network and gru with explainable ai: A promising approach for accurate diagnosis," *Machine Learning with Applications*, vol. 14, p. 100 492, 2023, ISSN: 2666-8270. DOI: https://doi.org/10.1016/j.mlwa.2023.100492. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666827023000452.

[33] V. Ravi, V. Acharya, and M. Alazab, "A multichannel efficientnet deep learning-based stacking ensemble approach for lung disease detection using chest x-ray images," en, *Cluster Computing*, vol. 26, no. 2, pp. 1181–1203, Apr. 2023, ISSN: 1386-7857, 1573-7543. DOI: 10.1007/s10586-022-03664-6. [Online]. Available: https://link.springer.com/10.1007/s10586-022-03664-6.

[34] N. A. Hroub, A. N. Alsannaa, M. Alowaifeer, M. Alfarraj, and E. Okafor, "Explainable deep learning diagnostic system for prediction of lung disease from medical images," *Computers in Biology and Medicine*, vol. 170, p. 108 012, 2024, ISSN: 0010-4825. DOI: https://doi.org/10.1016/j.compbiomed.2024.108012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010482524000969.

[35] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, *et al.*, Eds., vol. 30, Curran Associates, Inc., 2017.

[36] S. Singh, M. Kumar, A. Kumar, B. K. Verma, K. Abhishek, and S. Selvarajan, "Efficient pneumonia detection using vision transformers on chest x-rays," en, *Scientific Reports*, vol. 14, no. 1, p. 2487, Jan. 2024, ISSN: 2045-2322. DOI: 10.1038/s41598-024-52703-2. [Online]. Available: https://www.nature.com/articles/s41598-024-52703-2.

[37] Z. Pan, B. Zhuang, H. He, J. Liu, and J. Cai, "Less is more: Pay less attention in vision transformers," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 2, pp. 2035–2043, Jun. 2022. DOI: 10.1609/aaai.v36i2.20099. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/20099.

[38] Z. Pan, J. Cai, and B. Zhuang, "Fast vision transformers with hilo attention," in *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., vol. 35, Curran Associates, Inc., 2022, pp. 14 541–14 554.

[39] B. Slika, F. Dornaika, H. Merdji, and K. Hammoudi, "Lung pneumonia severity scoring in chest x-ray images using transformers," *Medical & Biological Engineering & Computing*, Apr. 2024. DOI: 10.1007/s11517-024-03066-3. [Online]. Available: https://doi.org/10.1007/s11517-024-03066-3.

[40] Q. Fan, H. Huang, X. Zhou, and R. He, "Lightweight vision transformer with bidirectional interaction," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. [Online]. Available: https://openreview.net/forum?id=492Hfmgejy.

[41] P. K. A. Vasu, J. Gabriel, J. Zhu, O. Tuzel, and A. Ranjan, "Fastvit: A fast hybrid vision transformer using structural reparameterization," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2023, pp. 5785–5795.

[42] R. Ghali and M. A. Akhloufi, "Vision transformers for lung segmentation on cxr images," en, *SN Computer Science*, vol. 4, no. 4, p. 414, May 2023, ISSN: 2661-8907. DOI: 10.1007/s42979-023-01848-4. [Online]. Available: https://link.springer.com/10.1007/s42979-023-01848-4.

[43] S. I. Nafisah, G. Muhammad, M. S. Hossain, and S. A. AlQahtani, "A comparative evaluation between convolutional neural networks and vision transformers for covid-19 detection," *Mathematics*, vol. 11, no. 6, 2023, ISSN: 2227-7390. DOI: 10.3390/math11061489. [Online]. Available: https://www.mdpi.com/2227-7390/11/6/1489.

[44] E. Selvano, A. Paulindino, G. Elwirehardja, and B. Pardamean, "Evaluating self-supervised pre-trained vision transformer on imbalanced data for lung disease classification," *ICIC Express Letters*, vol. 15, pp. 2302–004, Jan. 2024. DOI: 10.24507/icicelb.15.01.XXX.

