Identification of Pneumothorax from Chest X-ray Images Using Artificial Intelligence Techniques.

Author

Tahira Iqbal

NUST CEME 00000273391

Supervisor

Dr. Arslan Shaukat

DEPARTMENT OF COMPUTER AND SOFTWARE ENGINEERING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY,

ISLAMABAD

FEBRUARY, 2021

Identification of Pneumothorax from Chest X-ray Images Using Artificial Intelligence Techniques.

Author

Tahira Iqbal

NUST CEME 00000273391

A thesis submitted in partial fulfillment of the requirements for the degree of

MS Computer Engineering

Thesis Supervisor:

Dr. Arslan Shaukat

Thesis Supervisor's Signature:_____

DEPARTMENT OF COMPUTER AND SOFTWARE ENGINEERING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING

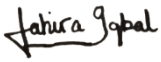
NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY,

ISLAMABAD

FEBRUARY, 2021

Declaration

I certify that this research work titled "*Identification of Pneumothorax from Chest X-ray Images Using Artificial Intelligence Techniques."* is my own work. The work has not been presented elsewhere for assessment. The material that has been used from other sources it has been properly acknowledged / referred.



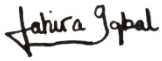
Signature of Student

Tahira Iqbal

NUST CEME 00000273391

Language Correctness Certificate

This thesis has been read by an English expert and is free of typing, syntax, semantic, grammatical and spelling mistakes. Thesis is also according to the format given by the university.



Signature of Student

Tahira Iqbal

NUST CEME 00000273391

Signature of Supervisor

Dr. Arslan Shaukat

Copyright Statement

Acknowledgements

I am extremely thankful to Allah Almighty for giving me patience, guidance and ability to carry out this work. Without HIS blessings I could not have completed my thesis. Indeed Allah is the most merciful and worthy of praise.

I would like to express my gratitude to my supervisor Dr. Arslan Shaukat and co-supervisor Dr. Muhammad Usman Akram for their guidance, valuable suggestions and moral support in the entire duration. It's a blessing to have such great people as mentor for research.

I am also thankful to the entire thesis committee: Dr. Farhan Hussain and Dr. Wasi Haider for their tremendous support and cooperation.

My acknowledgment would be incomplete without mentioning my beloved parents without whom I am nothing and who have supported me in every second of my life. I would also like to thank my sisters who have always supported me and especially during the research work.

Finally, I would like to express my gratitude to my friends and all the individuals who have encouraged and supported me throughout the entire duration.

Dedicated to my exceptional parents and adored siblings whose tremendous support and cooperation led me to this wonderful accomplishment

Abstract

Technology today has revolutionized the world by replacing the manual system with automatic ones by deploying artificial intelligence, which enables the system to mimic human brain by making wise decisions on the basis of the past experiences. In this research, such computer aided design is proposed which is able to distinguish between pneumothorax and normal X-ray and also solves the class imbalance issue which is troublesome in most of the machine learning classification problems. Such system will help in minimizing the risk of pneumothorax which is a life threatening disease. The proposed CAD system consists of two modules, i.e. classification of chest radiographs as normal or pneumothorax and segmentation model for identifying the location of pathology. For pneumothorax classification, firstly existing approaches for class imbalance are experimented and after finding out that data-level-ensemble outperforms others, an ensemble model is proposed which is actually a model-level-ensemble of multiple data-level-ensembles. The different models used in this framework are three different CNN architectures including VGG-16, DenseNet-121 and VGG-19. These architectures are used as fixed feature extractor and support vector machine is used as classifier. The proposed framework is experimented on two datasets: SIM ACR Pneumothorax dataset and Random Sample of NIH-Chest X-ray dataset (RS-NIH). The model has achieved testing score of 86.0% Area under the Receiver Operating Characteristic curve (AUC) with 85.17% recall on SIIM dataset, while on RS-NIH 95.0% AUC with 90.9% Recall is achieved with random split of data and 77.06% AUC with 85.45% Recall is achieved with patient-wise split of the dataset. Our model has performed very well on both the datasets as the AUC achieved on RS-NIH is the best achieved so far, while for SIIM dataset, a direct comparison cannot be made as we are the first to use this dataset for classification. For identification of area of pathology in the CXRs, a two stage segmentation framework is proposed in which the main building block is U-Net architecture with EfficientNet-B4 encoder. The images and corresponding masks are resized to 256 x 256 and 384 x 384 for training the first and second stage respectively, using the SIIM dataset. 84.56% dice score is obtained for the segmentation model. Our results prove that the proposed techniques can be generalized to any other medical imagining domain classification and segmentation problem.

Key words: *class-imbalance, chest x-rays, classification, deep learning, pneumothorax, segmentation.*

Table of Contents

List of Figures

List of Tables

CHAPTER 1: INTRODUCTION

Technology has revolutionized the world by automating many processes which were human dependent earlier. Starting from the minimal device such as calculator to giant machines in factory, technology is everywhere. The machines are made to follow the set of instructions/commands in order to do the job without human supervision. With more advancement in technology Artificial Intelligence came to revolutionize the world beyond human imagination. By utilizing AI, the machines are taught to mimic the human brain by learning from experience. Like any other real life field, using AI in medical field has also become a topic of interest, and the first effort to make a Computer Aided Design (CAD) was made in 1960s. In the field of medicine, AI techniques have been deployed in order to automate the detection of several diseases like detection of cancer and identifying thoracic pathologies. Automatic detection of chest diseases form chest radiographs has become a hot topic and many researchers have contributed in this field by proposing several CAD systems. Such systems help the radiologist in identifying the diseases in lesser time and also act as a second opinion for the radiologist in making correct decision regarding the pathology. However in order to enable such systems to make unsupervised decision, transfer learning needs to be done which required large amount of data and it has been observed that most medical images dataset are imbalance in nature. So this research presents a CAD based automatic diagnosis of Pneumothorax from chest X-rays along with proposing a solution for class imbalance issue.

1.1 Motivation

Pneumothorax is a life threatening diseases which affects the lungs. According to a study [1], thoracic trauma is becoming one of the primary cause of death in developing countries. So early diagnosis of pneumothorax is important in order to save people from losing their lives. Moreover to carefully observe the presence of pneumothorax from chest radiograph is a cumbersome task as the area of pneumothorax range from small line to whole lung area. Additionally according to a survey almost 2 billion CXRs are taken per year and a single radiologist has to read 100 Chest X-rays per day so in order to lessen the work load on the doctor and to assist him in making decision, an automated system is needed which not only differentiate

between normal and pneumothorax chest radiographs but also identifies the location of the pathology. Secondly there is class imbalance problem in most of the medical images data and it is impossible to say which technique is better among all the existing techniques, as it is domain dependent [2]. Thus, designing an automatic detection system which learns from a class imbalance dataset and maximizes the truthfulness in identification of pneumothorax CXRs is the motivation behind this research.

1.2 Problem Statement

Timely diagnosis of pneumothorax is important in order to avoid more damage to the lung. To date, many CAD systems have been proposed for the said purpose however due to the non-availability of the data used in those researches, further improvement can't be made. Moreover to date, most of the pneumothorax chest x-ray datasets are imbalance in nature. So the purpose of this research is to explore different AI techniques in order to propose an efficient diagnostic system while catering the class imbalance problem.

1.3 Aims and Objectives

Main objective of this research are:

- Review and compare the existing work for the automatic diagnosis of pneumothorax and identification of location of pathology.
- Study the different approaches to solve the class imbalance problem along with proposing a framework for differentiating between normal and pneumothorax chest x-rays.
- Proposing a segmentation model for identifying the location of pneumothorax in the chest x-ray using artificial intelligence techniques.

1.4 Structure of Thesis

This report is structured as follows:

Chapter 2 briefly describes the vital organ for respiration which are lungs and their anatomy. Furthermore, the details of pneumothorax and its treatment is also discussed.

Chapter 3 covers the review of the existing work done for the detection of pneumothorax by the researchers in past years

Chapter 4 describes the proposed framework in detail. It consist of two main modules: classification of pneumothorax and segmentation of the area of pathology

Chapter 5 gives the overview of the databases and the performance measures used for evaluation of the proposed framework. All the results are discussed in detail along with the tables and figures required.

Chapter 6 concludes this research and presents the future scope of this thesis.

CHAPTER 2: PNEUMOTHORAX AND ITS RADIOLOGY

In human body, there are five organs that are extremely crucial for survival, these are heart, brain, liver, kidney and lungs. The vital organ of respiratory system are lungs and the working mechanism is the exchange of carbon dioxide and oxygen between body and the air. In this chapter the anatomy and imaging techniques for lungs along with the brief introduction about pneumothorax will be discussed.

2.1 Structure of Lungs

The lungs are most important organ of the respiratory system, and are composed of tissues and muscles. These are located in the thoracic cavity of the chest between rib cage and the diaphragm. The lungs are divided in five subsections (lobes) and their job is to take-out carbon dioxide from the body and take-in oxygen into the body. The heart works in collaboration with lung in order to do the respiration process [3].

Thousands of thin tubes are present in the lungs and their ending are kind of sacs which are actually small air sacks known as alveoli. Every single sac is covered with bunch of blood vessels connected to the system of arteries and veins which are used for circulating blood throughout the body.

When a person breaths, huge amount of blood rich of carbon dioxide and zero oxygen level is brought by the pulmonary (lung) artery. This air enters the body through nose or mouth and travel through the pharynx to the trachea. The air travels down till it reaches bronchi which connects the lungs to the trachea. The blood releases this carbon dioxide into the air and allows the oxygen molecules to enter from air into the body. Once the blood has adequate amount of oxygen and no carbon dioxide left in it, it is returned to the heart and then pumped to the body. The structure of lungs and thoracic cavity is shown in Figure 2.1.

In order to expand and relax during the breathing process, lungs required several muscles which are described below:

- **The diaphragm**

  It is present between the lungs, and it is the most important muscle needed for breathing. Chest is separated from the abdominal cavities via diaphragm muscle and they also helps the lungs to inflate by contracting itself.

- **Intercostal muscles**

  These are present between the ribs and help the lung during breathing process by expansion and contraction of the chest cavity.

- **Abdominal muscles**

  The organs present in the abdominal cavity are compressed by these muscles in order to assist the lung in expansion and contraction during breathing process.

- **Neck and upper chest muscle**

  These muscles are used when some lung damage or disease make it difficult for the person to breath.

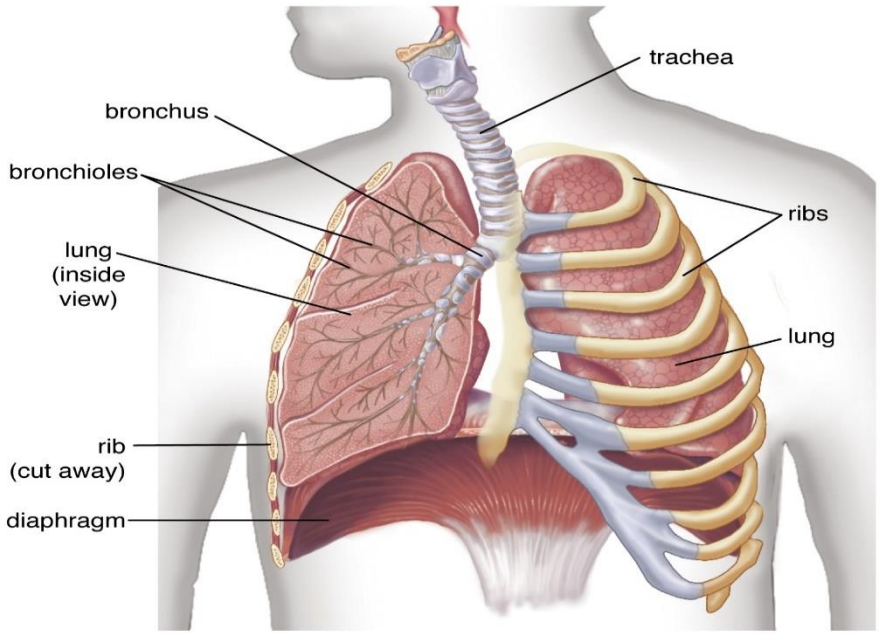
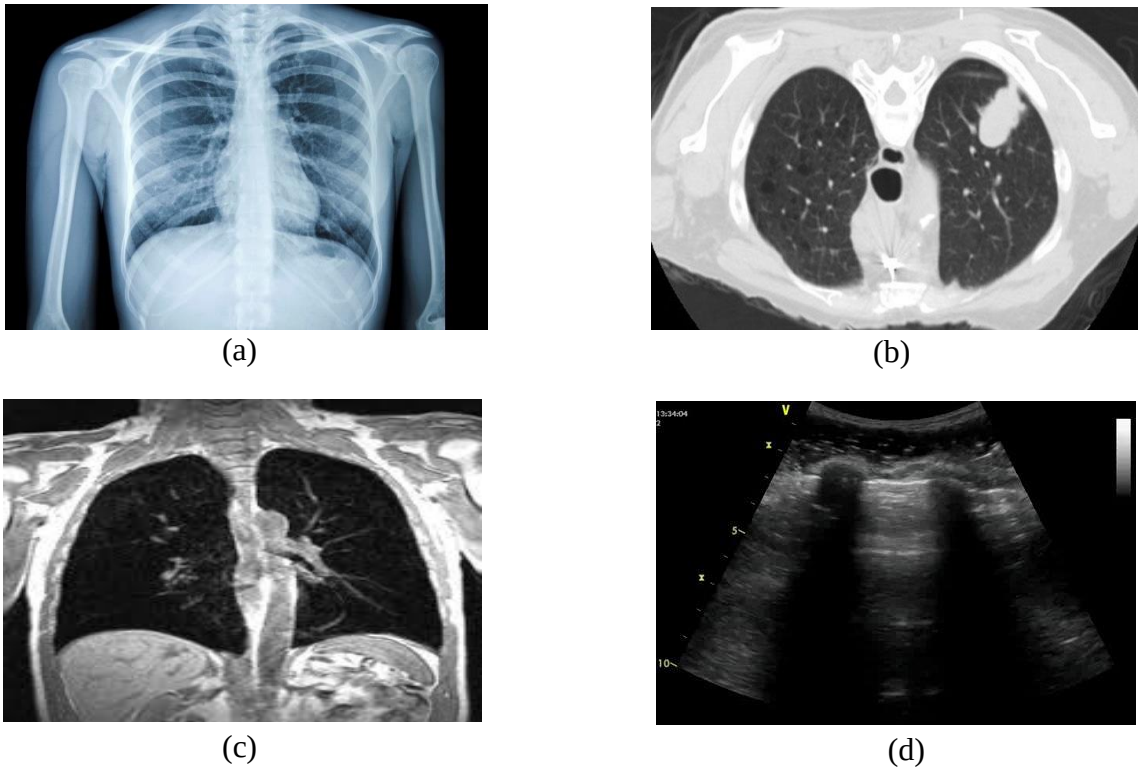


Figure 2.1: Structure of Lungs and Thoracic cavity [4]

2.2 Techniques to Analyze Lungs

For medical analysis, medical imaging techniques are used for visual representation of body. The different medical imagining techniques available include X-rays, Magnetic Resonance Imagining (MRI), Computed Tomography (CT) scan and Ultrasound. X-rays are obtained by passing X-ray radiations through the body which produce an image on the x-ray file according to the density of the objects in the body with which it strikes. X-rays provide 2D view of the body part. CT scans are done via CT machines which generates 3D image of the body organ under consideration. MRI provides visualization of the body by combining powerful magnetic field and radio ways with advanced computer technologies, without using any radiations. Ultrasound provide image of internal body parts by using sound wave which have higher frequency compared to human audible range. For analysis of lungs, mostly X-rays and CT scans are used however because of low cost of X-rays and relatively less amount of radiations as compared to CT scans, X-rays are more preferred by the doctors. Figure 2.2 shows the different Medical Imagining techniques for visualization of lung and thoracic cavity.



(a) (b)

(c) (d)

Figure 2.2: Different Medical Imagining Techniques for lung visualization. (a) X-ray. (b) CT-scan. (c) MRI. (d) Ultrasound

2.3 Pneumothorax

Pneumothorax is a disease in which lungs collapse. It occurs when the lung from inside of the lung is leaked into the space between lungs and the chest walls. The air exerts force on the outer walls of the lungs, and since the lungs have air present inside it as well, so this creates pressure and the lungs collapse. Pneumothorax can effect whole lung or only small portion of lung might be effected. In Figure 2.3 the right hand side lung is normal while lung on the left hand side is collapse and air is filled into the pleural space.

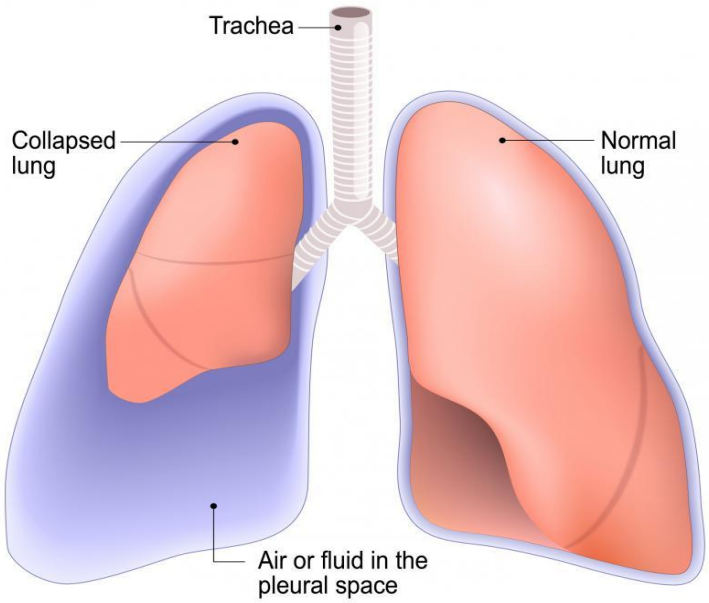


Figure 2.3: Collapsed lung (pneumothorax) and Normal lung [5]

2.3.1 Causes of Pneumothorax

Some of the major causes of pneumothorax include chest injury or other lung diseases, however sometimes it may even occur without any obvious reasons. Some of the causes are discusses briefly in this section.

- **Chest Injury**

  A blunt or a certain injury penetrating the chest can be a reason of lung collapse. Some of these injuries might happen during car crashes or physical assault, while some might occur during certain medical procedures in which needle is inserted into the chest.

- **Lung diseases**

  A damaged lung tissue is more likely to collapse as compared to healthy one. The lung tissues damage might be caused by several underlying diseases including asthma, tuberculosis, chronic obstructive pulmonary diseases (COPD), cystic fibrosis and whooping cough.

- **Raptured air blisters**

  Small blebs (bubbles) can develop on the lungs which sometimes might burst. These busted blebs can allow the air to leak from inside of the lungs into the pleural space.

- **Mechanical ventilation.**

  People who need mechanical assistance to breath may suffer from sever kind of pneumothorax. It occur because the ventilator can cause imbalance of air pressure inside the chest cavity. Thus the lung may collapse.

2.3.2 Risk Factors

Pneumothorax is more common in men as compared to women. It is more likely that people in between age 20 to 40 years, especially if they are abnormally tall and underweight might develop the type of pneumothorax which is caused by the reputed air blebs.

Some of the risk factors include:

- Smoking
- Genetic issues
- Previous pneumothorax

2.3.3 Symptoms of Pneumothorax

Commonly known symptoms of pneumothorax include:

- Sudden sharp pain in chest or shoulder which is worse if person has cough or breathing issues
- Shortness of breath
- Nasal flaring (widening of nostrils which occur from shortness of breath)

More severe symptoms are observed in case of wide lung area collapse. These includes:

- Bluish color of skin (occur because of lack of oxygen)
- Feeling of becoming faint
- Chest tightness
- Fatigue
- Rapid heart rate
- Abnormal pattern of breathing

2.3.4 Diagnosis

The doctor will suspect collapse lung by observing the breathing pattern and asking the patient about symptoms felt and medical history. The health personnel will examine the patient by focusing on the vital signs including temperature, breath rate, pulse rate and blood pressure. Moreover visualization of lungs by means of medical imaging technique will also be done [6]. The doctor check following signs to declare if the person has pneumothorax or not.

- Low blood pressure
- Increased heart beat rate
- Loss of normal breath sounds in the area of the chest where the lung is damaged
- Low level of oxygen of oximeter
- Hollow sound when fingers are tapped on the damaged part of the chest.

The different medical imagining tests for pneumothorax detection might include Chest X-rays, CT scan, Arterial blood gases (blood test in which the blood from wrist artery is taken in order to

get detailed measurement of the oxygen and carbon dioxide in blood) and Electrocardiogram (ECG). However mostly the doctor prefer to recommend Chest X-rays because of its cheap cost and availability of X-ray machines in almost everywhere. The area of the chest affected by pneumothorax will appear dark black on the Chest X-ray which will confirm the lung collapse.

2.3.5 Visual Identification of pneumothorax from Chest X-ray

As mentioned earlier, usually X-rays are preferred in order to confirm the collapse lung. Figure 2.4 shows the labelled X-ray image of a healthy person.

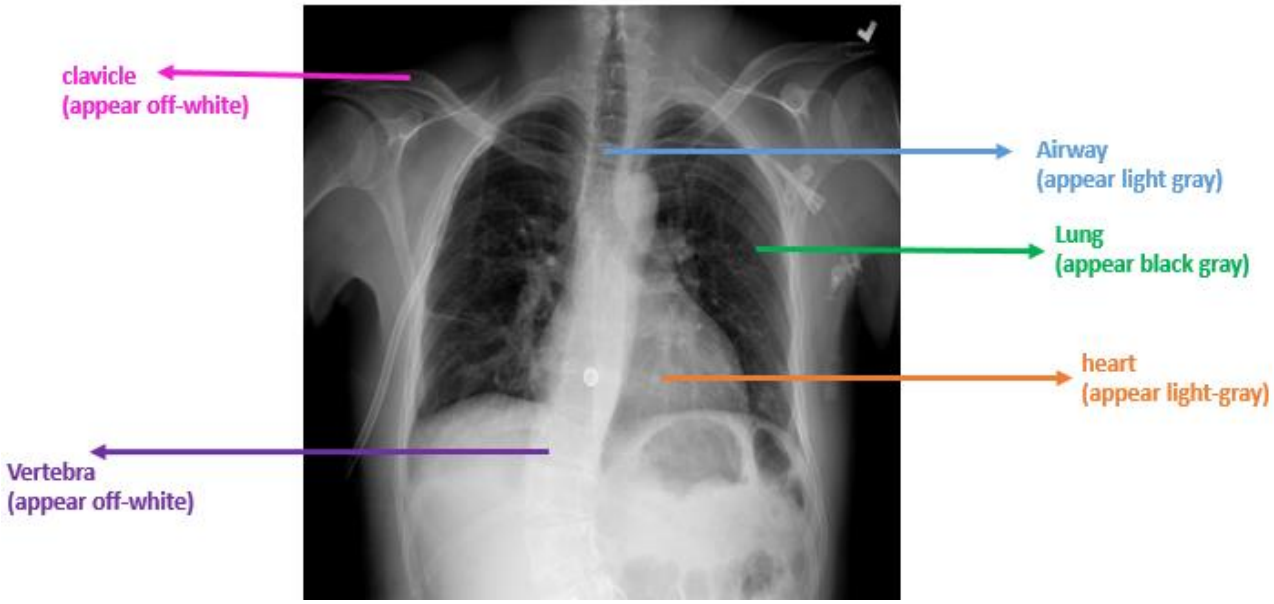


Figure 2.4: Labeled diagram of Normal Chest X-ray

In a normal CXR, the lungs appear black gray, the bones appear off-white while the heart appears light gray. If there is dense air or fluid present inside the pleural cavity (containing the lungs), then it appears dark black. Moreover there is a fine line which covers the boundary of the lung and it is not visible in a normal lung Chest X-ray.

In order to check if a person has pneumothorax or not, the doctor check two things [7].

- If there is some area which is dark black in the pleural cavity, as in normal cases, lung appear black gray however if some portion of whole lung appears dark black and there is noticeable difference between the appearance of the two lungs in terms of colors, then it is a sign of collapsed lung.
- If a thin line outside the lateral aspect of the lung is visible or not. If the line is visible then the person should be given proper treatment.

If either of the above two conditions fulfil, then it a guaranteed sign that the person is suffering from pneumothorax. Figure 2.5 shows the X-ray image with collapsed lung [8].

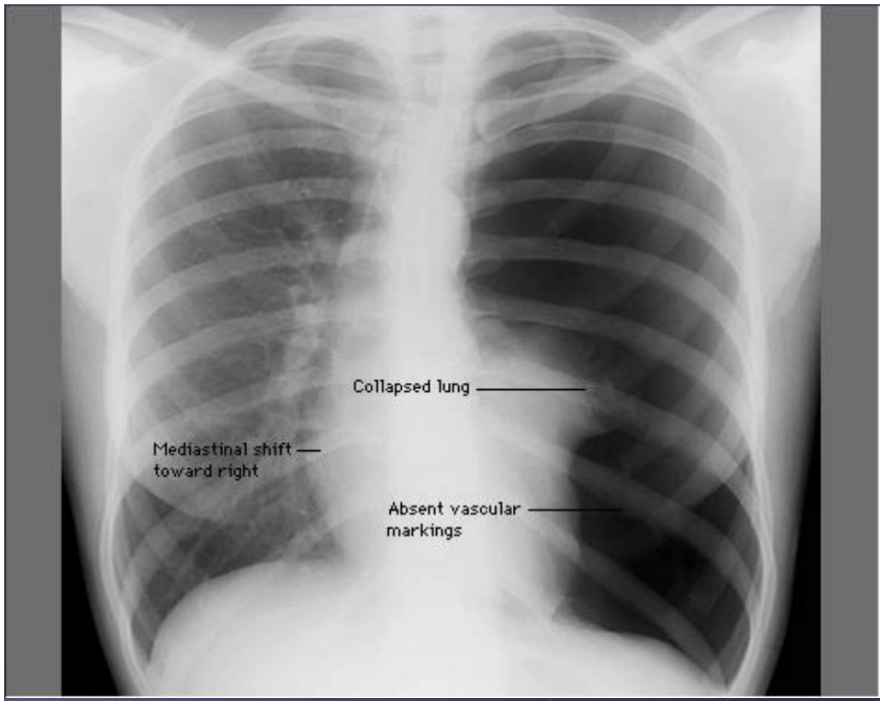


Figure 2.5: Chest X-ray with pneumothorax

2.3.6 Prevention

Most of the different types of lung collapse can't be prevented however quitting smoke can reduce the risk of developing certain type of lung diseases. Avoiding activities that can cause chest injury can also reduce the risk of being affected by pneumothorax.

2.3.7 Treatment

Several ways of treating pneumothorax are present, which are suggested by the doctor based on the location and size of the collapsed lung and the medical history.

The different treatments include:

- Continuous observation of the patient's condition by the doctor to see if the collapse lung heals itself or not.
- Using needle and syringe for the removal of air from the pleural cavity.
- In case of large pneumothorax, the air is removed by inserting chest tube, which is a hollow plastic tube, between the ribs and attaching the suction device at the other end.

If a person has sever pneumothorax, he may be admitted in hospital and will likely receive oxygen through mask or nasal prongs. The patient may be kept under observation for serval days in order to ensure that the collapse lung is normalized. In case the applied treatment is not positively effecting the collapsed lung then surgery may be advised in order to repair the lung and the pleural cavity. Traditional or old techniques needs the chest to be opened however with advancement in technology tiny cameras (thoracoscopy) are used which causes less scars and results in fast recovery.

People who experience the pneumothorax over and over again may need to be treated in order to remove the root cause. Such treatments include:

- Surgery to remove the air blisters or the injured area
- An injection that fuse the chest wall and the lungs.

2.3.8 When to Consult a Doctor

A person should call a doctor if he feels any symptom of pneumothorax, especially if a person is already suffering from a lung disease. Even if the symptoms appear to decrease, still consult a doctor as the symptoms may disappear while the lung is still not healed.

2.3.9 Prognosis

Once the pneumothorax is normalized, and the lung is healed and re-expanded to normal size, no side effects appear in the body however there are almost 50% chances that the collapse of lung occur again within next few months. To lessen this risk some precautions should be taken including:

- Avoid smoking
- Avoid areas which may lead to air pressure fluctuation, such as scuba diving or flying in an aircraft which is unpressurized.

CHAPTER 3: LITERATURE REVIEW

Chest X-rays are one of the most common examination mean for detection of several diseases. There are some other means as well, like CT scan, however CT scan is an expensive screening method. In contrast, X-ray is more preferred by patients as well as doctors because of its cheap cost and availability of X-ray machine in almost every clinical setup. Using Chest X-rays, different pathologies can be detected, including pulmonary nodules, cardiac abnormalities, pneumonia, pneumothorax etc. Large amount of information is present in the chest radiograph of the patient in terms of different organs of the thoracic region, so it is a challenging task for the doctor to analyze the radiographs. It is because of the complex overlapping structure of the tissues that makes it a challenge to interpret the Chest X-rays. The importance of Chest radiography can be noticed by observing the fact that over 2 billion Chest X-rays are taken every year throughout the world. [9]. With the advancements in computer technologies, a computer aided system (CAD) can help in better diagnosis of the disease from the chest X-ray as well as gives the radiologist a second opinion about their diagnosis. In past years, wide range of work has been done in the field of automatic detection of different pathologies including Pneumothorax using Chest Radiographs. Some of the researchers presented their work for classification of the pneumothorax, while few have designed segmentation models which efficiently identifies the area of lung with pathology.

However it has been observed that most of the medical images datasets are imbalance in nature, i.e. the number of samples of each class in a dataset is not the same [10]. Even there are dataset available which contains very few samples of positive class as compared to negative class samples. So it is important to tackle this problem along with presenting a classification or segmentation model. This chapter summarizes the valuable work done in the domain of class imbalance problem and automatic detection of pneumothorax from chest Radiographs.

3.1 Class Imbalance:

Designing an automated way of disease diagnosis comes under classification category of machine learning. However it has been observed that most of the classification problem especially dealing with medical images dataset encounters class imbalance problem. So several researches have been published in which this problem has been studied by comparing different existing approaches. These approaches can be divided into two main categories [11]. Data level methods:

in which the dataset is altered in such a manner that each class has equal number of samples. Classifier level methods: in which different machine learning algorithms are adjusted by means of tune-able parameters. We will first describe these techniques and then brief overview of the work done in the said field is presented in this section.

3.1.1 Class imbalances approaches:

The different existing for solving the class imbalance problem are explained below:

i. ***Weight balancing (Classifier Level Method):*** In this approach, whole training set is fed to the classifier as it is, however different weights are assigned to each class based on the frequency of occurrence of each class [12]. Thus while learning, the classifier gives more importance to the samples with more class weights. One of the way to assign weights is by using the formula given below:

$$class\ weight = \frac{n_{samples}}{n_{classes} * np.bincount\ (y)} \tag{3.1}$$

Here $n_{samples}$ represents sample size of the training set, $n_{classes}$ denotes the total number of class in the dataset and $np.bincount\ (y)$ gives total count of the samples of each class.

ii. ***Over-sampling (Data Level Method):*** As the name suggests, in this approach the total number of minority class samples is increased in such a way that it becomes equal sample size of majority class [13]. Some of the techniques for this approach include SMOTE [14], Cluster based oversampling [15] and Data-Boost-IM [16]. Another technique, also experimented in this research is Data Augmentation [17], by means of which the samples size of minority class can be increased.

iii. ***Under-sampling (Data Level Method):*** This approach solves the class imbalance problem by reducing the total number of samples in majority class making it equal to sample size of minority class [18]. As enormous number of samples are discarded in this approach, it may lead to under-fitting, however it is still found to be effective in some cases.

iv. ***Ensemble (Hybrid approach):*** This method is the combination of multiple techniques from both or one of the above mentioned approaches. One of the data-level ensemble approach as described in [19] assumes equal distribution of samples in a dataset, and small subsets are created with the same class distribution as in the original dataset. The ensemble model approach, which is utilized in our research finds its root from [20], in which subsets of training data are created in such a way that each subset contains whole minority class samples and same number of samples from majority class. These subsets of training data which are now class balanced, can be utilized for training a machine learning algorithm. There are two famous Classifier level ensemble methods known as Bagging and Stacking-C [21]. In Bagging, different subsets of training data are created by randomly picking samples from training data and named as bootstrap samples, these samples are independently trained on different classifiers and then voting or averaging of the results is done in order to obtain a single prediction. Note that the different training subsets can be trained using either a single or different classifiers. Stacking-C is a classifier level ensemble in which whole training dataset is trained on different machine learning classifiers known as base classifiers and predictions are made individually by each base classifier on the validation set. These predictions along with ground truth are fed to Meta level machine learning classifier which train itself in order to give predictions for the un-seen test set. This method reduces overfitting.

3.1.2 Existing Work:

Several researchers have published their work in order to compared different approaches for class imbalance using commonly available datasets. Buda *et al* [11] made comparison of different existing approaches including oversampling, under-sampling and thresholding. The classifiers used were different CNN architectures. It was found that oversampling performed best of all the three approaches experimented for the class imbalance problem. Another important observation of this experiment was that in contrary to the machine learning algorithms, CNN doesn't encounter the problem of overfitting when oversampling technique was used. The authors used three different datasets, MNIST CIFAR-10 and ILSVRC-2012 with three different CNN architectures LeNet-5, All-CNN, ResNet-10 respectively, for the experiment purpose.

Raskutti *et al* [12] compared the two class imbalance approaches including weight balancing and under-sampling. The experiments were performed using two textual datasets which

include AHR data and Reuters data. The selected classifier was Linear SVM. The results of these experiments proved that under-sampling surpass weight balancing technique.

Zhongbin [16] presented a novel ensemble technique while comparing the different existing approaches including sampling methods, bagging and cost sensitive learning techniques. The ensemble model presented in this research is designed by creating subsets of training data in such a manner that minority class samples are combined with equal instances of majority class and for each subset different machine learning algorithm was used as classifier. The classifiers employed in this research include Naïve Bayes, C4.5, Random Forest, SMO, RIPPER and IBK. The results from each classifier were then combined together by Max Distance Rule of ensemble modelling. The AUC was calculated for each experiment performed on different datasets of KEEL repository and it was found that proposed ensemble model performs better than other approaches compared.

In [21] Salunkhe et al made comparison of oversampling and under-sampling approaches using KEEL repository datasets. For oversampling SMOTE technique was utilized to increase the sample size of minority class. K fold cross validation was used for experimentation purpose with K set to 5 and Classifier ensemble was created by utilizing three different classifiers which include J48, Logistic Regression and Bagging. Each classifier was fed with different fold of training data and it was found that this approach gave better AUC value as compared to a normal stacking ensemble in which single dataset is used for training purpose.

Olivier [22] gave their contribution in the field of solving the class imbalance problem by experimenting two approaches including oversampling and under-sampling. Three different type of classifiers were used in order to determine which of the technique performed better. The selected classifiers were CNN, Radom Forest and SVM with performance measure chosen as F1. The research was carried out on COCO dataset and it was found that CNN performed better without any sampling technique. SVM gave equal performance with all the different sampling techniques while under-sampling was found to enhance the results in case of Random Forest classifier.

Joffrey [23] published a survey of existing work on class imbalance problem. The summarized overview of the work done so far in this field along with the achieved results were presented in which it was found that data-level methods (i.e. sampling techniques) as well as classifier-level techniques performed best in different cases and none of the technique can be said as best one.

From the literature it has been observed that none of the research has been done using a CXR dataset, moreover no generic rule can be made regarding the class imbalance problem. The summarized literature review for the class imbalance problem is shown in Table 3.1.

Table 3.1: Literature Review on Class Imbalance Problem

Author	Year	Summary	Dataset	Result
Buda [11]	2018	Compared oversampling under-sampling and thresholding.	MNIST CIFAR-10 ILSVRC-2012	Oversampling was found better
Raskutti [12]	2004	Compared under-sampling and weight balancing.	Textual Data: AHR-data Reuter-data	Under-sampling performed better
Zhongbin [20]	2014	Proposed a novel ensemble model by creating subsets of data and training each data with a specific classifier. Results were combined with Max. distance ensemble rule.	KEEL repository	Ensemble performed better
Salunkhea [21]	2016	Proposed a Classifier Ensemble in which different subsets of data were created after random sampling and different classifiers were used for different subsets of data.	KEEL repository	Classifier ensemble perform better than Stacking or Bagging

Olivier [22]	2019	Compared under-sampling and oversampling techniques using different classifiers	COCO dataset	All classifiers performed differently with different sampling approaches
Joffrey [23]	2018	Survey of existing work on class imbalance	N/A	Class imbalance solution is dependent on dataset

3.2 Classification

Like any other classification problem, for designing a medical image classification model, features are extracted from the dataset and a classifier is trained on those features in order to differentiate between normal and abnormal cases. In past years, several researches have been done in the field of chest pathologies detection including automatic diagnosis of pneumothorax from chest radiographs. The published work in this field is summarized below.

In [24] a model was presented in which features were extracted using local binary pattern and fed to SVM for training. These features were extracted after identification of lung region by utilizing multiscale intensity segmentation method. The proposed model was experimented on a local dataset obtained from Chung Shan Medical University Hospital, Taiwan. The dataset was comprised of 32 CXRs with pneumothorax and 52 CXRs with no pathology. The results with 5 fold cross validation were presented with mean Accuracy of 82%.

In [25] experiments were performed in order to differentiate between normal and abnormal cases. Different thoracic pathologies were studied as binary classification problem. One of the pathology under consideration was pneumothorax. 78 Animal CXRS containing 39 normal and 39 pneumothorax cases obtained from a database at local hospital were utilized in this research. Two different classification approaches were utilized, first one was features extraction by BoF (bag of features) technique and SVM as classification while second one was using CNN for feature

extraction and classification with soft-max classifier. It was found the CNN outperformed other approach with achieved result of 100% accuracy.

Park [26] proposed a pneumothorax detection model by training a YOLO-pre-trained model on a dataset containing 1596 Pneumothorax and 11137 normal CXRs obtained from local hospital. The class imbalance problem was solved by thresholding technique in which a prior probability is multiplied with the obtained probability value before outputting the final result. The model was evaluated on a test set containing 253 pneumothorax and 250 Normal CXRs and achieved result AUC of 90.8%.

In [27] classification model was designed using a dataset containing 1003 DICOM images obtained from University of Washington Medical Center. Three different classification models including CNN, Multiple Instance Learning (MIL) and Fully Convolutional Networks were utilized with different input sizes for each classifier separately. It was found that CNN performed best of all with AUC of 96% on a test set of 87 normal and 113 Pneumothorax CXRs.

In [28], an eight layered CNN model was designed for automated diagnosis of pneumothorax and the model was trained on 280 CT scans from which 36×36 patches were extracted. The features extracted from the proposed CNN architecture were fed to SVM classifier, and evaluation was performed on the test set containing 160 CXRs with pathology and 40 normal CXRs. The model achieved 94% mean accuracy on the test set.

Tony [29] used thoracic ultrasound in B-mode video and M-mode images separately. Pre-trained VGG-16 model was fine-tuned on the ultrasound images. The dataset was acquired from a local database. Different image processing techniques were used in order to enhance the image quality. The model performance was evaluated on validation set containing B-mode 4053 frames and 467 M-Mode images which obtained accuracy of 99.78% and 98.29% respectively.

In [30] texture analysis based technique was used to design an automated model for pneumothorax detection. Feature extracted by texture analysis technique from training set were fed to KNN classifier. The proposed model was tested on 41 Normal and 48 Pneumothorax CXRs and achieved sensitivity value of 81% and specificity of 87%.

In [31] Subrato designed a model named as VDSNet by combination of VGG, Data augmentation and spatial transformer network STN. Binary classification was performed in order to differentiate normal CXRs and abnormal CXRs from a dataset containing 14 different chest pathologies. Experiments were performed using Random Sample NIH Chest Xray-14 (RS-NIH)

dataset and full NIH Chest X-ray-14 (NIH) dataset. The model achieved Accuracy of 70.8% on RS-NIH dataset and 73% on full NIH CXRs dataset. Along with utilizing the image features, metadata including patient age, gender etc. was also used in training the model.

In [32] a classification model was designed for differentiation of normal and abnormal CXRs. 4479 Normal CXRs from NIH dataset were used for training a GAN (generative adversarial network) model. The main idea was to train the model on normal CXRs only. At test time, if the model perfectly recreated the input CXR, then it was assumed to be a Normal CXR else it was declared as abnormal CXR. AUC of 84.1% was achieved on a test set containing 667 normal and 667 abnormal CXRs.

Qingji [33] proposed AG-CNN which was an attention guided convolutional neural network based on ResNet50 architecture for diagnosis of 14 chest diseases. The model was trained on global image as well as local image which contained the portion of pathology. These global and local images were fed to AG-CNN model for multi-label classification. The training was done using 70% samples from NIH dataset and tested on 205 CXRs. The proposed model achieved AUC of 92.1% on pneumothorax detection.

In [34] Pranav proposed a 121-layered DenseNet named as ChexNet, initially proposed for pneumonia detection while later it was extended to thee multiclass classification problem in order to diagnose 14 different pathologies from the CXRs. Training was done on 70% data samples of NIH while the proposed model was evaluated on a validation set containing 420 samples. The AUC achieved on pneumothorax detection was 88.87%.

Imane [35] proposed a multi-label classification model by training DenseNet-121 architecture on NIH and CheXpert dataset separately. The training was done by keeping image size 224×224. Different approaches for tackling the multi-label classification were experimented. The model performance was evaluated on test set of both datasets separately. The proposed model achieved AUC of 92.9% on NIH dataset and 84% AUC on CheXpert dataset for pneumothorax detection.

In [36] automatic detection of multiple thoracic pathologies was done by fine tuning the pre-trained Xception mode. RS-NIH dataset was used for this purpose which was randomly split into 75-25, i.e. 75% was used for training purpose while 25% was used as test set. The images were down-sampled from 1024x1024 to 128x128. All 14 classes were considered in this

experiment. The obtained accuracy for overall test set was 88.76% however the AUC for pneumothorax was 54%.

In [37] Tae proposed an ensemble model created from three different trained models. ResNet-50 model was used with three different input resolutions. NIH dataset was used for the research. 80% of data was used for training the model and remaining 20% was used to evaluate the model. The proposed model achieved AUC of 91.1% on the test set. Table 3.2 summarize the existing work on pneumothorax classification.