[45] A. S. Fazal, S. Khan, and A. Khan, "Covit: Convolutions and vit based deep learning model for covid19 and viral pneumonia classification using x-ray datasets," in *Proceedings of the 2023 7th International Conference on Computational Biology and Bioinformatics*, ser. ICCBB '23, ¡conf-loc¿, ¡city¿Kuala Lumpur¡/city¿, ¡country¿Malaysia¡/country¿, ¡/conf-loc¿: Association for Computing Machinery, 2024, pp. 53–60, ISBN: 9798400716331. DOI: 10.1145/3638569.3638577. [Online]. Available: https://doi.org/10.1145/3638569.3638577.

[46] X. Wu, Y. Feng, H. Xu, *et al.*, "Ctranscnn: Combining transformer and cnn in multilabel medical image classification," *Knowledge-Based Systems*, vol. 281, p. 111 030, 2023, ISSN: 0950-7051. DOI: https://doi.org/10.1016/j.knosys.2023.111030. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705123007803.

[47] O. N. Manzari, H. Ahmadabadi, H. Kashiani, S. B. Shokouhi, and A. Ayatollahi, "Medvit: A robust vision transformer for generalized medical image classification," *Computers in Biology and Medicine*, vol. 157, p. 106 791, 2023, ISSN: 0010-4825. DOI: https://doi.org/10.1016/j.compbiomed.2023.106791. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010482523002561.

[48] C. M. Thwal, M. N. Nguyen, Y. L. Tun, S. T. Kim, M. T. Thai, and C. S. Hong, "Ondev-lct: On-device lightweight convolutional transformers towards federated learning," *Neural Networks*, vol. 170, pp. 635–649, 2024, ISSN: 0893-6080. DOI: https://doi.org/10.1016/j.neunet.2023.11.044. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608023006676.

[49] K. Ren, G. Hong, X. Chen, and Z. Wang, "A covid-19 medical image classification algorithm based on transformer," en, *Scientific Reports*, vol. 13, no. 1, p. 5359, Apr. 2023, ISSN: 2045-2322. DOI: 10.1038/s41598-023-32462-2. [Online]. Available: https://www.nature.com/articles/s41598-023-32462-2.

[50] T. Wang, Z. Nie, R. Wang, *et al.*, "Pneunet: Deep learning for covid-19 pneumonia diagnosis on chest x-ray image analysis using vision transformer," en, *Medical Biological Engineering Computing*, vol. 61, no. 6, pp. 1395–1408, Jun. 2023, ISSN: 0140-0118, 1741-0444. DOI: 10.1007/s11517-022-02746-2. [Online]. Available: https://link.springer.com/10.1007/s11517-022-02746-2.

[51] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication Efficient Learning of Deep Networks from Decentralized Data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, A. Singh and J. Zhu, Eds., ser. Proceedings of Machine Learning Research, vol. 54, PMLR, 20–22 Apr 2017, pp. 1273–1282.

[52] J. Konečný, H. B. McMahan, F. X. Yu, A. T. Suresh, D. Bacon, and P. Richtárik, *Federated learning: Strategies for improving communication efficiency*, 2018. [Online]. Available: https://openreview.net/forum?id=B1EPYJ-C-.

[53] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '15, Denver, Colorado, USA: Association for Computing Machinery, 2015, pp. 1310–1321, ISBN: 9781450338325. DOI: 10.1145/2810103.2813687. [Online]. Available: https://doi.org/10.1145/2810103.2813687.

[54] B. Casella, R. Esposito, A. Sciarappa, C. Cavazzoni, and M. Aldinucci, "Experimenting with normalization layers in federated learning on non-iid scenarios," *IEEE Access*, vol. PP, pp. 1–1, Jan. 2024. DOI: 10.1109/ACCESS.2024.3383783.

[55] Y. Xu, Y. Liao, L. Wang, H. Xu, Z. Jiang, and W. Zhang, "Overcoming noisy labels and non-iid data in edge federated learning," *IEEE Transactions on Mobile Computing*, no. 01, pp. 1–16, May 5555, ISSN: 1558-0660. DOI: 10.1109/TMC.2024.3398801.

[56] Y. Wang, W. Wang, X. Wang, H. Zhang, X. Wu, and M. Yang, "Fedtweet: Two-fold knowledge distillation for non-iid federated learning," *Computers and Electrical Engineering*, vol. 114, p. 109 067, 2024, ISSN: 0045-7906. DOI: https://doi.org/10.1016/j.compeleceng.2023.109067. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045790623004913.