Table 3.2: Literature Review on Classification

Author	Year	Dataset	Technique	Results
Yuan [24]	2018	Local dataset	LBP+SVM	Acc=82%
Yoon [25]	2018	Local dataset	BoF + SVM and CNN (Own architecture)	Acc= 100%
Park [26]	2019	Local dataset	YOLO-Darknet19	AUC=90.8%
Gooben [27]	2019	Local dataset	Ensemble(CNN + FCN + MIL)	AUC= 96%
Xiang [28]	2019	Local dataset	CNN+SVM	Acc=94%
Tony [29]	2018	Local dataset	CNN (VGG-16)	Acc= 99.78%
Geva [30]	2015	Local dataset	Textual features + KNN	Sens= 81% Spec=87%
Subrato [31]	2020	RS-NIH NIH	VDSNet (VGG+ Data-Augmentation + STN)	Acc= 70.8% Acc= 73%
Xing [32]	2019	NIH	Generative Adversarial Network	AUC=84.1%

Qingji [33]	2018	NIH	AG-CNN (ResNet-50)	AUC=92.1%
Pranav [34]	2017	NIH	ChexNet (DenseNet-121)	AUC=88.8%
Imane [35]	2019	NIH CheXpert	CNN (DenseNet-121)	AUC= 92.9% AUC=84%
Mondal [36]	2019	RS-NIH	CNN (Xception)	AUC= 54.0%
Tae [37]	2018	NIH	Ensemble (3x ResNet-50)	AUC=91.1%

3.3 Segmentation:

Identification of location of pneumothorax is an important task for the automatic diagnosis of pathology. Several studies had been done in order to design an automated way of segmentation of pneumothorax area in a chest radiograph, however the main challenge is that the area of pathology is really small and there is no fixed shape. So segmentation task have imbalance problem in terms of CXR images as well as a CXR with pneumothorax contains very minimal amount of pixels in foreground as compared to the number of pixels in background. Appreciable work has been done by the researchers in this field which is briefly explained in this section.

In [38] a segmentation model built on U-Net architecture comprising of encoder and decoder part was proposed for identifying the location of pneumothorax. The experiments were performed on CT scans obtained from Medical University of Vienna. The model training was done on pixel-level and volume level. For the pixel-level segmentation method 43 Chest CT scans were used while 9 CT scans were used for evaluation and for volume-level method 567 CT scans were utilized. The model achieved mean Dice coefficient score (DSC) of 94% with 98% mean AUC score.

Gouting [39] presented a novel segmentation framework consisting of three modules. The first being a fully convolutional DenseNet, second being a spatial and channel squeeze and excitation module (scSE) and third being a multi-scale module. The proposed model was tested on

2213 CXRs obtained from their institution's image database. The model achieved Dice score of 92±0.14% and mean pixel accuracy (MPA) of 93±0.13%. The important contribution was proposing a weighted binary cross entropy loss function in order to cater the pixel imbalance problem.

In [40] concatenation operation in traditional U-Net was replaced with content-adaptive convolutional layers for reducing the number of parameters. The proposed model was trained and tested on SIIM Pneumothorax dataset containing total of 12047 CXR images and same number of masks. The images were down-sampled to 128x128 in this research. The pixel imbalance problem was solved by combining binary-cross entropy loss with soft dice loss. The proposed model achieved dice score of 76.04%.

Ostap [41] proposed a segmentation model containing three modules. Firstly Class activation maps (CAM) were generated using GradCAM++ method on trained ResNet-50 CNN model. These class activation maps were processed with Inter-pixel Resolution (IR-Net) in order to improve the class boundary between different class objects of the image. Finally the output from IR-Network was trained on U-Net model with ResNet-50 backbone. The proposed model was trained on 10,675 CXR images from SIIM pneumothorax dataset, while 145 and 541 CXRs with no-pathology and with-pathology were used for validation. The results were reported for test set containing 145 Pneumothorax samples and 541 Normal samples. The achieved dice score for this proposed model was 76.7%.

In [42] pneumothorax segmentation model was proposed by training images along with binary masks on a pre-trained U-Net model with ResNet backbone. The images were first preprocessed by means of resizing the data to 256x256 and contrast correction was performed. The training and testing was performed on SIIM pneumothorax dataset. The training was done with binary cross-entropy loss and model achieved dice score of 84.3% and IoU score of 82.6% on the test set.

In [43], an ensemble model was presented for segmentation of pneumothorax. The ensemble model comprised of three LinkNet networks with three different backbones including se-resnext50, SENET154 and se-resnext101. These encoders were used with ImageNet weights for training process. The training was done using Cosine Annealing Scheduling Learning rate for first 40 and last 15 epochs, while CyclicLR scheduling was used for 40th -65th epochs. For solving the class-imbalance problem, non-empty sampling was done in order to select a mini-batch in such

a way that it has at least one positive sample. The ensemble was created by union of binary prediction from each backbone model. The training and testing was performed using SIIM pneumothorax dataset and results were reported for stage1 and stage2 test sets separately. The model achieved 88.21% and 86.14% dice score on stage1 and stage2 test set respectively.

Syncho [44] presented a segmentation model for pneumothorax localization. Firstly preprocessing of images was done using Gaussian filter with Sigma equal to 0.5 in order to smoothen the images. Then fixed thresholding was applied in order to extract the thoracic region from the chest X-ray image. Morphological operations including erosion and dilation were performed on the threshold image in order to fill any holes in the image. After all this preprocessing, histogram was plotted in order to determine the threshold value to differentiate between lung tissues and the chest wall. In this way area of pneumothorax was identified. 137 CT scan images were used in this research and the proposed model performed with average error rate of only 1%.

Abedalla [45] proposed a 2 stage segmentation model for extracting the region of pathology (pneumothorax) from the images. The main architectural block was U-Net with ResNet34 encoder. Basically the model was first trained for 100 epochs with image size 256 using ImageNet weights as initializer and then these tuned weights were utilized for training the model with image size 512 for 70 epochs. Adam optimizer with learning rate 0.001 was used. The proposed scheme was tested on stage1 and stage 2 test set of SIIM Pneumothorax dataset and achieved dice score of 85.02% and 83.56% on stage1 and stage 2 test set respectively. Table 3.3 shows the existing work on pneumothorax segmentation.

Table 3.3: Literature Review on Segmentation

Author	Year	Dataset	Technique	Results
Rohrich [38]	2020	Local dataset	U-Net based (Own architecture)	DSC=94% AUC=98%
Gouting [39]	2019	Local dataset	Fully Convolutional DenseNet	DSC= 92±0.14% MPA=93±0.13%

Mostayed [40]	2019	SIIM dataset	U-Net (with content adaptive conv.layers)	DSC= 76.04%
Ostap [41]	2020	SIIM dataset	GradCAM + IR-Network + U-Net with ResNet50	DSC= 76.7%
Jakhar [42]	2019	SIIM dataset	U-Net (with pre-trained ResNet encoder)	DSC= 84.3% IoU= 82.6%
Groza [43]	2020	SIIM dataset	LinkNet ensemble (se-resnext50, SENet-154, se-resnext101)	DSC= 88.21%
Abedalla [45]	2020	SIIM dataset	2ST U-Net (2 stage U-Net with ResNet34 as encoder)	DSC=85.02%

3.4 Research Gaps

Some of the research gaps in the existing literature are summarized below:

- Many researchers used private dataset for classification purpose because of which further research in that work can't be carried out.
- From the literature, it can't be inferred that which of the class-imbalance solving technique is best for the CXRs dataset, as these techniques have never been compared with a CXR images dataset.
- Two major issues were found in the papers using openly available datasets like NIH Chest Xray-14 and CheXpert:
    - Most of the papers used Random split of dataset in which there are chances of patient-wise overlap in training and testing sets.
    - The papers in which patient-wise split was done, mostly have used random patient-wise split instead of using the official split provided by NIH and CheXpert, while competitive results can only be claimed while using a standard test set.

    - Some of the researchers didn't explicitly explained the technique followed to solve the class imbalance issue which is present in both these CXRs dataset.
- The datasets like NIH and CheXpert contains samples with multiple thoracic diseases, i.e. these are not dedicated to a single diseases, so the imbalance ratio between the CXRs with any pathology to the CXRs with No pathology is really high. For the NIH dataset there are only 5298 CXRs with pneumothorax while 60,412 samples with No finding. So the imbalance ratio is 1:11, i.e. for every 11 samples with No finding, there is just 1 sample with pneumothorax. On the other hand, in case of CheXpert dataset, although the imbalance ratio between samples with pneumothorax and samples with No-pathology is not very large, however the problem with CheXpert dataset is that there is a large number of CXRs with uncertainty-labels (i.e. the labeler of the CheXpert couldn't properly interpret the pathology from the radiology report, hence the presence or absence of a pathology for which uncertainty label is assigned is just 50-50%.). So because of this Uncertainty label presence, the results of the researchers may vary depending on the technique followed to solve this issue.
- In contrast, SIIM Pneumothorax dataset is the first openly available CXR dataset dedicated just for a single diseases, i.e. pneumothorax. Moreover the imbalance ratio for pneumothorax and No-pathology CXRs is relatively low (i.e. 3:1), so it will give more robust results for detection of pathology. In addition this dataset had never been explored for Classification purpose.
- The most important point is that NIH and CheXpert were proposed with the aim to detect the presence of pathology. Identifying the location of pathology was not the main goal. So only image level labels were provided in these two datasets for the 14 common thoracic diseases. Although NIH provided bounding boxes for the localizing the pathology but that too, were only for limited number of CXRs. However it is well understood that detection along with localization of area of pathology is more effective in medical field. So the plus point for the SIIM Pneumothorax dataset is that it not only provide image level labels, but also provide pixel-level label information (i.e. it provided RLE masks) for identifying the location of pathology, thus proposing a framework for classification and segmentation using this dataset yield more reliable results.

CHAPTER 4: METHODOLOGY

In this research, a classification and segmentation model is presented for the automated detection of pneumothorax from chest radiographs. We will first cover the classification technique followed by segmentation process.

4.1 Classification

As describes earlier, our proposed model not only detect the presence of pneumothorax but also solves the issue of class-imbalance which is troublesome in most of the classification problems. So a combination of model-level-ensemble [46] (i.e. ensemble of different CNN architectures used as feature extractor, with each architecture being fed with whole training data) and data-level-ensemble [20] (which is ensemble of different subsets of training data being fed to a single CNN architecture) is used in this research. The proposed model is a model-level-ensemble of three different data-level-ensembles. The CNN architectures used in this research for feature extraction purpose are VGG-16, DenseNet-121 and VGG-19 and, thus the model is named as VDV model. SVM is used as classifier. Figure 4.1 describes the basic work flow of the VDV model for predicting the class of any test image. The training and testing set were created using the data-split-lists provided in the original dataset link. The details of the CNN architectures and SVM classifier are given in this section.

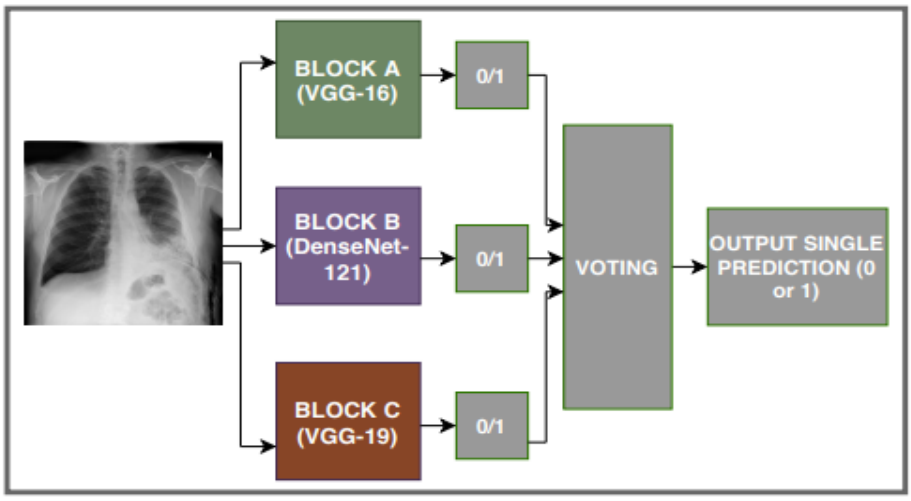


Figure 4.1: Basic Work flow of VDV model

Since we have first compared the existing approaches to tackle the class-imbalance issue for the classification model, so we will describe the experimental setup for comparison followed by explanation of VDV model.

4.1.1 Convolutional Neural Network

Convolutional neural networks are extremely popular in the field of Artificial intelligence. Because of their surprising ability to make unsupervised wise decision, deep learning has become one of the hot topics when it comes to utilization of AI in real life problems. The pioneer of CNN is Yann Lecon from New York who also work as director in Facebook's AI group. The structure of deep neural networks have resemblance with human neural system. Just like in human or mammal's neural system, deep learning architectures learn things by a layered structure. This way, artificial neural networks were designed in order to make right decision at right time without human supervision [47]. The working mechanism of CNN find its roots from the work principle of Artificial Neural Network (ANN). ANN is comprised of multiple perceptron (also known as neurons) at different layers. The input weights from one layer are sent to the next one but are never propagated back. So it is also called Feed Forward Neural Network. Moreover an ANN may or may not have hidden layer(s) but an input and output layer is a must. On the other hand CNNs have multiple hidden layers along with input and output layer containing several neurons. In CNN, the input and hidden layers are actually convolutional layers. The different layers can either be directly connected with each other or via pooling layer. The convolutional layers meant for extracting useful features of the image which is broken into several rectangles and then sent for non-linear processing. The structure of a typical CNN is shown in Figure 4.2.

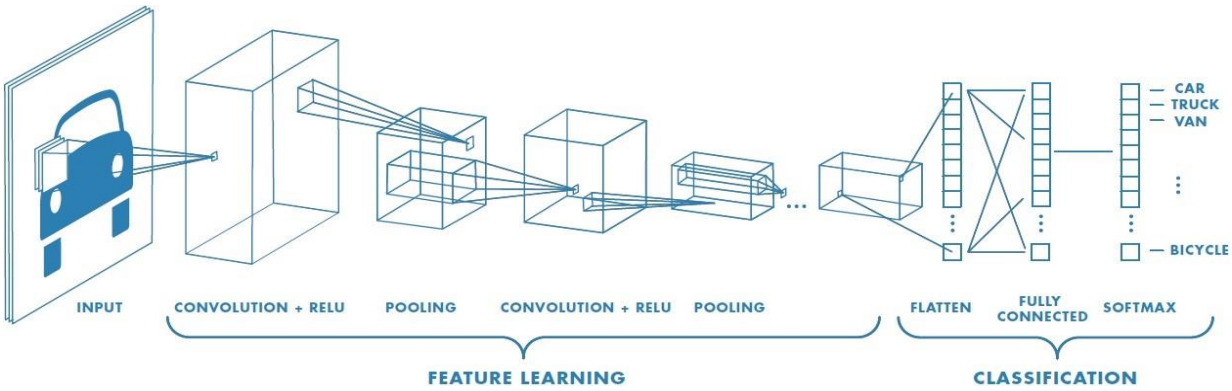


Figure 4.2: Typical CNN architecture. [48]

The detailed architecture and utilization of CNN are described in this section.

Architecture

A Convolutional Neural network is made up of input layer, hidden layers and an output layer. The hidden layer comprise of multiple convolutional layers, activation, pooling layers and fully connected layers (FC). The CNN consist of different number of convolutional, activation and pooling layers depending on the architecture. The structure up to the hidden layer is utilized for feature extraction while the last Fully Connected (FC) layers are used for classification [49]. The main building blocks of a typical CNN are described below:

- **Convolutional layers:**

  The convolutional layer consist of several independent kernels or filters and those filters are convolved separately with the input image. The convolution process is done in such a way that the kernel/filter slides over the image and the dot product between each corresponding image pixel and the kernel is take. Random initialization of filters is done and these filter values are updated based on the subsequent learning by the network. Initially the starting layers look for the basic patterns like lines or corner. As the network continues to train, the filters are actually taking dot product of the neurons of the previous layers with the respective weights.

- **ReLU (Rectified Linear Units) layer:**

  In CNN, it is convention to apply an activation layer or nonlinear layer immediately after the convolutional layer. The aim is to introduce non-linearity to the system that was performing only linear operations throughout the convolutional layers. ReLU basically speed up the training process of the network without causing a noticeable effect in the model performance. It also alleviate the vanishing gradient issue which occurs as the training process is very slow in lower layers of the network because the gradient decreases exponentially through the layers. The activation function used by ReLU is $f(x)=\max(0, x)$ which is applied to all values of the current layer outputs. In short, the ReLU convert the negative values to zero. This way it increases the amount of nonlinearity in the model without disturbing the receptive fields of the convolutional layer.

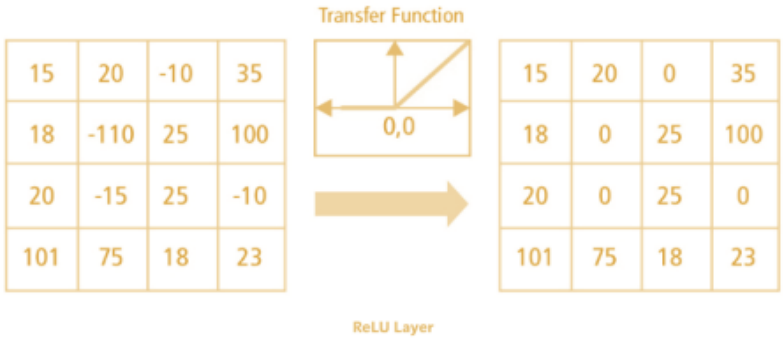


Figure 4.3: ReLU Operation [48].

- **Pooling Layer**

  Pooling layers are also knowns as down-sampling layers, and their function is to reduce the spatial resolution of representation and ultimately lessen the network computation by reducing the number of parameters. There are several nonlinear functions available for implementing pooling such as average pooling, max pooling and L2 norm. However most commonly used is max pooling. The working mechanism is that the image is divided into several non-overlapping chunks and the non-linear function selects the maximum value from every chunk. This way the spatial size of the input is reduced. The concept behind pooling function is that once the specific feature for a location is found that, then the exact location is not necessarily required. Thus there are two advantages of pooling. First one is that it reduces the computation by reducing the number of parameters. And secondly, it reduces overfitting. Figure 4.4 shows the example of Max and Average pooling.

- **Fully Connected**

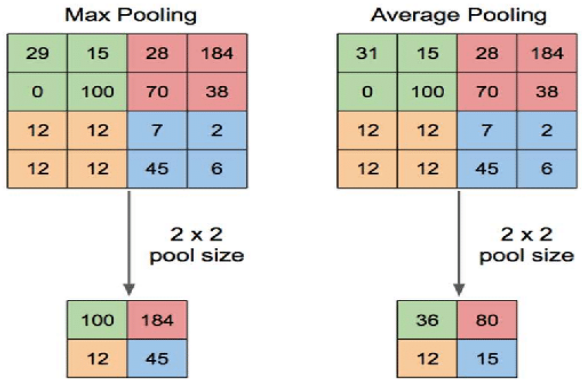


Figure 4.4: Max and Average pooling example [50]

In a typical CNN, after several convolutional, ReLU and pooling layers, Fully Connected (FC) layers are used for high level reasoning. The high level features in an image are extracted by the convolutional layers and those features can then be combined in a non-linear fashion using FC layers. The features from convolutional layers perform better when combined together by FC layer. The neurons in FC work on the mechanism same as normal artificial neural network (ANN) or multi-layer Perceptron (MLP).

CNN based methods

When using CNN for any classification problem, it can be done via two approaches shown in Figure 4.5. The first approach is known as "Learning from Scratch". In this approach the weights of the neurons are initialized with random values and the network is entirely dependent on the selected dataset for training. The other approach is known as Transfer Learning in which the trained parameters in terms of weights learned from some other dataset are used for initialization of network training for the selected dataset. Transfer learning can be applied in two ways. First one is Fixed Feature extraction [51] [52] and second one is Fine tuning. Following is the brief explanation for these two methods.

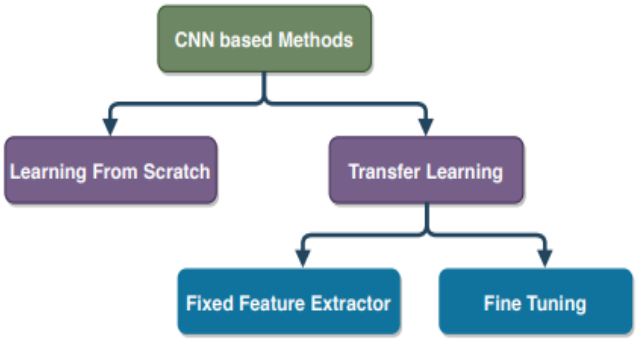


Figure 4.5: CNN based different Methods

- **Fixed Feature Extractor**

  Fixed feature extractor is a method of transfer learning in which the weights and bias learned from other big dataset are directly used for the classification problem of the specified dataset and there is no need to retrain the network on the selected/specified

dataset. So it means that only the Feature extractor part of the CNN architecture is being used and features extracted by this mechanism can then be classified by using any classifier like linear SVM or softmax classifier.

In our research we have used this approach of transfer learning and the features are send to SVM for training and classifying the chest radiographs as pneumothorax or normal CXR.

- **Fine tuning**

  Fine tuning is implemented by retraining a part of network on the specified dataset. This means that we freeze the weights of some of the initial layers and train only few layers on the task specific dataset. So for doing this, first of all, the number of channels/neurons in the last layer (i.e. output layer) is set to the total number of classes in the dataset. After this the weights and bias values of freeze layers of the network are set according to the pre-trained architecture. Then training parameters including total epochs, learning rate, batch size and optimizers etc. are defined and then training of the unfreeze layers is done in order to adjust the weights and bias values of those layers according to the task specific dataset.

4.1.2 Pre-trained Models

As mentioned earlier, we have used CNN pre-trained models are fixed feature extractor. The pre-trained models used in this research are VGG-16, DenseNet-121 and VGG-19. The detailed architecture of these models are explained in this section.

4.1.2.1 VGG-16

VGG-16 [53] is a convolutional neural network which was proposed back in 2014 by K.Simonyan and A.Zisserman in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition". The top test accuracy achieved by VGG-16 on ImageNet dataset was 92.7%. ImageNet is a large dataset containing around 14 million images which belong to 1000 different categories. This model is an improvement in AlexNet architecture and it replaces the large size of kernel (11x11 and 5x5 in the first two convolutional layers) by 3x3 multiple kernels.

The default input size of the image is 224x224x3, where 3 represents number of channels in the image. The image is passed through the convolutional blocks containing several

convolutional layer followed by maximum pooling layer. The maximum kernel size in any convolutional layer is 3x3. Each of these filters slides over the image and extract corresponding features. Even in one of the convolutional layer, the filter size is just 1x1 which actually is a linear transformation followed by non-linearity. The stride in case of convolutional layers is fixed to 1. The padding is set in such a way that after each convolution the spatial resolution is preserved, so the padding value is set to 1. It can be seen that max pooling layer is present after most but not all of the convolutional layers. The window size in case of max pooling is 2x2 with stride set to 2.

Followed by stack of convolutional blocks (convolutional layers + max. pooling), there are three Fully connected Layers (FC). There are 4096 channels/neurons in first two FC layers while third have number of channels equal to the number of classes in the dataset. In case of ILSVRC competition, as the dataset (ImageNet) has 1000 classes, so the number of channels in VGG-16's last FC layer was kept equal to 1000. It is to be noted that all the hidden layers (i.e. convolutional blocks) are equipped with ReLU, which is a non-linear activation function. This is used for reducing the computation [54]. The detailed architecture of VGG-16 is shown in Figure 4.6.

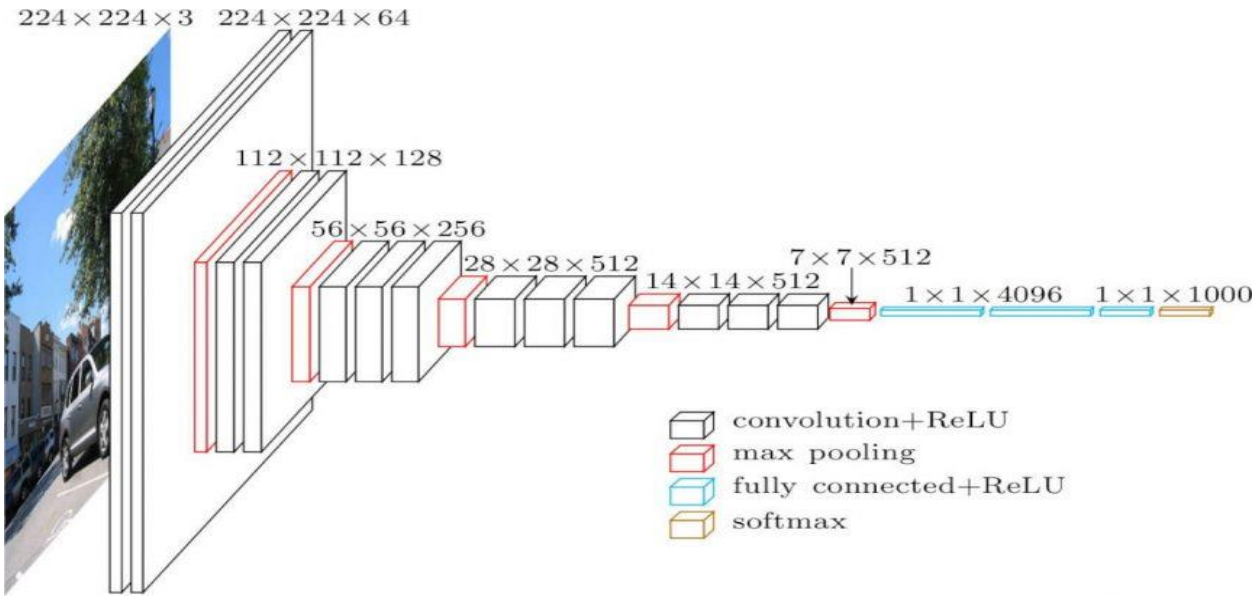


Figure 4.6: Detailed Architecture of VGG-16 [54]

4.1.2.2 VGG-19

The VGG-19 architecture is actually the extension of VGG-16, presented in the same paper [53] [55]. The only difference is that it contains 16 convolutional layers instead of twelve convolutional layers. Followed by the convolutional blocks are three FC layers. The number of channels in first two FC layers is 4096 just like in VGG-16. The last FC layer has 1000 neurons which is equal to the total number of categories in ImageNet dataset. Finally activation function (softmax) is applied which outputs the probability of each class separately. The basic structure of VGG-19 is shown in the Figure 4.7.

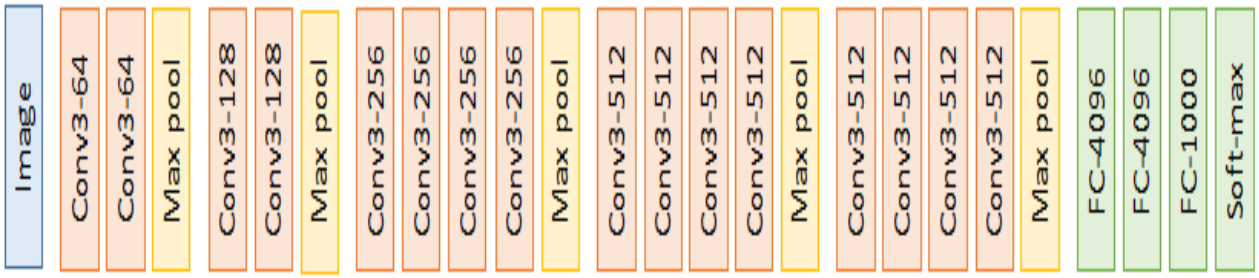


Figure 4.7: VGG-19 architecture [56]

Just like VGG-16, the default input size is 224x224x3. The image is passed to the stack of convolutional blocks, in which the window size of most of the convolutional layers is 3x3 with all the convolutional blocks followed by max pooling layer. The stride and padding in case of all convolutional layers is fixed to 1. The Max pooling is done using a window with 2x2 size with a stride of 2. Non-linear activation function ReLU is used in all the convolutional layers. The total number of parameters is 20,024,384 with input size kept as 224x224.

4.1.2.3 DenseNet-121

DenseNet (Dense Convolutional Network) is an architecture presented in 2016 by Goa Huang [57]. This architecture was presented based on the concept that adding inter-layer connections other than the direct connection between adjacent layers increases depth, and performance of the network. The DenseNet consist of dense blocks, in each dense block, every layer is connected to adjacent layer in a feed forward fashion as shown in Figure 4.8. Every dense

layer receives the feature maps from al the previous layers and concatenate them depth wise. This way, DenseNet can explicitly distinguish the information that is currently added to the network from the already present information. It results in increasing the discrepancy in the input of the subsequent layers and hence improve model performance.

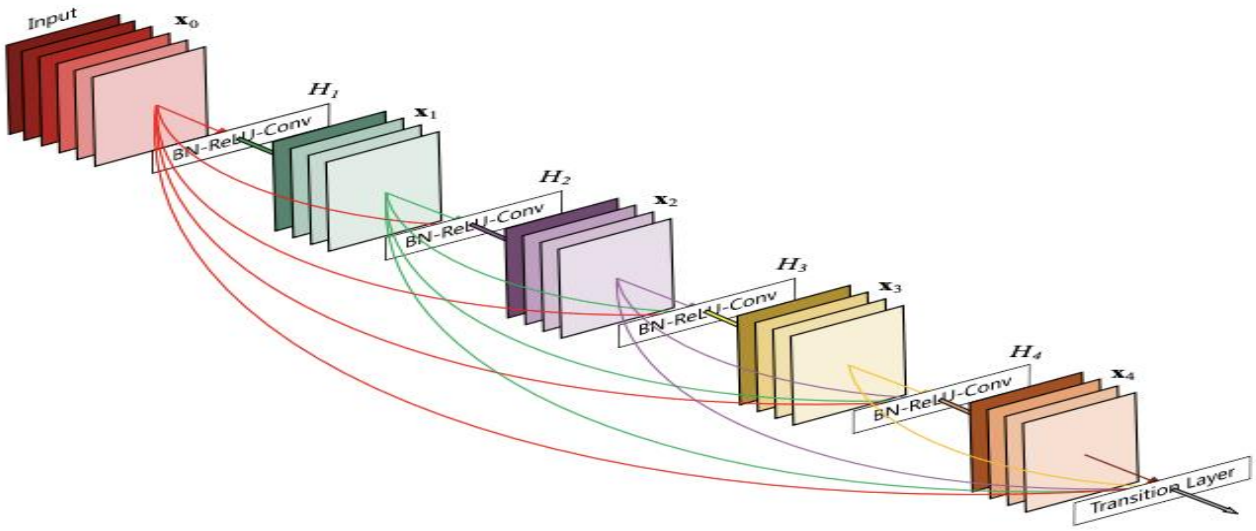


Figure 4.8: A 5-layer dense block with growth rate k=4

The main advantages of using dense connections are:

- Less number of parameters are required as compared to conventional CNNs and relearning redundant features from previous layers is not required here.
- Information and gradient flow is improved throughout the network and thus they are easy to train.
- Some regularization effect is present in dense layers which reduces overfitting when trained on smaller training dataset.

The typical DenseNet consist of: convolutional layers which are actually feature layers and captures low level features from the images, several dense blocks with transition layer between adjacent dense blocks and finally there is a classification layer. "Growth rate" is the depth of the output of dense block layer and it is denoted by k. Since the output from all the previous layers is received by every dense layer, the input depth for nth layer (D_n) is given in Eq. 4.1 as:

$$D_0 = k_0 + k * (n - 1) \quad (4.1)$$

where k_0 shows the number of channels present in the input or first layer. The name "growth rate" comes from the fact that although k output features are produced by each layer, however it has many more inputs. The depth of output keep on increasing linearly with adding more layers in the dense block. Let's assume that the initial growth rate is 40. If 100 more layers are added, then the depth will be around 4000. However this will exponentially increase the computations. To solve this problem, bottleneck layers (1x1 convolution) were used with the aim to reduce the depth of the feature maps. The bottleneck layer also ensures that the second convolutional layer has fixed depth of the input. It can be seen that throughout the dense layers, the size of feature maps remains fixed, which helps in stacking any number of dense layers in order to build a dense block.

In order to abstract higher level features, the output size of every layer decreases in a traditional CNN. In DenseNet, this task is performed by the intermediate transition layers, while the size and depth of the dense block remains the same. The transition layer is comprised of batch normalization layer (BN), convolutional layer with filter size 1x1 and average poling layer of size 2x2 with stride set as 2 in order to reduce the size to half. The depth is reduced by the 1x1 convolutional layer while the size (i.e. width and height) is reduced by average pooling layer. "Compression" was introduced in order to further decrease the depth by reducing the number of feature maps at the transition layers. If m feature maps are present at the output of the dense block then the adjacent transition layer will generate θ_m feature maps, where θ is the compression factor. The value of θ is within the range $0 < \theta < 1$.

The detailed architecture of DenseNet is presented in Table 4.1 with different variants of DenseNet which are 121, 169,201 and 161 layers for the ImageNet dataset. For our pneumothorax classification purpose, we used DenseNet-121 as fixed feature extractor.

Table 4.1: DenseNet architecture for ImageNet [57]. The sequence in each "conv" layer is BN-ReLU-Conv.

Layers	Output Size	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
Convolution	112 × 112	7 × 7 conv, stride 2			
Pooling	56 × 56	3 × 3 max pool, stride 2			
Dense Block (1)	56 × 56	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	56 × 56	1 × 1 conv			
	28 × 28	2 × 2 average pool, stride 2			
Dense Block (2)	28 × 28	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer (2)	28 × 28	1 × 1 conv			
	14 × 14	2 × 2 average pool, stride 2			
Dense Block (3)	14 × 14	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 64$
Transition Layer (3)	14 × 14	1 × 1 conv			
	7 × 7	2 × 2 average pool, stride 2			
Dense Block (4)	7 × 7	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1\times1 \text{ conv} \\ 3\times3 \text{ conv} \end{bmatrix} \times 48$
Classification Layer	1 × 1	7 × 7 global average pool			
		1000D fully-connected, softmax			

4.1.2.4 Support Vector Machines

A machine learning algorithm that can be utilized for the purpose of classification as well as regression. SVM classification is performed by finding hyperplane in such a way that maximizes the distance between two closest points of the different classes in the training set, known as support vectors [58].

Consider we have a training set belonging to two classes $(x_1, y_1), \ldots . (x_l, y_l)$ where $x_i \in R^n$, and $y^i \in \{-1, +1\}$.

The hyperplane is defined as $w.x = b = 0$. The support vectors are said to be separated optimally if the margin is maximum and error is minimum. The main parameters w and b for a canonical hyperplane have constraint: $min_{x_i}\, y_i\, (w.x + b\,) = 1$.

A separating hyperplane must satisfy the following condition

$$y_i\, [(w.x_i) + b] \geq 1, \qquad i = 1, \ldots \ldots \tag{4.2}$$

The distance of any data point x from the hyperplane can be found as:

$$d(w, b; x) = \frac{|w.x + b|}{||w||} \tag{4.3}$$

The margin of the hyperplane is $\frac{2}{||w||}$. So optimal hyperplane minimizes the following equation:

$$\phi(x) = \frac{1}{2} ||w||^2 \tag{4.4}$$

The solution to above equation is given by Lagrange equation, which has to be minimized with respect to b . Lagrange equation is given as :

$$L(w, b, \alpha) = \frac{1}{2} ||w||^2 - \sum_{i=1}^{l} \alpha_i \{y_i [(w.x_i) + b] - 1\} \tag{4.5}$$

Here α_i is the Lagrange multiplier. Solving the above equations lead us to the equations of $\bar{w}$ and $\bar{b}$ given as following:

$$\bar{w} = \sum_{i=1}^{l} \alpha_i y_i x_i \tag{4.6}$$

$$\bar{b} = -\frac{1}{2} \bar{w} [x_r + x_s] \tag{4.7}$$

where $x_r$and x_s denotes the support vectors, satisfying the following condition

$$\overline{\alpha_\iota}, \overline{\alpha_s} > 0 \, , y_r = 1 \, , y_s = -1 \tag{4.8}$$

So for now any new data point$x$, the classification can be done using following expression

$$f(x) = sign(w.\bar{x} + \bar{b}) \tag{4.9}$$

The above equation is used for linear separation , However for the case of non-linear separable data, the data point x of the input training set is mapped to point z in higher dimensional space, where $z = \phi(x)$. The important thing is that mapping $\phi(.)$ should be such that the dot product of feature space $\phi(x).\phi(y)$ can be written as Kernel function $K(x, y)$

The equation of decision surface is:

$$f(x) = \sum_{i=1}^{l} y_i \alpha_i \ K(x, x_i) + b \tag{4.10}$$

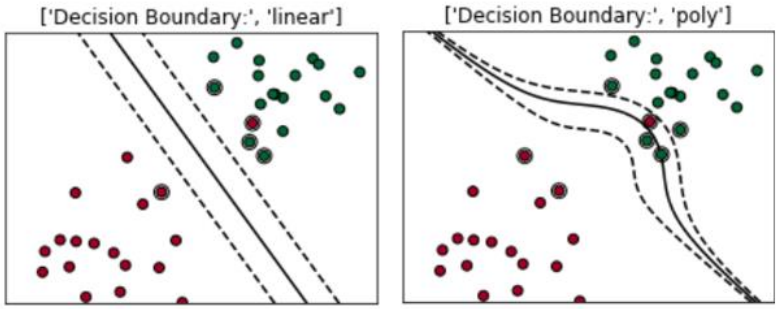
Here α_i and b are the solutions of the quadratic problem.

The kernel function used in our experiments are linear and polynomial kernel function which can be defined as follows:

$$\text{Linear kernel: } K\left(x_i, x_j\right) = x_i . x_j \tag{4.11}$$

$$\text{Polynomial kernel: } K\left(x_i, x_j\right) = (x_i . x_j + 1)^d \tag{4.12}$$

Here x_i and x_j represents the two different samples. Degree d in case of polynomial kernel can be adjusted according to the problem statement. The separating hyperplanes in case of a linear and polynomial kernel can be shown by a simple 2D graph as in Figure 4.9.



(a) (b)

Figure 4.9: Hyperplane examples for 2 class problem [59] (a) With Linear SVM. (b) With Polynomial Kernel SVM

4.1.3 Comparison of Class Imbalance Approaches

In our research we first compared the existing approaches to solve the class imbalance problem. For this part, we used the SIIM pneumothorax dataset. For all these approaches VGG-16 was used as fixed feature extractor and Linear SVM was used as classifier. The different approaches experimented are:

- **Weight-balancing**

  For experimenting the weight balancing approach, which is a classifier level technique, we used the whole training set as provided in the original dataset. 8296 Normal CXRs with 2379 Pneumothorax CXRs were used for training purpose. The weights were assigned according to the Equation 3.1.

- **Down-sampling**

  The dataset was down-sampled in order to equalize the number of samples in minority class (pneumothorax) and majority class (Normal). So out of 8296 Normal CXRs, only 2379 CXRs were used in combination with 2379 Pneumothorax CXRs.

- **Over-sampling**

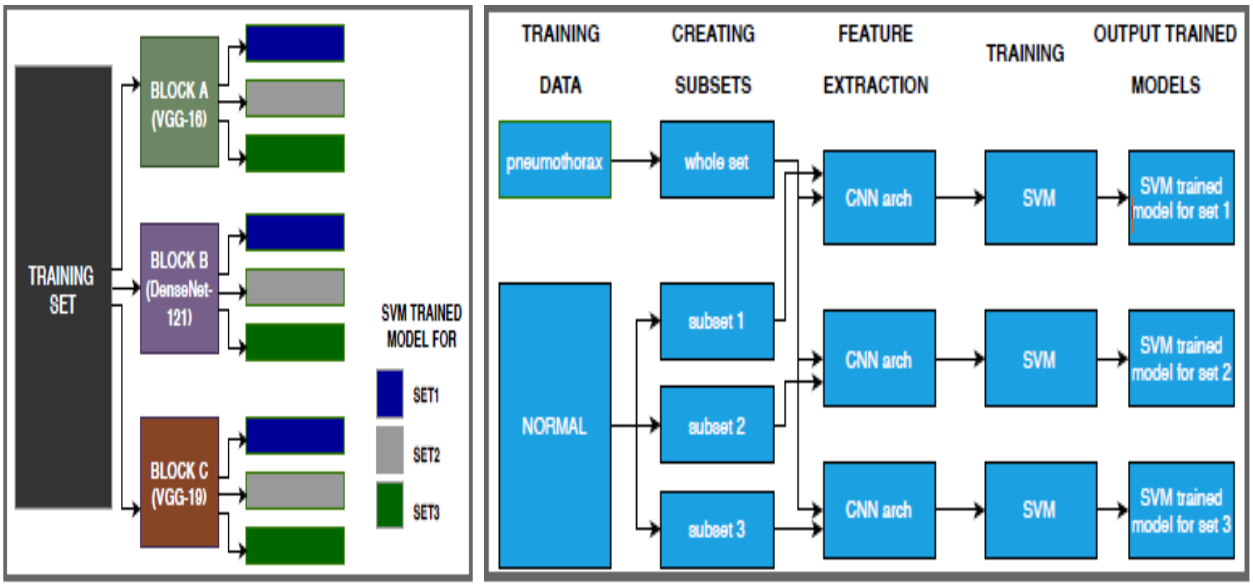
  Another data-level approach in which the aim is to make the number of samples equal in both classes, however here the number of minority class samples is increased. We used ImageAugmentation method in order to generate synthetic pneumothorax samples so we have 8296 Normal and 8296 Pneumothorax CXRs.

- **Ensemble**

  The data-level-ensemble was created by dividing the majority class into three subsets so that each subset has same number of samples as in the minority class. Each subset of training data was trained separately on the Linear SVM classifier. In test phase, the features of the test image were passed to every single trained model and prediction was obtained. Final prediction was obtained by combining the predictions from each single trained model using Voting method (i.e. maximum occurring class was selected as final output).

4.1.4 Proposed Model VDV

Among the multiple class imbalance solving approaches experimented in our research, data-level-ensemble was found to be performing best of all, so based on this and the concept of model-level-ensemble, we propose a VDV model which is a doubly ensemble based architecture. The VDV framework is a model level ensemble of three different data-level-ensembles. The different models utilized are VGG-16, DenseNet-121 and VGG-19. Thus the name VDV is given to the proposed framework. All these pre-trained CNN models were used as fixed feature extractor and polynomial kernel SVM (C=100, gamma=0.02 and kernel=poly) was used as classifier. Like any machine learning based CAD system, the VDV model also consist of training and testing module. The block diagram for the training module is shown in Figure 4.10.