[57] H. Chen, A. Frikha, D. Krompass, J. Gu, and V. Tresp, "Fraug: Tackling federated learning with non-iid features via representation augmentation," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, Los Alamitos, CA, USA: IEEE Computer Society, Oct. 2023, pp. 4826–4836. DOI: 10.1109/ICCV51070.2023.00447. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/ICCV51070.2023.00447.

[58] J. Ouyang and Y. Liu, "Learning efficiency maximization for wireless federated learning with heterogeneous data and clients," *IEEE Transactions on Cognitive Communications and Networking*, pp. 1–1, 2024. DOI: 10.1109/TCCN.2024.3394889.

[59] Y. Shi, P. Fan, Z. Zhu, C. Peng, F. Wang, and K. B. Letaief, "Sam: An efficient approach with selective aggregation of models in federated learning," *IEEE Internet of Things Journal*, vol. 11, no. 11, pp. 20 769–20 783, 2024. DOI: 10.1109/JIOT.2024.3373822.

[60] I. Feki, S. Ammar, Y. Kessentini, and K. Muhammad, "Federated learning for covid-19 screening from chest x-ray images," *Applied Soft Computing*, vol. 106, p. 107 330, 2021, ISSN: 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2021.107330. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494621002532.

[61] K. A. Bonawitz, H. Eichner, W. Grieskamp, *et al.*, "Towards federated learning at scale: System design," in *SysML 2019*, To appear, 2019. [Online]. Available: https://arxiv.org/abs/1902.01046.

[62] Y. Hao, C. Zhang, and X. Li, "Dbm-vit: A multiscale features fusion algorithm for health status detection in cxr / ct lungs images," *Biomedical Signal Processing and Control*, vol. 87, p. 105 365, 2024, ISSN: 1746-8094. DOI: https://doi.org/10.1016/j.bspc.2023.105365. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S174680942300798X.

[63] O. Uparkar, J. Bharti, R. Pateriya, R. K. Gupta, and A. Sharma, "Vision transformer outperforms deep convolutional neural network-based model in classifying x-ray images," *Procedia Computer Science*, vol. 218, pp. 2338–2349, 2023, International Conference on Machine Learning and Data Engineering, ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2023.01.209. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050923002090.

[64] A. Mezina and R. Burget, "Detection of post-covid-19-related pulmonary diseases in x-ray images using vision transformer-based neural network," *Biomedical Signal Processing and Control*, vol. 87, p. 105 380, 2024, ISSN: 1746-8094. DOI: https://doi.org/10.1016/j.bspc.2023.105380. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1746809423008133.

[65] R. Gehani and A. Victor, "X-ray based lung disease classification using fine-tuned vgg16 model," in *2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT)*, 2024, pp. 533–538. DOI: 10.1109/IDCIoT59759.2024.10467646.

[66] S. Goyal and R. Singh, "Detection and classification of lung diseases for pneumonia and covid-19 using machine and deep learning techniques," en, *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 4, pp. 3239–3259, Apr. 2023, ISSN: 1868-5137, 1868-5145. DOI: 10.1007/s12652-021-03464-7. [Online]. Available: https://link.springer.com/10.1007/s12652-021-03464-7.

[67] J. Guo, K. Han, H. Wu, *et al.*, "Cmt: Convolutional neural networks meet vision transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2022, pp. 12 175–12 185.

[68] J. Xu, B. S. Glicksberg, C. Su, P. Walker, J. Bian, and F. Wang, "Federated learning for healthcare informatics," en, *Journal of Healthcare Informatics Research*, vol. 5, no. 1, pp. 1–19, Mar. 2021, ISSN: 2509-4971, 2509-498X. DOI: 10.1007/s41666-020-00082-4. [Online]. Available: http://link.springer.com/10.1007/s41666-020-00082-4.

[69] H. Guan, P.-T. Yap, A. Bozoki, and M. Liu, "Federated learning for medical image analysis: A survey," *Pattern Recognition*, vol. 151, p. 110 424, 2024, ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2024.110424. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320324001754.