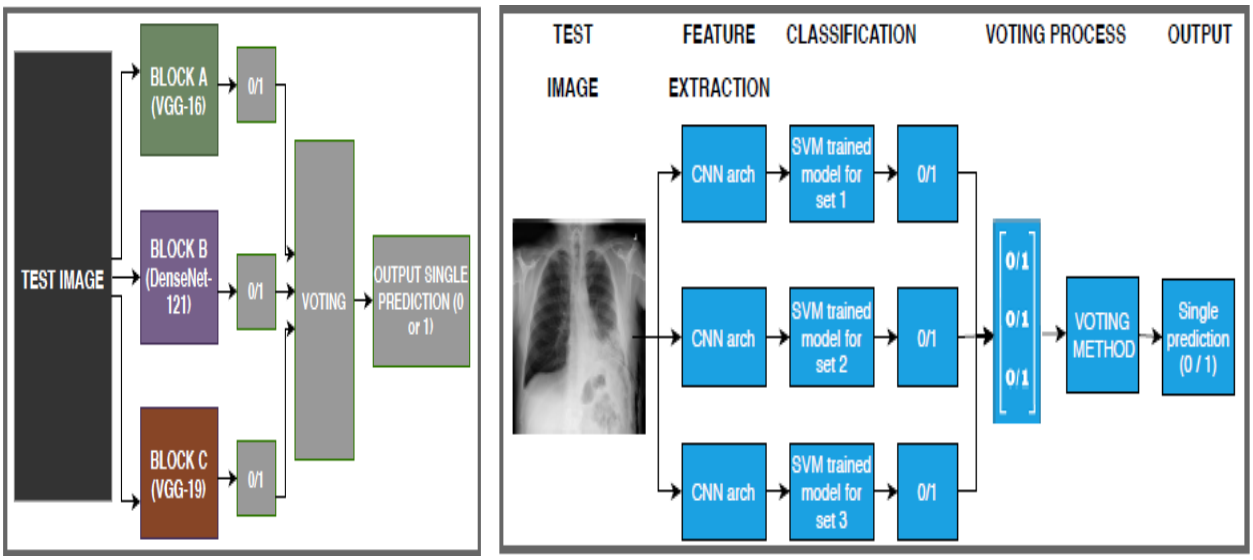
(a) (b)

Figure 4.10: Training Module.

The workflow of the training module is shown by a block diagram in Figure 4.10 (a). The training process utilized three different CNN architectures corresponding to each block. Block A refer to VGG-16, Block B refer to DenseNet-121 and Block C refer to VGG-19. Training set was sent to each block separately which generated three trained SVM models each. These trained SVM models were used for predicting the test sample class. The internal working of each block is same except for using different CNN architecture as feature extractor. Figure 4.10 (b) shows the generic working of each block. As our aim is to use whole training data in such a way that class-imbalance problem is solved as well, so we first created data-level ensemble of the training data. This was done by creating subsets of training data in such a way that each subset was class balanced, i.e. each subset had equal number of samples from both the classes. As the Normal CXRS (i.e. class 0) are in majority so they were divided in such a way that each subset had n number of normal CXRs where n is the number of samples in minority class (i.e. class 1 which is pneumothorax class). Thus N subsets of class 0 are created where N is equal to the class imbalance ration between the two classes. As the imbalance ratio is SIIM dataset is 3.4:1 so 3 subsets of Normal class samples were created. When subsets of Normal CXRs were created, they were combined together with whole minority class in order to make mini-training sets (referred as set1, set2 and set3 in the

Figure 4.10). Features were extracted from each mini-training set using the CNN architecture corresponding to each block and these extracted features were then trained using Kernel SVM.

The block diagram for the test module is shown in Figure 4.11 (a). The trained SVM models generated during training process were utilized here. For making class prediction for any test samples, it was sent to each block separately, which generated the class prediction. As we have three blocks, so three predictions were obtained. These class predictions were combined together by Voting Method, i.e. prediction with maximum frequency was selected as final output. In Figure 4.11 (b), the internal generic working of each block for the test module is shown. The test sample was sent to each block in which features were extracted from the CNN architecture with respect to each block. These extracted features were sent to the three trained SVM models (corresponding to set1, set2 and set3 of the training data). Each SVM model generated a prediction and these predictions were concatenated via Voting Method. Note that this process was repeated for each Block separately.



(b) (b)

Figure 4.11: Testing Module.

In our proposed model the input size for each of the Block (A, B and C) was kept as 224x224. For VGG-16 and VGG-19 all of the extracted features (i.e. 25088) were used for training kernel SVM, however for DenseNet-121 the number of extracted features was 50176 which is

huge number and difficult to train on SVM with 4758 samples, so the number of features was reduced using Principal Component Analysis (PCA), the number 50176 was reduced to 4758 and this number was obtained by using Singular value decomposition (svd) solver method [60]. Moreover instead of using 8296 samples we used 7137 samples of Normal CXRs in ensemble model, i.e. 3 times more than the number of samples in pneumothorax class. This was done in order to make the training subsets class balanced.

For authentication of the proposed model we also evaluated the performance on publicly available Random Sample of NIH chest X-ray dataset, from which we selected normal and pneumothorax samples. The imbalance ratio between normal and pneumothorax samples is 11:1. So the only difference while performing experiments with this dataset was that here we made 11 subsets of the training data instead of 3 subsets. The rest of the implementation was same just as describe above.

4.2 Segmentation

The purpose of segmentation is to extract the region of interest from the image. In our case, the region of interest is the area of the lung affected by the pathology (pneumothorax). The size of the affected area may vary from patient to patient. This problem comes under the category of sematic segmentation which is sub-category of Image segmentation [42]. Here every pixel of the image is considered as a particular class, either foreground or background. So every class is assigned a different color. As our segmentation problem is a binary problem so we have black and white mask. Instance segmentation is slightly different from the semantic segmentation as in the instance segmentation problem, different objects belonging to same class can also be differentiated as those are given separate colors. Semantic segmentation can be performed using a Fully Convolutional Network FCN

In this research, U-Net (FCN) was trained using the training images with the corresponding binary mask, in which white area represents the lung portion affected by the pathology while a whole black mask means that the corresponding X-ray image is of a healthy person. Figure 4.12 shows the flow chart for the proposed segmentation model. Training was done twice, firstly with image size 256 × 256 and then image size was increased to 384 × 384. The detailed architecture of a typical U-Net model and EfficientNet along with the proposed model are described in this section.

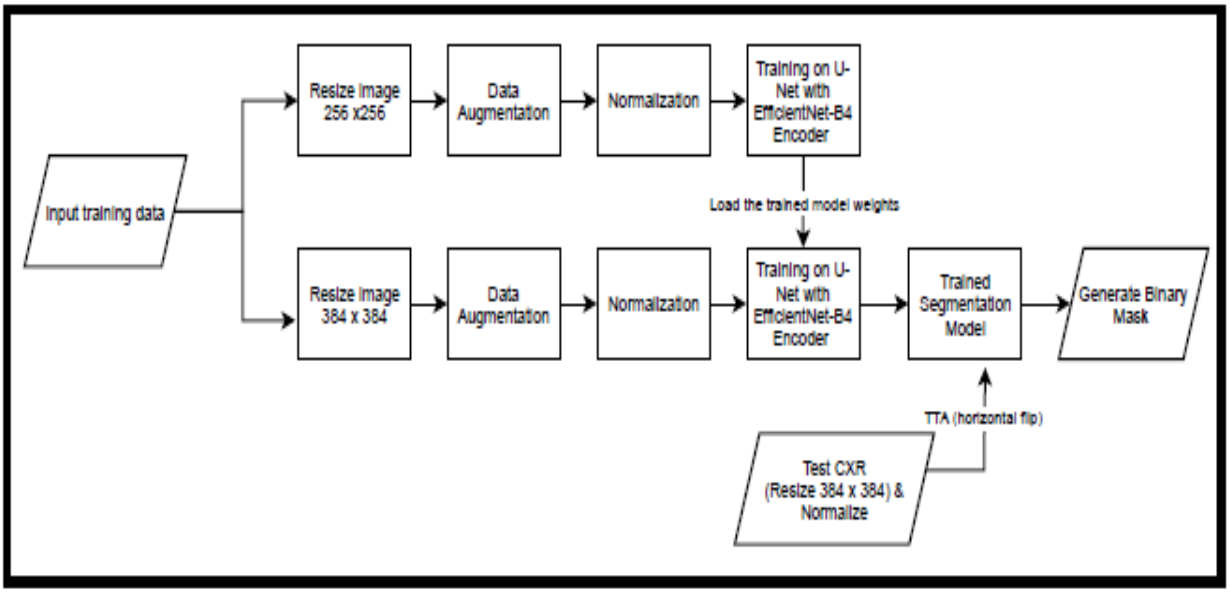


Figure 4.12: Flow Chart of Segmentation Methodology.

4.2.1 U-Net

U-Net is a popular architecture proposed by Olaf Ronneberger at al [61] for the purpose of Image Segmentation. It is a fully Convolutional Network in which all the layers are convolutional layers and there is no fully connected layer. U-Net is an Encoder-Decoder Network. Basically, spatial dimensions of the input image are gradually decreased by the Encoder (contracting path) while the depth of the image increases, this means that the Encoder part is trying to lean the information in the image however it loses the precise location of that particular information. On the other hand, the decoder (expansion path) gradually increases the spatial dimension to the original input's dimension and decreases the depth. In other words, it tries to recover the location of the information obtained by the encoder part.

In a typical U-Net architecture, the image input size can be reduced to minimum possible resolution of 32 x 32. Like any other FCN, U-Net consist of Encoder and Decoder part. The encoder consist of stack of convolutional layer and max pooling layer. A dropout layer can also be added after max pooling layer in order to avoid overfitting. Thus the encoder takes the image as input, followed by two convolutional layers with 3x3 kernel size and padding of 2. Each of the two convolutional layer is followed by ReLU activation layer, the image is down sampled into half by a max pooling layer with window size of 2x2. The number of convolutional filters is doubled

after every max pooling layer. As stated earlier that encoder is a stack of convolutional layers, so several convolutional layers are piled up with the aim of getting maximum information from the input image while the spatial dimension reduces and depth of feature map increases. The purpose of decoder is to recover the lost information about the location and to recover the size of the output to original spatial dimension. This is done by up sampling the feature map by using Transpose Convolutional layer. After up sampling Convolution is performed with a kernel size of 3 x 3 followed by activation layer. The number of convolutional filters is reduced to half after every up sampling process. This process is repeated till the original spatial dimension of the input image is recovered. The model of a typical U-Net architecture is shown in the Figure 4.13.

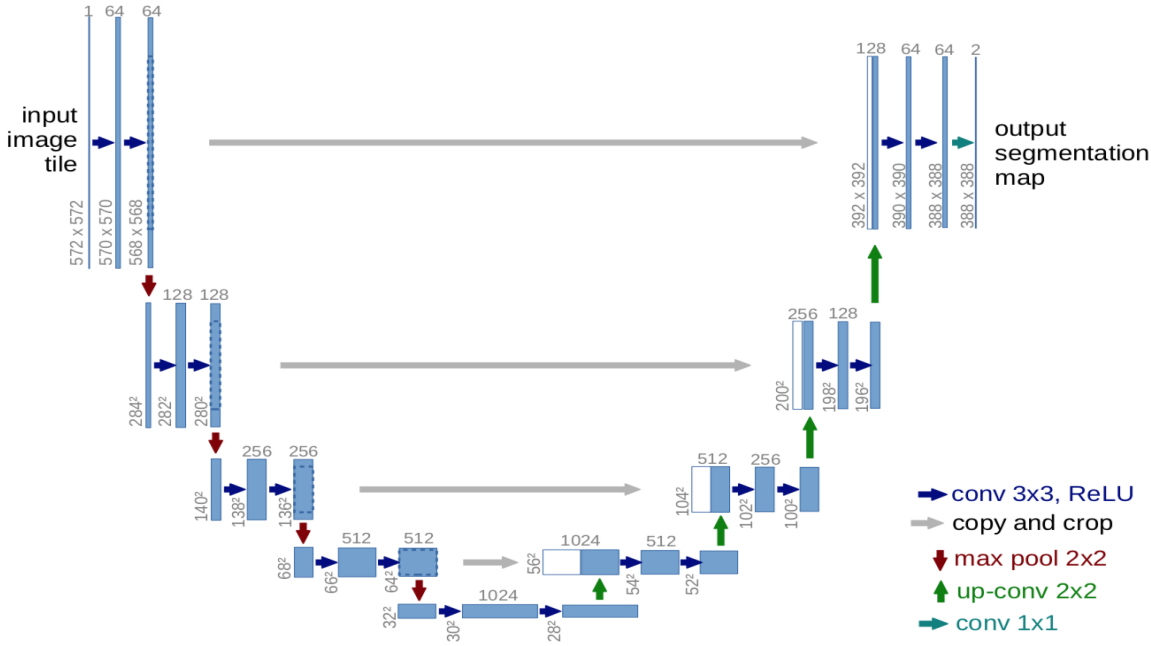


Figure 4.13: A typical U-Net architecture [61]. The blue rectangular boxes shows the multi-dimensional feature maps. The x-y dimensions of each box is shown on the lower left edge while the number of filters of convolutional layers is shown at the top of each box. The white boxes shows the copied corresponding feature maps.

4.2.2 EfficientNet

With the evolution of Convolutional Neural Networks and the fact that Alex-Net performed best of all in 2012 ImageNet Challenge, several different architectures of CNNs have been

presented which mainly differ in terms of number of layers and intermittent connections. However one of the main issues in designing neural networks is model scaling, i.e. to decide the criteria for increasing model size in order to get better performance [62].

So the designing of a CNN is basically a hit and trial method until required results are obtained, this is both resource and time consuming. So in order to solve this issue, a paper was released by Google in 2019 that introduced new family of convolutional neural networks known as EfficientNet [63]. These models not only perform better in terms of accuracy but also reduces the number of trainable parameters and floating point operations per second (FLOPS) thus enhancing the efficiency of the models. In the said paper, a mobile-sized baseline architecture EfficientNetB0 was introduced. Also a "compound scaling" method was introduced as a rule of thumb to increase the model size for maximizing the model performance.

The "Compound Scaling" method can be used for the existing CNN architectures as well, like Mobile Net and ResNet. However the choice of baseline architecture is an important decision as the compound scaling method can only increase the predictive capacity by replicating the convolutional operations of the base network.

The EfficieintNet-B0 proposed in the paper achieved 77.3% accuracy with only 5.3M parameters and 0.39B FLOPS while ResNet achieved 76% with 26M parameters and 4.1 FLOPS. The architecture of EfficientNet B0 is shown in Table 4.2. Each row represents stage i with Layers L_i , and input resolution of H_i, W_i and C_i output channels.

Table 4.2: EfficientNet-B0 architecture

Stage i	Operator $\hat{\mathcal{F}}_i$	Resolution $\hat{H}_i \times \hat{W}_i$	#Channels $\hat{C}_i$	#Layers $\hat{L}_i$
1	Conv3x3	224×224	32	1
2	MBConv1, k3x3	112×112	16	1
3	MBConv6, k3x3	112×112	24	2
4	MBConv6, k5x5	56×56	40	2
5	MBConv6, k3x3	28×28	80	3
6	MBConv6, k5x5	14×14	112	3
7	MBConv6, k5x5	14×14	192	4
8	MBConv6, k3x3	7×7	320	1
9	Conv1x1 & Pooling & FC	7×7	1280	1

The main building block of EfficientNet is MBConv which is similar to the inverted residual blocks used in MobileNetV2, along with adding squeeze-and-excitation optimization, as

shown in Figure 4.14. These MBConv actually forms a shortcut connection between beginning and end of convolutional blocks. Initially the depth of feature maps is increased by expanding the input activation maps using 1x1 convolutions. Then reduction of number of channels in the output feature map is done by Depth wise and then Point wise convolution with 3x 3 filters. The narrow layers are connected by a shortcut connection while the wider layers are located between the skip connections. Thus, the presented structure minimizes the number of operations and the size of the model without compromising on model performance.

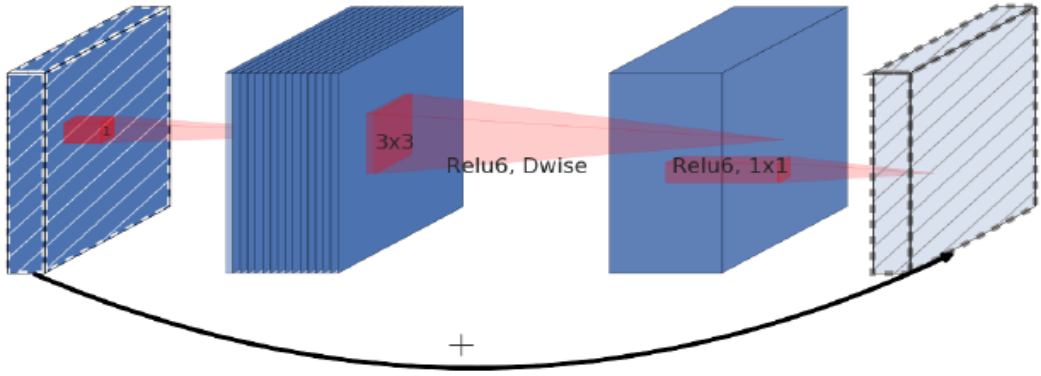


Figure 4.14: MBConv-Inverted Residual block [62]

4.2.2.1 Compound Scaling

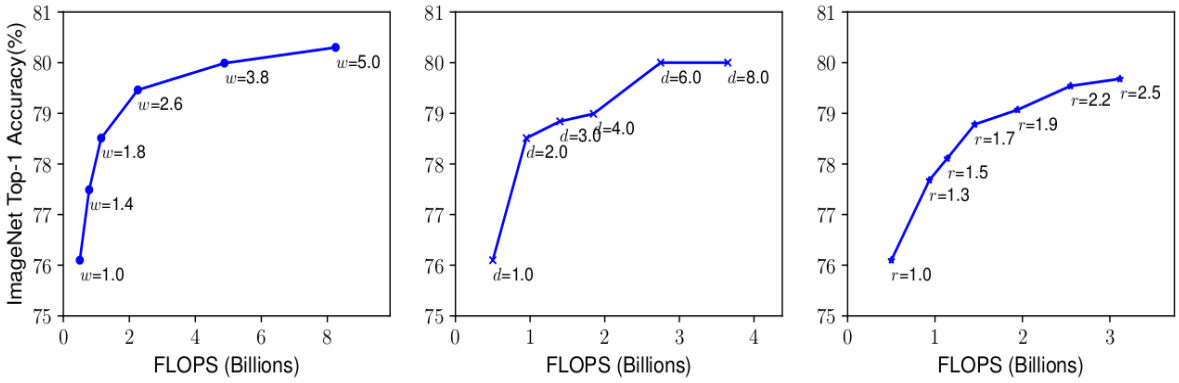
Scaling or increasing the size of the model can be done in three dimensions: depth, width and resolution. The depth of the network is associated with the number of layers, width correspond to the number of neurons in a layer or more precisely, the number of filters in the convolutional layer. The resolution is nothing but the width and height of the input image.

Increasing the depth, i.e. stacking more convolutional layers enable the network to learn complex features however deeper the network, more chances of gradient vanishing problem and so it becomes difficult to train the model properly. Although this problem is resolved by using skip connections and batch normalization however such deeper networks saturate quickly. For example, Resnet100 and Resnet1000 gives the same performance although the later one has more number of layers.

Model scaling by increasing width of the network allows the network to learn fine grained features. Model scaling this way, has been utilized in Wide ResNet and MobileNet, however

increasing only width disables the network to learn complex features, thus compromising on the performance

Increasing the model input resolution increases the model performance by providing greater details about the image however this technique has its own limitations in terms of performance gain. Figure 4.15 shows the comparison of performance on ImageNet dataset while adopting different techniques of model scaling. In the right most figure, r=1.0 shows the input resolution of 224 x 224 while r=2.5 denotes the input size of 560 x 560.



(a) (b) (c)

Figure 4.15: Baseline Model Scaling with different approaches [62]. (a) Width. (b) Depth. (c) Resolution.

Thus it can be concluded that increasing only one dimension increases accuracy however this effect is diminished for the bigger models. So it can be inferred that scaling should be done by combination of all these three dimensions. Now the important thing is to decide how to balance these three dimensions, i.e. how many new layers to add (depth increase), how many neurons to add in any layer (width wise increment) and what input size be finalized. For solving this issue, scaling method was proposed in the said paper which will be explained in following section.

4.2.2.2 Proposed Scaling Method

A CNN can be visualized as a stack of different convolutional layers. The layers can be partitioned into different stages and the layers in each stage has same type of convolutional layers. E.g. ResNet has 5 stages of layers present. So a CNN can be represented as:

$$N = \odot_{i=1\ldots s} F_i^{L_i} \left(X_{<H_i, W_i, C_i>}\right) \tag{4.13}$$

Here N represents the network, i denotes the stage number, F_i depicts the convolutional operation going on in i-th stage layers and L_i shows the frequency of F_i repeated in a particular stage. H_i , W_i and C_i shows the input shape and input channels. It can be observed that the depth of network is controlled by L_i , width by C_i and input resolution is controlled by H_i and W_i. Now finding an accurate set of coefficients to scale these parameters is impossible as it would require infinite computation. So the authors presented some ground rules to set these parameters.

- In a scaled model , all the layers will use the convolutional operation as present in the baseline network
- Scaling to be performed uniformly with a constant ratio for all the layers.

So equation can be parameterized as

$$N = \odot_{i=1\ldots s} \hat{F}_i^{d.\hat{L}_i} \left(X_{<r.\hat{H}_i, r.\hat{W}_i, r.\hat{C}_i>}\right) \tag{4.14}$$

where d, w, and r are the coefficients for scaling the model with respect to depth, width and input image resolution.

Effective scaling technique was proposed by the said authors using compound coefficient $\boldsymbol{\phi}$ for uniform scaling of depth width and resolution.

$$\begin{gathered} \text{depth: } d = \alpha^{\phi} \\ \text{width: } w = \beta^{\phi} \\ \text{resolution: } r = \gamma^{\phi} \\ \text{st } \alpha.\beta^2.\gamma^2 \approx 2, \quad \alpha \geq 1\ \beta \geq 1, \gamma \geq 1 \end{gathered} \tag{4.15}$$

$\boldsymbol{\phi}$ is a user defined variable which is a global scaling factor controlling the number of resources available, **α**, **β**, and **γ** decides how to assign the available resources to depth, width and resolution respectively. The FLOPS are directly proportional to w^2, r^2 and d, as doubling the depth will increase the FLOP two times, while FLOPS increase by four times while doubling the width and resolution.

In order to find parameters values (**α**, **β**, and **γ**) giving best results, grid search is used while setting **γ** equal 1. Once the parameters values are finalized, then ϕ can be increased to get larger and accurate model. This way EffiecientNet-B1 to EffiecientNet-B7 are formed with B indicating the compound coefficient value.

4.2.3 Proposed Segmentation Methodology

In this section, we will describe the model architecture for segmentation purpose in detail followed by description of training and testing module.

4.2.3.1 Model architecture

The main building block is a U-Net model which is used for segmentation purpose. As it contains encoder and decoder part, so for our experimentation, we created the encoder by removing the global average pooling layer and a fully connected layer from the end of the EffiicientNet-B4 architecture. The decoder is comprised of five blocks, each block consist of up-sampling layer with a filter size of 3x3, followed by a convolutional layer and two residual blocks with LeakyReLU (alpha=0.1) activation layer. The structure of a residual block is shown in the Figure 4.16.

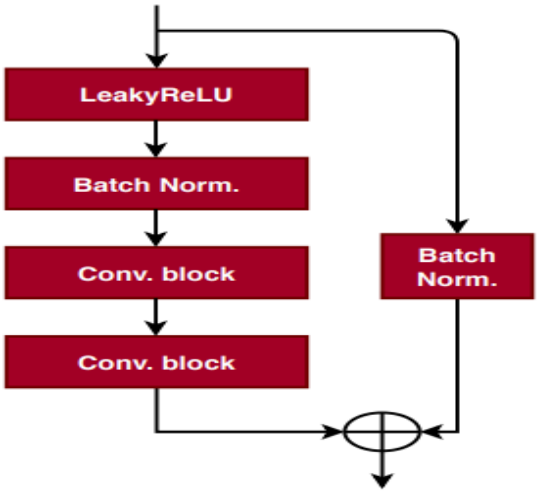


Figure 4.16: Residual Block

Inside Each residual block, firstly LeakyReLU activation is applied to the input layer and then batch normalization is performed. This processed layer is fed to a convolutional block twice and the output is added with the output of another layer in which batch normalization is applied to

the input layer of the block. The structure of convolutional block is shown in Figure 4.17. It consist of a convolutional layer with 3x3 filter size, batch normalization and LeakyReLU activation layer.

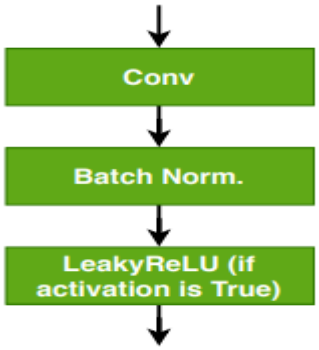


Figure 4.17: Convolutional Block

In the first four blocks of the decoder, concatenation is applied to the up sampled layer and the feature map of the same size from the encoder part. Lastly, after the fifth block convolution with 1x1 filter size is performed with sigmoid activation. At the end of the encoder and after each block of the decoder, a dropout layer is introduced to avoid overfitting. Basically dropout is a probability value at which output of the node/ layer is retained. The dropout value is set to 0.5. The detailed architecture is shown in Table 4.3.

Table 4.3: U-Net with EfficientNetB4 Encoder

Layer name	Layer detail	Encoder/Decoder
Input layer	Input to EfficintNet-B4 model	Encoder
Middle block	• Convolutional layer • Residual block × 2 • Leaky ReLU	---
Generic Decoder block	• Up-sampling (convolution Transpose) • Concatenate Output from layer number **N** of EfficientNet-B4 • Dropout • Convolutional layer • Residual block × 2 • Leaky ReLU	
Decoder 1	Generic decoder block with N=342	Decoder

Decoder 2	Generic decoder block with N=154	Decoder
Decoder 3	Generic decoder block with N=92	Decoder
Decoder 4	Generic decoder block with N=30	Decoder
Decoder 5	• Up-sampling (convolution Transpose) • Dropout • Convolutional layer • Residual block × 2 • Leaky ReLU	Decoder
Output layer	Convolution of Decoder 5 output	Output

4.2.3.2 Training module

The proposed framework for segmentation finds its root from [45]. The main building block is U-Net architecture with EfficientNet-B4 encoder which is described in detail in the above section in Table 4.3. The framework consist of 2 stages. The detailed block diagram for training module is shown in Figure 4.18.

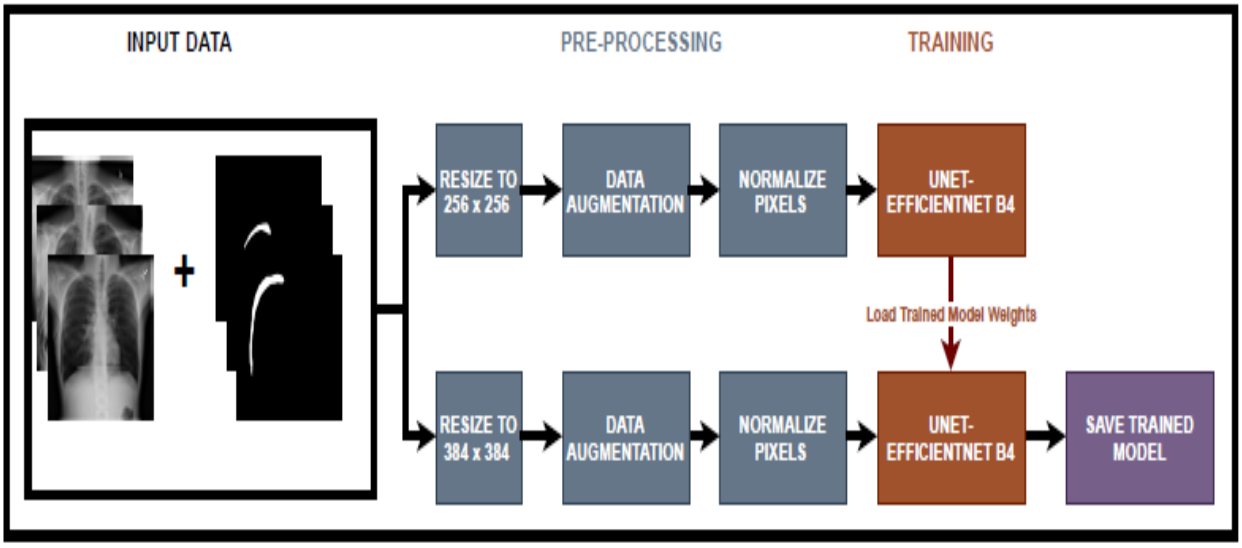


Figure 4.18: Segmentation Model - Training Module

The dataset contains images and corresponding binary mask in PNG format. Each image has resolution of 1024 x 1024. The images were resized to 256 x 256 and 384 x 384 for the 2 stages of training process. In the binary masks, the pixel value 0 shows the area without pneumothorax and the pixel value 1 shows the area with pneumothorax.

Data augmentation using the "albumentation library" was applied in both stages of the training module in order to reduce overfitting, since data augmentation increase the amount of training data and introduce diversity in the training data. Four groups of data augmentations were applied to images and binary mask including horizontal flip, one of random gamma, random contrast and random brightness, one of grid distortion, elastic transform and optical distortion, and random size crop. The pixel values of the masks were normalized so that the pixel values lies in the range of 0 to 1, by dividing each pixel value with 255.

In both the stage, Binary Cross Entropy and Dice loss were combined to make a combined loss function named as BCE-DICE-Loss. Cosine annealing scheduling was used to decay the learning rate with each epoch.

For the first stage, images and masks were resized to 256 x 256 and data augmentation was applied. The preprocessed images and masks were fed to the U-Net model with EfficientNet-B4 encoder and trained for 20 epochs. The initial weights were set to the ImageNet weights. Adam optimizer with initial learning rate 1e-3 was used.

For the second stage, images and were resized to 384 x384 and whole process of first stage was repeated except that here we trained the model for 25 epochs and the initial weights were set to the model's weights obtained from the first stage. Finally the trained model was saved for predicting binary masks of the test images.

4.2.3.3 Testing Module

For predicting the area of pneumothorax, i.e. generating binary mask for the test image, the images were resized to 384 x 384 resolution. Then test time augmentation (TTA) was done by applying horizontal flip to the test images. After that the average of predicted mask (without TTA) and predicted masks (with TTA) was considered as final predicted mask for any test image. The block diagram for the test module is shown in Figure 4.19.

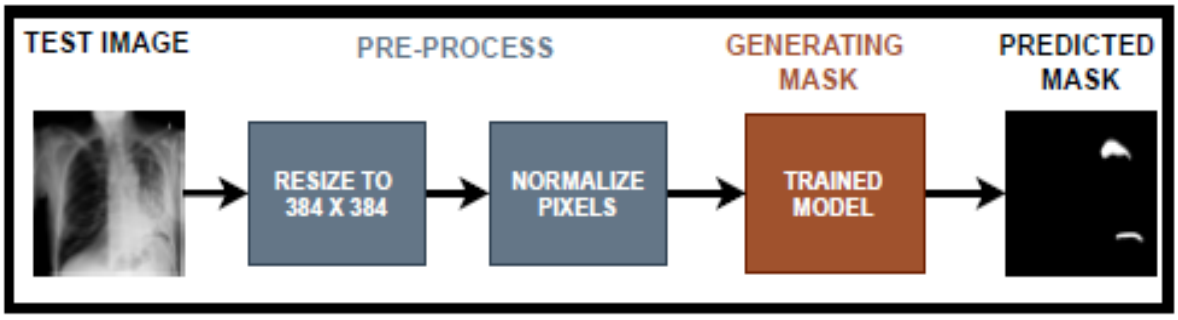


Figure 4.19: Segmentation Model - Testing Module

4.3 Combining Classification and Segmentation

After obtaining the results of classification and segmentation, we combined their results by means of class prediction. Since classification module give class prediction from the VDV framework. The proposed two stage segmentation framework generates binary mask. Now in the generated mask, if a mask has white area, it shows the presence of pneumothorax.

We performed a different kind of experimentation by predicting class of an image from the generated mask. For this purpose, we counted the number of white pixels in every generated mask. If a mask has even a single white pixel, we say that the mask belongs to an image with pneumothorax present. This way we had class predictions for all the images using the segmentation results. The flow chart for obtaining class prediction from the predicted mask is shown in Figure 4.20.

Once we got the class prediction, we combined these prediction with those generated by the classification VDV framework by two different methods, i.e. Union and Intersection

- Union: For any image, if the predicted class by classification OR segmentation module is “1”, i.e. pneumothorax, then we say that the image belong to Pneumothorax Class.
- Intersection: For any image, if the predicted class by classification AND segmentation module is “1”, i.e. pneumothorax, then we say that the image belong to Pneumothorax Class, else it belong to Normal Class.

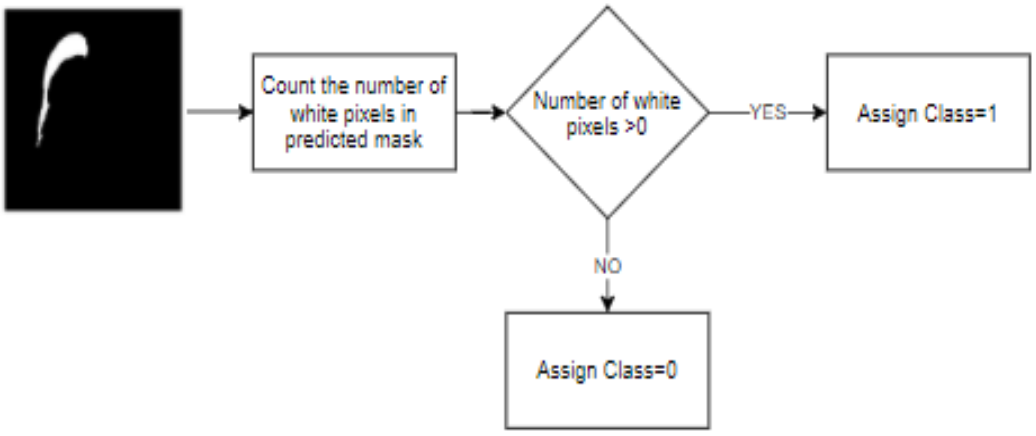


Figure 4.20: Flow chart for predicting class from the generated mask.

CHAPTER 5: EXPERIMENTAL RESULTS

5.1 Databases

The proposed methodology in this research is evaluate on two openly available datasets, namely SIIM ACR Pneumothorax dataset (SIIM) and Random sample of NIH Chest X-ray dataset (RS-NIH). The details of these datasets are given in this section.

5.1.1 SIIM ACR Pneumothorax dataset:

SIIM ACR Pneumothorax Segmentation dataset is gathered by Society for Imagining Informatics in Medicine (SIIM) with the aim to contribute towards automated diagnosis of pneumothorax from the chest radiographs. A competition was held on Kaggle in order to develop a robust segmentation model using the training and testing set provided in this dataset. The dataset provides 12047 chest X-ray images and corresponding masks in Run Length Encoding (RLE) format, divided into training and testing set. This dataset is also available with masks converted into Portable Network Graphics (PNG) format. In our research we have utilized the same dataset available on Kaggle [64] with CXR images and binary masks in .PNG format, and stage-1 testing set. Each image and mask has resolution of 1024×1024 with 8 bits depth per image. The details of this dataset as used for experimentation is given in Table 5.1.

Table 5.1: Overview of SIIM dataset

Resolution		1024 x 1024
Dataset size		12047
No of classes		2
Training set	**N**	8296
	P	2379
Testing set	**N**	1082
	P	290

N: Normal CXRs, P: Pneumothorax CXRs

5.1.2 Random Sample NIH Chest X-ray dataset

The Random Sample NIH Chest X-ray (RS-NIH) dataset is sample version of full NIH Chest X-ray-14 (NIH-CXR) dataset provided by NIH itself and also available on Kaggle [65]. This dataset is 5% of the NIH-CXR dataset. Just like the full NIH Chest X-ray-14 dataset, the RS-NIH contains total of 15 class, 14 classes representing different thoracic diseases and 15th being the Normal class. The resolution of each image is 1024×1024. The sample dataset contains 3044 images of No-finding, Infiltration:967, Effusion:664, Atelectasis:508, Nodule:313, Mass:284, Pneumothorax:271, Consolidation:226, Pleural Thickening:176, Cardiomegaly:141, Emphysema:127, Edema:118, Fibrosis:84, Pneumonia:62 and 13 images of Hernia. The dataset is divided into 80% training and 20% testing data. The details of the dataset as used in this research are given in Table 5.2. Note that for RS-NIH dataset we performed the experiments twice. First with completely random split of dataset (as followed in most of the literature) and second with a patient-wise split of data (i.e. there was no overlap of patients in training and testing set).

Table 5.2: Overview of RS-NIH dataset

Resolution		1024 x 1024
Dataset size (with 14 classes)		5606
No of classes chosen		2
Training set	**N**	2379
	P	216
Testing set	**N**	609
	P	55

N: Normal CXRs, P: Pneumothorax CXRs

Some of the sample images from the afore-mentioned datasets are shown in Figure 5.1.

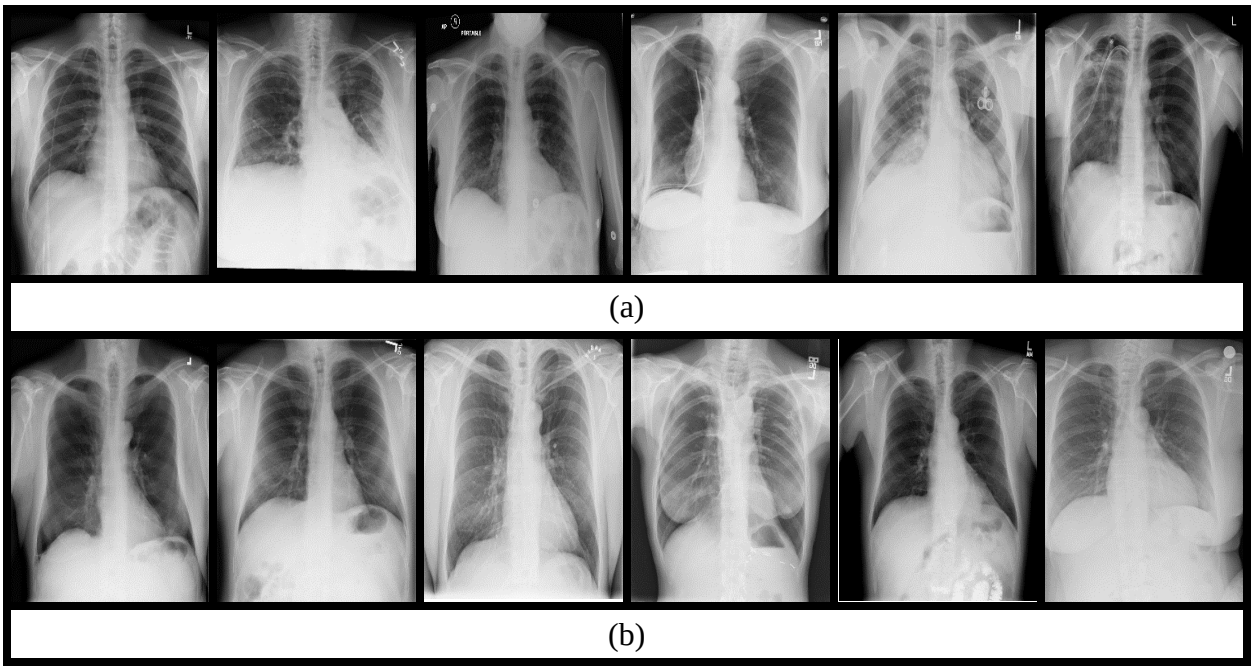


Figure 5.1: Samples from dataset. (a) SIIM ACR Pneumothorax. (b) RS-NIH

5.2 Performance Measures

In our research, as we have carried out two tasks, which are classification and segmentation. For classification the main performance measures are Recall (Sensitivity) and Area under the Receiver Operating Characteristic curve (AUC). Besides we have also reported the results in terms of other performance metrics including Accuracy, Specificity, Precision, F1 score, F2 score and G-mean.

In order to calculate AUC the area under ROC curve is calculate, where ROC is defined in terms of False positive rate (FPR) and True Positive Rate (TPR) [66]. F1 and F2 are calculated by using F_β expression in which β is assigned value depending on the problem statement. If the aim is to design a model which avoids classifying negative class sample as positive one, then β is given value 0.5. If importance is given to recall, i.e. aim is to never miss a positive class sample, then β is assigned value equal to 2. If both recall and precision are to be given equal importance then β

is set to 1 [31]. The expressions for calculating the above mentioned performance metrics are given below:

$$Accuracy = \frac{TN + TP}{TN + FP + FN + TP} \tag{5.1}$$

$$Recall = \frac{TP}{TP + FN} \tag{5.2}$$

$$Precision = \frac{TP}{TP + FP} \tag{5.3}$$

$$Specificity = \frac{TN}{TN + FP} \tag{5.4}$$

$$F_\beta = (1 + \beta^2)\ \frac{Recall \times Precision}{(\beta^2 . Precision) + Recall} \tag{5.5}$$

$$G\ mean = \sqrt{Recall \times specificity} \tag{5.5}$$

In all of these expressions True Positive (TP) and True Negative (TN) denotes the total number of correctly classified positive class samples and total correctly classified negative class samples respectively. False Positive (FP) represents the total number of negative class samples misclassified as positive ones, while False Negative (FN) is the number of positive class samples wrongly classified as negative class.

For segmentation, the performance measures used is Dice coefficient score (DSC) while during experimentation Intersection over union (IoU) score was used as performance measure. These are calculated as follows:

$$DSC = 2 \times \frac{Intersection\ \times Union}{Intersection + Union} \tag{5.6}$$

$$IoU = \frac{Intersection\ \times Union}{Intersection + Union} \tag{5.7}$$

In the above two equations, intersection is calculated by counting the total number of mutual foreground pixels in predicted and original binary masks and union is calculated by the total number of foreground pixels in predicted and original masks.

5.3 Results

As this research is comprised of three parts, i.e. comparing existing Class Imbalance approaches, designing a classification model and a segmentation model. The results for each part are describe separately in this section.