[70] C. C. Ukwuoma, Z. Qin, M. Belal Bin Heyat, *et al.*, "A hybrid explainable ensemble transformer encoder for pneumonia identification from chest x-ray images," *Journal of Advanced Research*, vol. 48, pp. 191–211, 2023, ISSN: 2090-1232. DOI: https://doi.org/10.1016/j.jare.2022.08.021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2090123222002028.

[71] H. Yang, L. Wang, Y. Xu, and X. Liu, "Covidvit: A novel neural network with self-attention mechanism to detect covid-19 through x-ray images," *International Journal of Machine Learning and Cybernetics*, vol. 14, no. 3, pp. 973–987, Mar. 2023, ISSN: 1868-808X. DOI: 10.1007/s13042-022-01676-7. [Online]. Available: https://doi.org/10.1007/s13042-022-01676-7.

[72] S. d'Ascoli, H. Touvron, M. L. Leavitt, A. S. Morcos, G. Biroli, and L. Sagun, "Convit: Improving vision transformers with soft convolutional inductive biases," in *International conference on machine learning*, PMLR, 2021, pp. 2286–2296.

[73] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[74] H. Wu, B. Xiao, N. Codella, *et al.*, "Cvt: Introducing convolutions to vision transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 22–31.

[75] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial nets," in *Neural Information Processing Systems*, 2014. [Online]. Available: https://api.semanticscholar.org/CorpusID:261560300.

[76] F. N. Iandola, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size," *arXiv preprint arXiv:1602.07360*, 2016.

[77] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *CoRR*, vol. abs/1710.10196, 2017. arXiv: 1710.10196.

[78] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[79] B. Graham, A. El-Nouby, H. Touvron, *et al.*, "Levit: A vision transformer in convnet's clothing for faster inference," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 12 259–12 269.

[80] V.-T. Hoang and K.-H. Jo, "Practical analysis on architecture of efficientnet," in *2021 14th International Conference on Human System Interaction (HSI)*, IEEE, 2021, pp. 1–4.

[81] S. Mehta and M. Rastegari, "Separable self-attention for mobile vision transformers. arxiv 2022," *arXiv preprint arXiv:2206.02680*,

[82] S. Mehta and M. Rastegari, "Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer," *arXiv preprint arXiv:2110.02178*, 2021.

[83] Z. Liu, Y. Lin, Y. Cao, *et al.*, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.

[84] J. Pan, A. Bulat, F. Tan, *et al.*, "Edgevits: Competing light-weight cnns on mobile devices with vision transformers," in *European Conference on Computer Vision*, Springer, 2022, pp. 294–311.

[85] A. Srinivas, T.-Y. Lin, N. Parmar, J. Shlens, P. Abbeel, and A. Vaswani, "Bottleneck transformers for visual recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 16 519–16 529.

[86] S. Chaudhary, V. Gupta, K. Deo, and K. Sharma, "Melanoma skin cancer detection using involutionnet," in *2024 Control Instrumentation System Conference (CISCON)*, IEEE, 2024, pp. 1–6.

[87] W. Wang, E. Xie, X. Li, *et al.*, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 568–578.

[88] B. M. Abuhayi, Y. A. Bezabh, and A. M. Ayalew, "Inv-alxvggnets: Cervical spine disease classification using concatenated involutional neural networks with residual net," *IEEE Access*, 2024.

[89] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *International conference on machine learning*, PMLR, 2021, pp. 10 347–10 357.

[90] P. Pradhan, B. Kumar, K. Kumar, R. Bhutiani, *et al.*, "Plant disease detection using leaf images and an involutional neural network," *Environment Conservation Journal*, vol. 25, no. 2, pp. 452–462, 2024.

[91] D. Li, J. Hu, C. Wang, *et al.*, "Involution: Inverting the inherence of convolution for visual recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12 321–12 330.

[92] A. G. Howard, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[93] T. Rahman, *Covid-19 radiography database*, Accessed: 2025-01-24, 2020. [Online]. Available: https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database.

[94] M. K. U. Ahamed, M. M. Islam, M. A. Uddin, *et al.*, "Dtlcx: An improved resnet architecture to classify normal and conventional pneumonia cases from covid-19 instances with grad-cam-based superimposed visualization utilizing chest x-ray images," *Diagnostics*, vol. 13, no. 3, 2023.

[95] D. Alexey, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv: 2010.11929*, 2020.