5.3.1 Class Imbalance approaches

The existing class imbalance approaches were compared using SIIM ACR Pneumothorax dataset. The total number of training and testing samples with respect to each class, utilized for each approach are shown in Table 5.3. For all these experiments, VGG-16 with ImageNet weights was used as fixed feature extractor and Linear SVM was used as classifier. The results show that a data-level-ensemble model outperforms other existing approaches, with highest AUC of 80.02% and Recall value equal to 79.65%.

Table 5.3: Comparison of existing Class-Imbalance approaches

Technique	**No of Training Samples**		**ACC (%)**	**REC (%)**	**SPE (%)**	**AUC (%)**
	Normal	**Pneumothorax**				
Weight balancing	8296	2379	79.08	48.96	87.15	78.8
Under-sampling	2379	2379	72.15	68.62	73.10	77.67
Over-sampling	8296	8296	77.7	50	85.20	77.76
Ensemble	2379 (in each subset)	2379 (in each subset)	75.22	**79.65**	74.09	**80.02**

ACC: Accuracy, REC: Recall, SPE: Specificity

5.3.2 Classification

In this research, the proposed classification model VDV was tested on SIIM ACR Pneumothorax dataset and Random Sample of NIH Chest X-ray dataset. The results of individual data-level-ensemble with each CNN architecture (i.e. VGG-16, VGG-19 and DenseNet-121) separately are reported for SIIM dataset only along with the performance of proposed VDV model on the same dataset in Table 5.4. The results prove that the proposed VDV model performs better than single CNN architecture in a data-level-ensemble in terms of all performance metric reported here. For RS-NIH dataset we have reported the performance of VDV model in Table 5.5.

Table 5.4: Performance of proposed model on SIIM dataset

	ACC (%)	**REC (%)**	**SPE (%)**	**PREC (%)**	**F1 (%)**	**F2 (%)**	**G-mean (%)**	**AUC (%)**
VGG-16	77.55	83.79	75.87	48.21	61.2	73.01	79.73	86±0.01
VGG-19	77.04	82.06	75.69	47.5	60.17	71.64	78.8	86± 0.01
DenseNet-121	76.32	80.68	75.04	46.4	58.94	70.31	77.81	85±0.00
Ensemble(VDV)	**78.27**	**85.17**	**76.43**	**49.2**	**62.37**	**74.3**	**80.68**	**86±0.00**

PREC: Precision, F1: F_1 score, F2: F_2 score.

Table 5.5: Performance of proposed model on RS-NIH dataset.

	ACC (%)	**REC (%)**	**SPE (%)**	**PREC (%)**	**F1 (%)**	**F2 (%)**	**G-mean (%)**	**AUC (%)**
RANDOM SPLIT OF DATA								
Ensemble (VDV)	82.68	**90.9**	81.93	31.25	46.5	65.78	86.3	**95 ±0.01**
PATIENT-WISE SPLIT OF DATA								
Ensemble (VDV)	69.12	**85.45**	67.65	19.26	31.4	50.64	76.03	**77 ±0.06**

The confusion matrix for the classification results on SIIM dataset and RS-NIH dataset are shown in Table 5.6 and 5.7 respectively

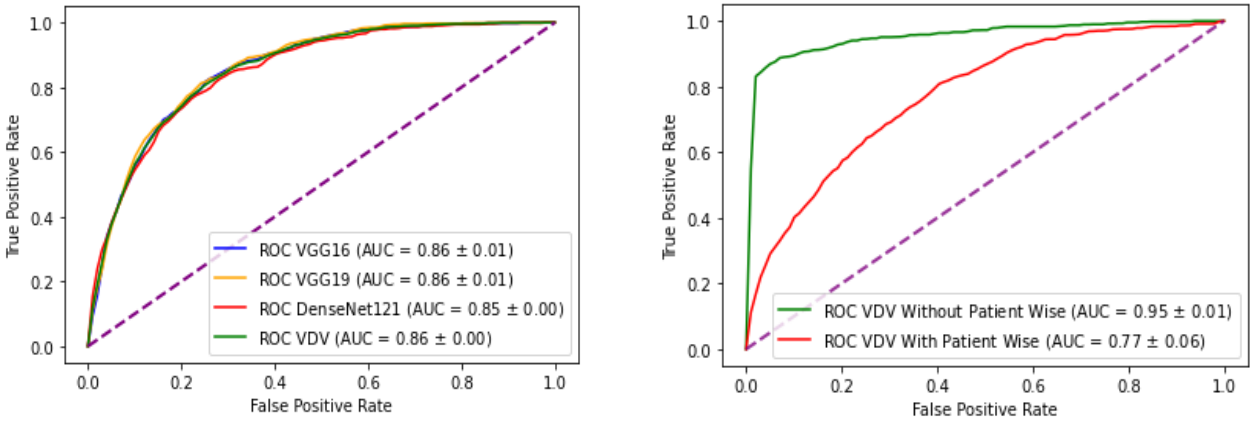
Table 5.6. Confusion Matrix for SIIM data

	Predicted Class	
Actual Class	Normal	Pneumothorax
Normal	827	255
Pneumothorax	43	247

Table 5.7. Confusion Matrix for RS-NIH data

	Predicted Class	
Actual class	Normal	Pneumothorax
Normal	499	110
Pneumothorax	05	50

The results in terms of ROC plots are shown in Figure 5.2. On left hand side image, the plots for individual architectures on a data-level ensemble are shown along with the ROC curve for the proposed VDV model on SIIM dataset. On right hand side image, ROC graph for VDV model on RS-NIH dataset is plotted for both type of data splits experimented in this research.



(a) **(b)**

Figure 5.2: AUC plot for proposed VDV model. a) On SIIM dataset. b) On RS-NIH dataset.

We have also presented the comparison of our results for both datasets with existing literature. In case of SIIM dataset, although a direct comparison can't be made as we are the first one to use to this dataset for classification however we have tried to compare our results with the already present work on pneumothorax detection in Table 5.8. In the table the total number of

class-wise samples in training and testing set for each research is mentioned. The sub-column B denotes that if the training or testing set is class balance in nature or not. It can be seen from the table that most of the work done has used local dataset and there is minimal ratio of imbalance. And if otherwise (i.e. data is imbalance in nature) then only single technique has been used instead of comparing multiple class imbalance techniques. Moreover in most of the cases, the training as well as testing data is small as compared to our SIIM dataset which is openly available, has high imbalance ratio and large number of samples in both training and testing set. So our proposed model performance surpass the existing work on pneumothorax detection with AUC of 86.0%.

Table 5.8: Comparison of existing work for SIIM dataset

Author	**Training Set**			**Testing Set**			**Description**	**Dataset public**	**Results (%)**
	N	**P**	**B**	**N**	**P**	**B**			
Yuan [24]	36	22	×	16	10	×	Ignored the minimal IR	×	ACC= 82.20
Yoon [25]	24	24	✓	15	15	✓	Completely balanced	×	ACC=96.60 REC=100 SPEC=93.8
S.Park [26]	10887	1343	×	250	253	✓	Down-sampling for class balance	×	REC=89.7 SPEC=96.4
Andrea [27]	350	453	×	87	113	×	Ignored the minimal IR	×	AUC=96.2
Xiang [28]	30	50	×	40	160	×	Ignored the IR	×	ACC=96.5 REC=100 SPEC=82.5
Andrew [67]	7095	2214	×	1553	437	×	Down-sampling for class balance	×	REC=55 PREC=90 AUC=82
Propose model (VDV)	**8296**	**2379**	×	**1082**	**290**	×	**Comparison of multiple approaches and ensemble was chosen based on performance**	✓	**ACC=78.27 REC=85.17 PREC=76.4 AUC=86.0**

B: Class Balance, IR: Imbalance Ratio

The performance of our proposed model on RS-NIH can be compared with existing work in which RS-NIH dataset has been used for multiple thoracic pathologies detection, as summarized in Table 5.9. We have referred only that paper in which RS-NIH dataset was used for multiple thoracic pathologies detection while the papers in which binary classification (without considering any specific pathology, i.e. normal and abnormal CXRS) was done were ignored while comparing the results. It can be clearly seen that the AUC achieved in [36] (for pneumothorax detection) is only 54% which is very less as compared to our achieved results with both types of data split.

Table 5.9: Comparison of existing work for RS-NIH dataset

Author	Data-set	Description	Results (%)
Modal[36]	RS-NIH	Multi-label classification of 14 thoracic diseases	AUC= 54.0
Random split of data			
Proposed model	RS-NIH	Binary classification (normal and pneumothorax CXRs)	**ACC=82.68 AUC=95.0**
Patient-wise split of data			
Proposed model	RS-NIH	Binary classification (normal and pneumothorax CXRs)	**ACC=69.12 AUC=77.06**

RS: Random Samples of NIH Chest X-ray dataset

5.3.3 Segmentation

The proposed segmentation framework was trained and tested on SIIM Pneumothorax dataset using the official split as described in Table 5.1. The validation set was kept same as the testing set. During training process, Intersection over Union (IoU) was taken as performance measure with BCE-DICE Loss. While after generating predicted masks for the test set, Dice score was measured for every generate mask, where dice score is the measure of correctly identified area of pathology with respect to the ground truth mask.

Since we performed our experiments with two different sizes of images and mask, the selection of initial size (i.e. 256 x 256) was done randomly. However the size 384 x 384 was selected based on the concept of resolution given in [63]. In the paper a graph was given which shows that increasing resolution by factor “r” increases the performance of the CNN architecture,

so we took r=1.5 which yields 256*1.5=384, so for our second stage of the proposed segmentation framework we resized to images to 384 x 384 resolution.

5.3.3.1 Selection of dropout value

Firstly we performed two sets of experiments with image size 256 x 256. We tried two different values of dropout in the U-Net architecture. As discussed earlier, dropout layer is added in order to avoid overfitting. So we experimented with two different values which are 0.25 and 0.5. The reported results in Table 5.10 are on a small subset of test data containing 300 images and 300 corresponding masks. These results were obtained after training the first stage of proposed framework (i.e. U-Net model with Efficinet-B4 encoder after preprocessing and data augmentation) with whole training set containing 10675 images and 10675 corresponding binary masks for 20 epochs. The binary threshold was kept 0.5 for training and testing purpose.

Table 5.10: Comparison of dice score with different dropout values

Dropout	Threshold		Epochs	Dice Score (%)		
	Train	Test		Whole	Normal	Pneum
0.25	0.5	0.5	20	83.04	95.43	36.47
0.5	0.5	0.5	20	83.7	94.69	**42.4**

Pneum: Pneumothorax.

For fair comparison we calculated the dice score class wise (i.e. Normal and Pneumothorax) along with the dice score of whole test set (reported as “Whole” in Table 5.10). It can be seen that although the Whole dice score is not increased to a considerable value when dropout is changed from 0.25 to 0.5, however the dice score for pneumothorax increased by 5.93% which is good progress as our main aim is to identify the presence and location of pneumothorax. So we chose 0.5 for the rest of experimentation.

Note that for all the experiments, the default setting was: initial learning rate of 0.001 with Adam optimizer and binary threshold of 0.5. The learning rate was decayed in every epoch using cosine annealing scheduling. ImageNet weights were assigned to the U-Net architecture as initial weights. Dropout value was selected as 0.5 after comparing the two different values. Default resolution was 256 x256 except for the second stage in which resolution was set as 384 x 384.

5.3.3.2 Selection of Number of Epochs

After selection of dropout value, we first generated the predicted masks of the whole test set using the trained model from Section 5.3.3.1 with dropout value 0.5. The performance is shown in first row of Table 5.11. Then we trained the model with default experimental settings for 45 epochs and masks were predicted for the test set. The dice score of the whole test set along with class wise dice scores are reported in second row of Table 5.11.

Table 5.11: Comparison of dice score with different epochs

Dropout	Threshold		Epochs	Dice Score (%)		
	Train	Test		Whole	Normal	Pneum
0.5	0.5	0.5	20	82.14	93.64	**39.23**
0.5	0.5	0.5	45	82.2	96.7	28.4

Although the overall dice score increases a little bit with more number of epochs, however better results in terms of dice score of pneumothorax were achieved with less number of epochs, i.e. 20 epochs.

5.3.3.3 Two stage segmentation model

As the first stage of the proposed 2 stage segmentation model give better result with 20 epochs, so we loaded the weights of trained model with 20 epochs for the second stage, i.e. initial weights for training the U-Net Model with EfficientNet-B4 encoder with resolution of 384 x 384 were obtained from the trained model with resolution of 256 x 256 with 20 epochs. In the second stage the number of epochs was set to 25. The results of both the stages along with the experimental settings are summarized in Table 5.12.

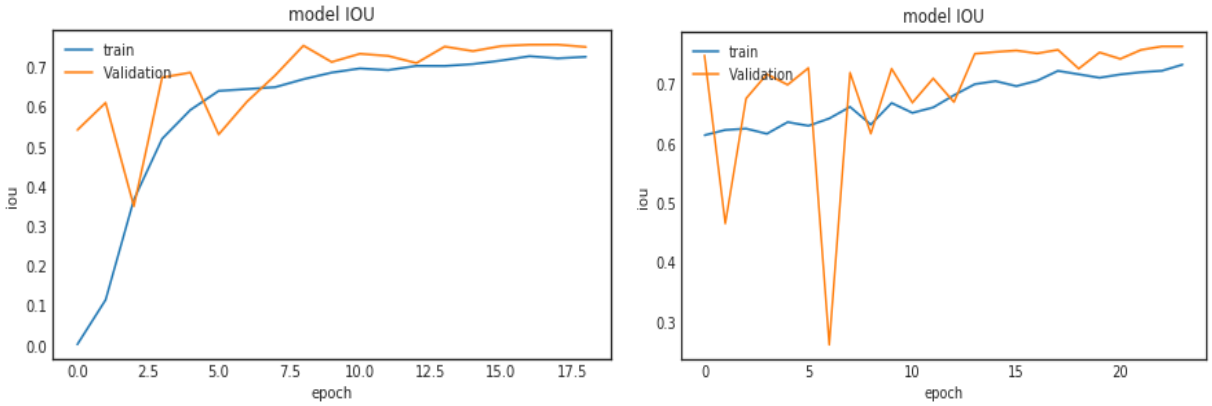
Table 5.12: Dice score for two stage segmentation model

Label	Res	BS	Init Wt	Epc	DO	Threshold		Dice Score (%)		
						Train	Test	Whole	Nor	Pnu
Set 1	256 x 256	16	Image-Net	20	0.5	0.5	0.5	82.14	93.64	39.23
Set 2	384 x 384	8	Weight from Set 2	25	0.5	0.5	0.5	**83.76**	93.09	**48.9**

Res: Resolution. BS: Batch size. Epc: Epochs. Init Wt: Initial Weights.
DO: Dropout. Nor: Normal. Pnu: Pneumothorax. Set: Setting

It can be seen that when the image and mask size is increased to 384 x 384, the overall dice score increase by 1.62% and the dice score of pneumothorax increase by 9.67% which is a huge improvement. The increase in overall dice score is not as greater as compared to that of pneumothorax samples, it is because there are very few samples of pneumothorax as compared to normal CXR in the training and testing set, so even an increase of almost 10% doesn't increase the overall score to a noticeable value.

The plot for training process of both stages is shown separately in Figure 5.3. As stated earlier, the performance measure during training process was IoU. Although the IoU score is not changed to a noticeable value in the right most graph, however this can be justified by the induvial results reported in Table 5.12 which proves the effectiveness of increasing the resolution of images and masks.



(a) (b)

Figure 5.3: Training and validation IoU score with respect to number of epochs. (a) With resolution 256 x 256. (b) With resolution 384 x 384.

5.3.3.4 Test Time Augmentation

Test time augmentation (TTA) is a process similar to Data augmentation done during training phase. Here, instead of showing the trained model only a cleaned, preprocessed version of test image, we will feed it with augmented images of testing image, final result is obtained by averaging the predictions for the original test image and the augmented test image.

In our experimentation, we performed two sets of TTA on the results obtained after the second stage of segmentation framework, first being a horizontal flip. Second being a combination of horizontal flip and one of Elastic Transform, Grid Distortion and Optical Distortion. The individual results are shown in Table 5.13.

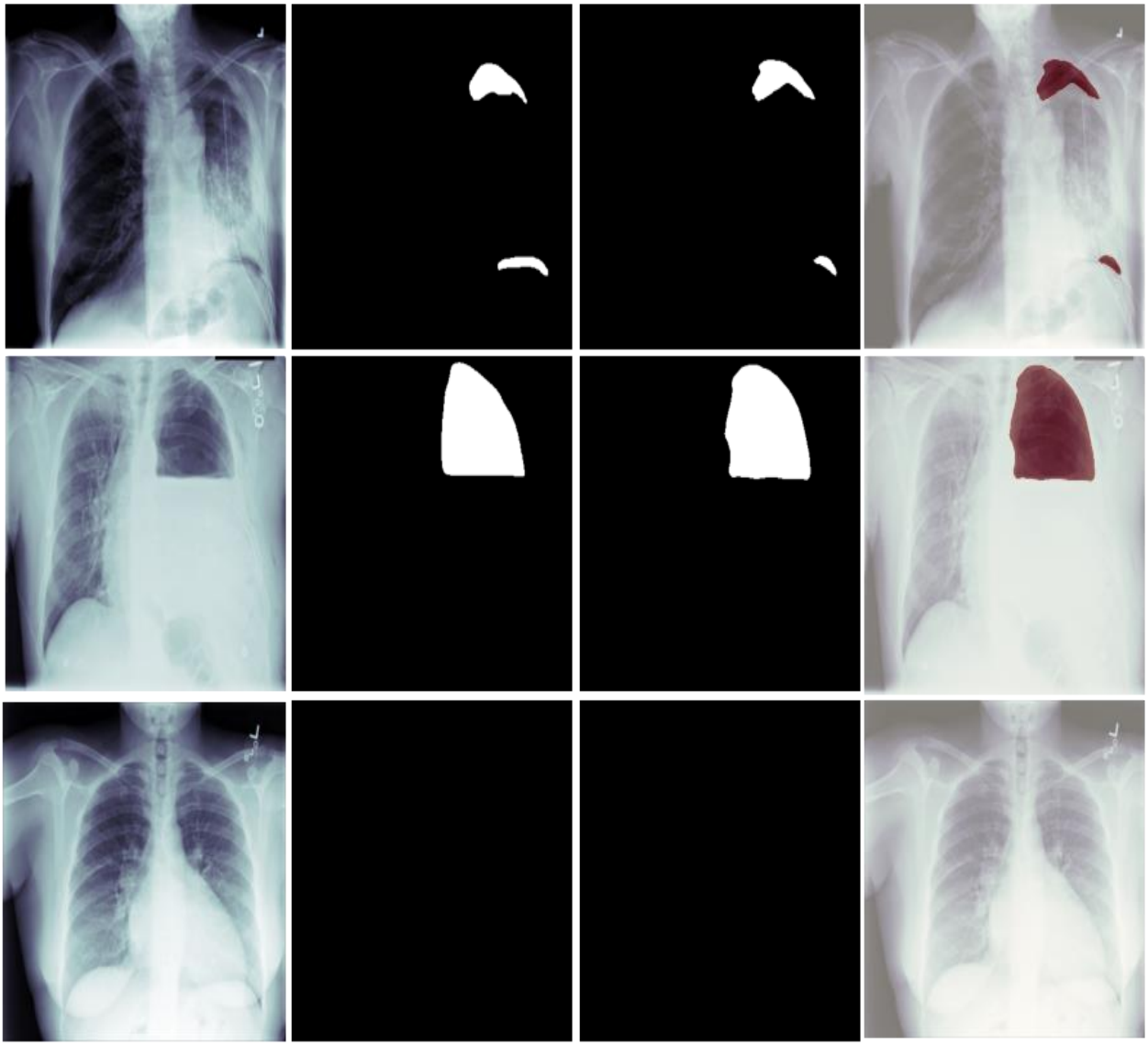
Table 5.13: Comparison of different sets of TTA

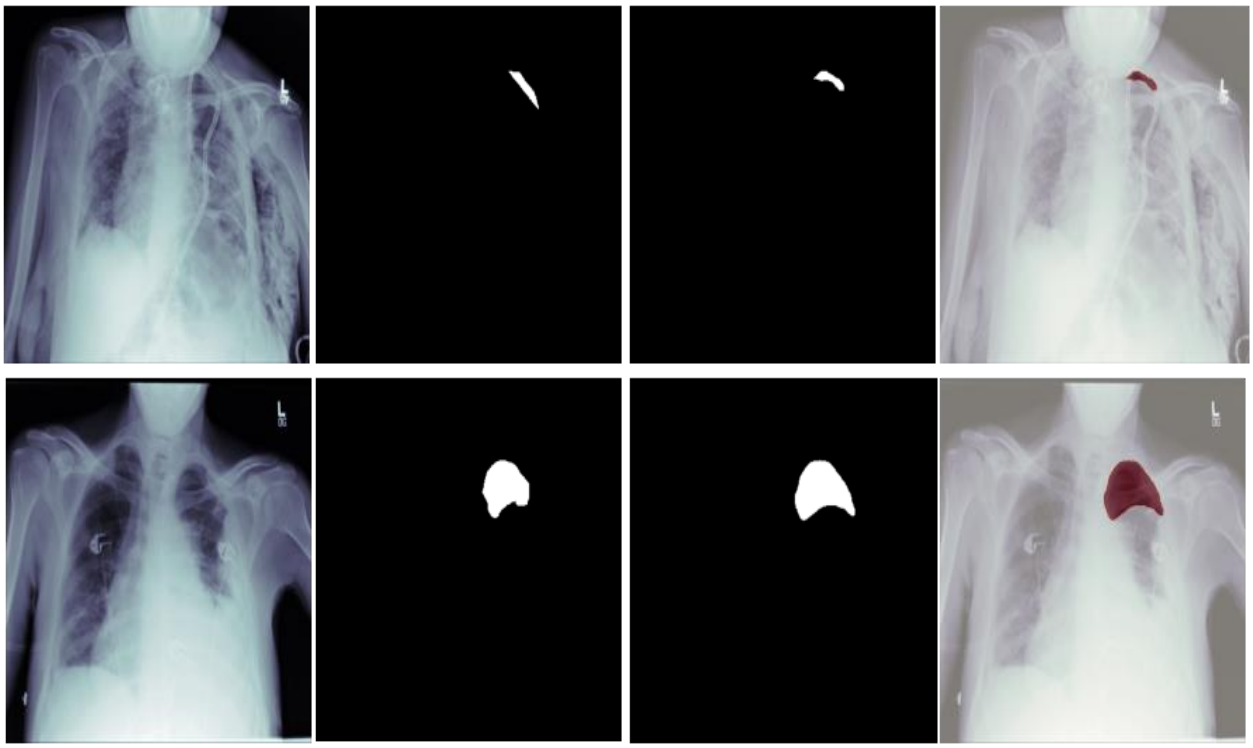
TTA	Dice Score (%)		
	Whole	Normal	Pnu
Horizontal flip	**84.56**	94.44	47.72
Horizontal flip & One of Elastic Transform, Optical Distortion and Grid Distortion	83.9	95.2	41.9

Nor: Normal. Pnu: Pneumothorax.

As results in Table 5.12 and Table 5.13 show that better results in terms of dice score of pneumothorax are achieved without TTA, so to conclude we can safely say that if the aim is to increase the precise identification of location of pathology then prediction of mask without TTA gives better result. However, in the literature, class wise dice scores have not been presented and only the overall dice scores are reported so we will report the overall dice score achieved after TTA as our final result.

Figure 5.4 shows some of the predicted masks against the ground truth mask for the test set of SIIM pneumothorax dataset.





(a) Original CXR (b) Original Mask (c) Predicted Mask (d) Mapped Predicted Mask on CXR image.

Figure 5.4: SIIM Pneumothorax Segmentation Results.

The comparison of existing work for the segmentation of region of pneumothorax on SIIM dataset is presented in Table 5.14. Although our obtained results are less than those reported in [43], however we hope that with more resources available, our proposed framework can achieve better results if trained on greater image resolution.

Table 5.14: Comparison of existing segmentation techniques

Author	Year	Dataset	Technique	Results
Mostayed [40]	2019	SIIM dataset	U-Net (with content adaptive conv.layers)	DSC= 76.04%
Ostap [41]	2020	SIIM dataset	GradCAM + IR-Network + U-Net with ResNet50	DSC= 76.7%

Jakhar [42]	2019	SIIM dataset	U-Net (with pre-trained ResNet encoder)	DSC= 84.3% IoU= 82.6%
Abedalla [45]	2020	SIIM dataset	2ST U-Net (2 stage U-Net with ResNet34 as encoder)	DSC=85.02%
Groza [43]	2020	SIIM dataset	LinkNet ensemble (se-resnext50, SENet-154, se-resnext101)	DSC= 88.21%
Our method	**2020**	**SIIM dataset**	**2 stage U-Net model with EfficientNet-B4 encoder**	**DSC=84.56%**

5.3.4 Combining Classification and Segmentation modules

The confusion matrix of the class predictions obtained from the generated masks as discussed in Section 4.3 is shown in Table 5.15.

Table 5.15: Confusion matrix of class predictions from segmented masks

	Predicted Class	
Actual Class	Normal	Pneumothorax
Normal	1021	61
Pneumothorax	54	236

Here it can be seen that greater number of samples are correctly classified as Normal class, this can be justified by the fact that the segmentation framework was trained with the original Class Imbalance dataset in which almost 70% of the data is comprised of Normal CXRs.

The results obtained after combining the class predictions by classification and segmentation framework, as discussed in Section 4.3, in terms of confusion matrix are shown in Table 5.16 and 5.17. The results in Table 5.16 are obtained from the Union of classification and

segmentation predictions, while the results in Table 5.17 are obtained from the Intersection of classification and segmentation predictions.

Table 5.16: Confusion Matrix for Union

	Predicted Class	
Actual Class	Normal	Pneumothorax
Normal	815	267
Pneumothorax	16	274

Table 5.17: Confusion Matrix for Intersection

	Predicted Class	
Actual class	Normal	Pneumothorax
Normal	1033	49
Pneumothorax	81	209

The results of combining classification and segmentation predicted class in terms of other performance measures including accuracy, recall, precision and specificity are shown in Table 5.18. The results shows that although accuracy is greater in case of Intersection of predictions, while recall (i.e. sensitivity) increases in case of Union of predictions, so we conclude that combining results by union is better as compared to intersection. It is because in the field of medicine, false positive is better than false negative.

Table 5.18: Performance of different class prediction approaches on SIIM dataset

Exp.No	**Technique (Class prediction by)**	**ACC (%)**	**REC (%)**	**PREC (%)**	**SPEC (%)**
1	Classification (VDV framework)	78.27	85.17	49.2	76.43
2	Segmentation (Two stage framework)	91.6	81.37	79.46	94.36
3	Union of 1 and 2	79.37	**94.44**	50.6	75.32
4	Intersection of 1 and 2	90.50	72.06	81.0	95.47

Exp No: Experiment Number.

CHAPTER 6: CONCLUSION & FUTURE WORK

6.1 Conclusion

The results proves that convolutional neural networks are excellent tool for designing an automatic pneumothorax diagnosis and segmentation system. Learning from the training data may take time but once the model is trained, it generates prediction in no time. For classification, the main issue was class imbalance while the existing class imbalance approaches have never been tested in chest radiography domain, so first task was to come up with the best technique. After that we were able to design a doubly ensemble model by combining the concepts of model level and data level ensembles. The results show that a doubly ensemble model perform better than a single data-level ensemble. The results achieved on SIIM dataset will serve as baseline as no one has yet used this data for classification purpose. Moreover the results obtained on RS-NIH dataset also authenticate the performance of our model. Additionally our proposed segmentation model achieved competitive performance on SIIM dataset for generating binary masks for the test samples, which is helpful in identifying the area of pathology. Moreover, based on the generated mask, a decision can be made if the CXR has pathology or not. Thus a combination of these predictions and the ones obtained during classification process give better results in terms of sensitivity, thus increasing the capability of automated detection of pathology.

6.2 Contributions

- Comparison of existing class imbalance approaches for classification model.
- Using the first openly available CXR dataset dedicated for pneumothorax identification.
- Automated pneumothorax classification system from model-level-ensemble of multiple data-level-ensembles.
- Reproducing the experimental results obtained in [45] for segmentation of the region of interest.
- Combining the classification and segmentation results.

6.3 Future Work

As a future work we can propose:

- Experimenting the proposed framework for other class-imbalance medical images dataset and for bigger chest X-rays datasets like full NIH Chest X-ray14 dataset, CheXpert and MIMIC dataset.
- Exploring different CNN architectures as feature extractor for the proposed VDV framework.
- Exploring the effect of higher input resolution and different types of encoder in U-Net architecture for the two-stage-segmentation-model.
- As mostly researchers had used U-Net architecture for segmentation purpose so there is a huge scope for exploring other segmentation models like LinkNet, Feature Pyramid Network (FPN) and Pyramid Scene Parsing Network (PSPNet) for medical images segmentation.

REFERENCES

1. Saeed, D.Y., Hamza, A.A., & Ismail, O.M. (2015). Pattern and Management Outcome of Chest Injuries in Omdurman Teaching Hospital. The Journal of medical research, 15.
2. Gary M Weiss. “Mining with rarity: a unifying framework”. In: ACM Sigkdd Explorations Newsletter 6.1 (2004), pp. 7–19.
3. How the Lungs Work. (2020). Retrieved 29 September 2020, from https://www.templehealth.org/services/lung/patient-care/patient-resources/understanding-lung-disease/how-lungs-work.
4. diaphragm | Definition, Function, & Location. Retrieved 14 October 2020, from https://www.britannica.com/science/diaphragm-anatomy.
5. Punctured lung (pneumothorax): Symptoms, treatment, and recovery. Retrieved 17 October 2020, from https://www.medicalnewstoday.com/articles/320797.
6. Publishing, H. (2019). Pneumothorax - Harvard Health. Retrieved 29 September 2020, from https://www.health.harvard.edu/a_to_z/pneumothorax-a-to-z.
7. How to identify pneumothorax on a chest x-ray. (2018). Retrieved 29 September 2020, from https://www.medmastery.com/magazine/how-identify-pneumothorax-chest-x-ray.
8. Rutherford, R. Pneumothorax. Retrieved 29 September 2020, from https://www.slideserve.com/romeo/pneumothorax.
9. Raoof S, Feigin D, Sung A, Raoof S, Irugulpati L, Rosenow EC. Interpretation of plain chest roentgenogram. Chest. 2012 Feb; 141(2):545–58. https://doi.org/10.1378/chest.10-1302 PMID.
10. Vasudevan, H., Michalas, A., Shekokar, N., & Narvekar, M. (2020). Advanced computing technologies and applications (p. 300). Singapore: Springer. doi: 10.1007/978-981-15-3242-9.
11. Buda, M., Maki, A., & Mazurowski, M. A. (2018). A systematic study of the class imbalance problem in convolutional neural networks. Neural Networks, 106, 249-259. doi: 10.1016/j.neunet.2018.07.011.
12. Raskutti, B., & Kowalczyk, A. (2004). Extreme re-balancing for SVMs: a case study. ACM Sigkdd Explorations Newsletter, 6(1), 60-69. doi: 10.1145/1007730.1007739.

13. Levi, G., & Hassner, T. (2015). Age and gender classification using convolutional neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops (pp. 34-42).
14. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, 321-357. doi: 10.1613/jair.953.
15. Jo, T., & Japkowicz, N. (2004). Class imbalances versus small disjuncts. ACM Sigkdd Explorations Newsletter, 6(1), 40-49. doi: 10.1145/1007730.1007737.
16. Guo, H., & Viktor, H. L. (2004). Learning from imbalanced data sets with boosting and data generation: the databoost-im approach. ACM Sigkdd Explorations Newsletter, 6(1), 30-39. doi: 10.1145/1007730.1007736.
17. Chollet, F. (2016). Building powerful image classification models using very little data [Blog]. Retrieved from https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html.
18. Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., & Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications. Expert Systems with Applications, 73, 220-239. doi: 10.1016/j.eswa.2016.12.035.
19. Sapp, S., van der Laan, M. J., & Canny, J. (2014). Subsemble: an ensemble method for combining subset-specific algorithm fits. Journal of applied statistics, 41(6), 1247-1259. doi: 10.1080/02664763.2013.864263.
20. Sun, Z., Song, Q., Zhu, X., Sun, H., Xu, B., & Zhou, Y. (2015). A novel ensemble method for classifying imbalanced data. Pattern Recognition, 48(5), 1623-1637. doi: 10.1016/j.patcog.2014.11.014.
21. Salunkhe, U. R., & Mali, S. N. (2016). Classifier ensemble design for imbalanced data classification: a hybrid approach. Procedia Computer Science, 85, 725-732. doi: 10.1016/j.procs.2016.05.259.
22. Claessen, O. (2019). On the classification of imbalanced image datasets (Master's thesis).
23. Leevy, J. L., Khoshgoftaar, T. M., Bauder, R. A., & Seliya, N. (2018). A survey on addressing high-class imbalance in big data. Journal of Big Data, 5(1), 42.

24. Chan, Y., Zeng, Y., Wu, H., Wu, M., & Sun, H. (2018). Effective Pneumothorax Detection for Chest X-Ray Images Using Local Binary Pattern and Support Vector Machine. Journal Of Healthcare Engineering, 2018, 1-11. doi: 10.1155/2018/2908517.
25. Yoon, Y., Hwang, T., & Lee, H. (2018). Prediction of radiographic abnormalities by the use of bag-of-features and convolutional neural networks. The Veterinary Journal, 237, 43-48. doi: 10.1016/j.tvjl.2018.05.009.
26. Park, S., Lee, S. M., Choe, J., Cho, Y., & Seo, J. B. (2019, January). Performance of a deep-learning system for detecting pneumothorax on chest radiograph after percutaneous transthoracic needle biopsy. European Congress of Radiology, 2019. doi:10.26044/ecr2019/C-0334.
27. Gooßen, A., Deshpande, H., Harder, T., Schwab, E., Baltruschat, I., Mabotuwana, T., Cross, N., & Saalbach, A. (2019). Deep Learning for Pneumothorax Detection and Localization in Chest Radiographs. arXiv preprint arXiv:1907.07324.
28. Li, X., Thrall, J. H., Digumarthy, S. R., Kalra, M. K., Pandharipande, P. V., Zhang, B., ... & Li, Q. (2019). Deep learning-enabled system for rapid pneumothorax screening on chest CT. European journal of radiology, 120. doi: 10.1016/j.ejrad.2019.108692.
29. Lindsey, T., Lee, R., Grisell, R., Vega, S., & Veazey, S. (2018, November). Automated pneumothorax diagnosis using deep neural networks. In Iberoamerican Congress on Pattern Recognition (pp. 723-731). Springer, Cham. doi: 10.1007/978-3-030-13469-3_84.
30. Geva, O., Zimmerman-Moreno, G., Lieberman, S., Konen, E., & Greenspan, H. (2015, March). Pneumothorax detection in chest radiographs using local and global texture signatures. In Medical Imaging 2015: Computer-Aided Diagnosis (Vol. 9414, p. 94141P). International Society for Optics and Photonics. doi:10.1117/12.2083128.
31. Bharati, S., Podder, P., & Mondal, M. R. H. (2020). Hybrid deep learning for detecting lung diseases from X-ray images. Informatics in Medicine Unlocked, 20, 100391. doi: 10.1016/j.imu.2020.100391.
32. Tang, Y. X., Tang, Y. B., Han, M., Xiao, J., & Summers, R. M. (2019, April). Abnormal chest x-ray identification with generative adversarial one-class classifier. In 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019) (pp. 1358-1361). IEEE. doi: 10.1109/ISBI.2019.8759442.

33. Guan, Q., Huang, Y., Zhong, Z., Zheng, Z., Zheng, L., & Yang, Y. (2018). Diagnose like a radiologist: Attention guided convolutional neural network for thorax disease classification. arXiv preprint arXiv:1801.09927.
34. Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., & Lungren, M. P. (2017). Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. arXiv preprint arXiv:1711.05225.
35. Allaouzi, I., & Ahmed, M. B. (2019). A novel approach for multi-label chest X-ray classification of common thorax diseases. IEEE Access, 7, 64279-64288.
36. Mondal, S., Agarwal, K., & Rashid, M. (2019, November). Deep Learning Approach for Automatic Classification of X-Ray Images using Convolutional Neural Network. In 2019 Fifth International Conference on Image Information Processing (ICIIP) (pp. 326-331). IEEE. doi: 10.1109/iciip47207.2019.8985687.
37. Jun, T. J., Kim, D., & Kim, D. (2018). Automated diagnosis of pneumothorax using an ensemble of convolutional neural networks with multi-sized chest radiography images. arXiv preprint arXiv: 1804.06821.
38. Röhrich, S., Schlegl, T., Bardach, C., Prosch, H., & Langs, G. (2020). Deep learning detection and quantification of pneumothorax in heterogeneous routine chest computed tomography. European Radiology Experimental, 4, 1-11.
39. Luo, G., Liu, Z., Wang, Q., Liu, Q., Zhou, Y., Xu, W., ... & Cheng, J. Z. (2019, November). Fully Convolutional Multi-Scale ScSE-DenseNet for Automatic Pneumothorax Segmentation in Chest Radiographs. In 2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) (pp. 1551-1555). IEEE.
40. Mostayed, A., Wee, W., & Zhou, X. (2019, December). Content-Adaptive U-Net Architecture for Medical Image Segmentation. In 2019 International Conference on Computational Science and Computational Intelligence (CSCI) (pp. 698-702). IEEE.
41. Viniavskyi, O., Dobko, M., & Dobosevych, O. (2020). Weakly-Supervised Segmentation for Disease Localization in Chest X-Ray Images. arXiv preprint arXiv:2007.00748.
42. Jakhar, K., Bajaj, R., & Gupta, R. (2019). Pneumothorax Segmentation: Deep Learning Image Segmentation to predict Pneumothorax. arXiv preprint arXiv: 1912.07329.

43. GROZA, V., & KUZIN, A. (2020, April). Pneumothorax Segmentation with Effective Conditioned Post-Processing in Chest X-Ray. In 2020 IEEE 17th International Symposium on Biomedical Imaging Workshops (ISBI Workshops) (pp. 1-4). IEEE.
44. Do, S., Salvaggio, K., Gupta, S., Kalra, M., Ali, N. U., & Pien, H. (2012). Automated quantification of pneumothorax in CT. Computational and Mathematical methods in Medicine, 2012.
45. Abedalla, A., Abdullah, M., Al-Ayyoub, M., & Benkhelifa, E. (2020). The 2ST-UNet for Pneumothorax Segmentation in Chest X-Rays using ResNet34 as a Backbone for U-Net. arXiv preprint arXiv:2009.02805.
46. Rashid, R., Khawaja, S. G., Akram, M. U., & Khan, A. M. (2018, December). Hybrid RID Network for Efficient Diagnosis of Tuberculosis from Chest X-rays. In 2018 9th Cairo International Biomedical Engineering Conference (CIBEC) (pp. 167-170). IEEE. doi: 10.1109/CIBEC.2018.8641816.
47. Bunrit, S., Kerdprasop, N., & Kerdprasop, K. (2019). Evaluating on the Transfer Learning of CNN Architectures to a Construction Material Image Classification Task. Int. J. Mach. Learn. Comput, 9(2), 201-207. doi: 10.18178/ijmlc.2019.9.2.787 .
48. Understanding of Convolutional Neural Network (CNN) — Deep Learning. Retrieved 19 October 2020, from https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148.
49. Rashid, R. (2018). Deep Convolutional Neural Networks for Lungs Segmentation and Diagnosis of Tuberculosis from Chest X-Rays (Master). National University of Science And Technology.
50. Yani, M. (2019, May). Application of transfer learning using convolutional neural network method for early detection of terry's nail. In Journal of Physics: Conference Series (Vol. 1201, No. 1, p. 012052). IOP Publishing.
51. CS231n Convolutional Neural Networks for Visual Recognition. Retrieved 8 September 2020, from https://cs231n.github.io/transfer-learning/.
52. Prajapati, S. A., Nagaraj, R., & Mitra, S. (2017, August). Classification of dental diseases using CNN and transfer learning. In *2017 5th International Symposium on Computational and Business Intelligence (ISCBI)* (pp. 70-74). IEEE.

53. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv: 1409.1556.
54. VGG16 - Convolutional Network for Classification and Detection. (2018). Retrieved 28 September 2020, from https://neurohive.io/en/popular-networks/vgg16.
55. Mateen, M., Wen, J., Song, S., & Huang, Z. (2019). Fundus image classification using VGG-19 architecture with PCA and SVD. Symmetry, 11(1), 1. doi: 10.3390/sym11010001.
56. Review: VGGNet — 1st Runner-Up (Image Classification), Winner (Localization) in ILSVRC 2014. Retrieved 19 October 2020, from https://medium.com/coinmonks/paper-review-of-vggnet-1st-runner-up-of-ilsvlc-2014-image-classification-d02355543a11.
57. Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).
58. Li, Z., & Tang, X. (2007). Using support vector machines to enhance the performance of Bayesian face recognition. IEEE Transactions on Information Forensics and Security, 2(2), 174-180.
59. Support Vector Machine—Simply Explained. Retrieved 20 October 2020, from https://towardsdatascience.com/support-vector-machine-simply-explained-fee28eba5496.
60. Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011, from https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html.
61. Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham.
62. EfficientNet: Scaling of Convolutional Neural Networks done right. (2020). Retrieved 20 October 2020, from https://towardsdatascience.com/efficientnet-scaling-of-convolutional-neural-networks-done-right-3fde32aef8ff.
63. Tan, M., & Le, Q. V. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. arXiv preprint arXiv:1905.11946.
64. Marsh (2020). Chest X-Ray Images with Pneumothorax Masks, Version 2. Retreived from https://www.kaggle.com/vbookshelf/pneumothorax-chest-xray-images-and-masks.

65. Random Sample of NIH Chest X-ray Dataset. (2017). Retrieved 6 September 2020, from https://www.kaggle.com/nih-chest-xrays/sample.
66. Degiorgis, M., Gnecco, G., Gorni, S., Roth, G., Sanguineti, M., & Taramasso, A. C. (2012). Classifiers for the detection of flood-prone areas using remote sensed elevation data. Journal of hydrology, 470, 302-315. doi: 10.1016/j.jhydrol.2012.09.006.
67. Taylor AG, Mielke C, Mongan J (2018) Automated detection of moderate and large pneumothorax on frontal chest X-rays using deep convolutional neural networks: A retrospective study. PLoS Med 15(11): e1002697. doi: 10.1371/journal.pmed.1002697.