Jimma University

Jimma Institute of Technology

Faculty of Computing and Informatics

MSc in Information Technology

Automatic Detection of Pulmonary Tuberculosis with MobileNet and Entropy-Based Feature Fusion Strategy

By:

Israel Wondimu

A Thesis Submitted to the School of Graduate Studies of Jimma University in Partial fulfillment of the Requirements for the Degree of Masters of Science in Information Technology

Principal Advisor: Worku Jifara (Ph.D)

Co-Advisor: Admas Abtew (Ph.D Candidate)

July, 2023

Jimma, Ethiopia

Jimma University

Jimma Institute of Technology

Faculty of Computing and Informatics

A Thesis Submitted to the School of Graduate Studies of Jimma University in Partial fulfillment of the Requirements for the Degree of Masters of Science in Information Technology

Approval sheet

Thesis Title: Automatic Detection of Pulmonary Tuberculosis with MobileNet and Entropy-Based Feature Fusion Strategy

Submitted By:

Student Name: Israel Wondimu

Signature

Approved by:

Main Advisor: Worku Jifara (Ph.D)

Signature

Co-Advisor: Admas Abtew (Ph.D Candidate)

Signature

External Examiner: Dr. Teklu Urgessa

Signature

Internal Examiner: Yonas Gido

Signature

Chairman: Intelligent System Chair __________

Signature

July, 2023
Jimma, Ethiopia

Acknowledgement

First and foremost, I would like to thank the Almighty God who blessed every minute of our life. Next, I would like to express my deepest gratitude and appreciation to my advisor Dr. Worku Jifara and co-advisor Admas Abtew for your advice and comment on this thesis work. I would like to take this occasion to express my deep appreciativeness to my families, friends and peers who, in one or the other way brought me up to a success in my academic endeavor, especially, my wife S/r. Tirhas Hindeya for their endless moral support. Next, I would like to thank you Jimma University Medical Center for providing me the necessary dataset that are used for the implementation of our work.

Abstract

The mycobacterium tuberculosis causes the chronic necrotizing disease known as tuberculosis. Complex pulmonary tuberculosis is a bacterial infection brought on by the easily airborne Mycobacterium TB. The primary infection is caused by mycobacterium TB, which gradually multiplies in the lungs after becoming infected. Frequent medical types of active lung TB symptoms and indicators include fever, coughing, weight loss, hemoptysis, and night sweats. The lack of human resources and radiological interpretation know-how weakens TB screening programs, especially in TB endemic countries. Additionally, poor judgment, a lack of proficiency, and diagnostic mistakes can directly result in patient damage or death. Making the right choice and automatically detecting CXR can serve as a reliable substitute for more complex and technically demanding methods. So that to address this problem, we created an AI model that automatically detect the existence of Pulmonary Tuberculosis disease on human beings. The model attempts to support physicians and patients for the diagnosis and treatment of the disease. We collected the necessary dataset from Jimma University medical center. We used image processing techniques that include image preprocessing, segmentation, feature extraction, and detection. We compared the newly proposed TBDMobileNetV2 model with other state-of-the-art models. TBDMobileNetV2 achieved 96.11% testing accuracy; AlexNet model achieved 83.39% testing accuracy; and VGGNet model achieved 73.33% testing accuracy. So that the proposed TBDMobileNetV2 model outperforms the other models f or the detection of Pulmonary Tuberculosis disease.

Keywords: MobileNetV2, Entropy feature fusion, Pulmonary Tuberculosis, Image processing, Image detection

Table of Content

List of Figures

List of Acronyms/Abbreviations

AI	Artificial Intelligence
CAD	Computer Aided Detection and Diagnosis
CNN	Conventional Neural Network Convolutional
CNN's	Neural Networks
COVID-19	Coronavirus Disease 2019
CT	Computed Tomography
CXR	Chest X-Ray (radiography)
DIP	Digital Image Processing
DL	Deep Learning
EM	Electro Magnetic
GLCM	Gray-Level Co-occurrence
HIV	Human Immune Deficiency Viruses
JUMC	Jimma University Medical Center
LTBI	Latent Tuberculosis Infection
MRI	Magnetic Resonance Imaging
MTB	Mycobacterium Tuberculosis
NTB	Not Tuberculosis
ROI	Regions Of Interest
PTB	Pulmonary Tuberculosis
TB	Tuberculosis
TBDMobileNetV2	Tuberculosis Detection MobileNetV2
WHO	World Health Organization

Chapter One

Introduction

1. Background of the Study

Tuberculosis is a chronic necrotizing illness that is affected by the mycobacterium tuberculosis complex [1]. According to 2020 Global TB Report, Ethiopia is amid the 30 high TB and TB/HIV load countries, with annual expected TB occurrence of 140/100,000 residents and death rate of 19 per 100,000 residents [2]. This consequence of the lung diseases called pulmonary tuberculosis (PTB). It is one of the top 10 causes of death and the leading cause of a single infectious agent and millions of people continue to fall sick with TB each year [3].

Pulmonary TB is a bacterial infection as a consequence of Mycobacterium TB, which easily feast airborne. Once contaminated, mycobacterium TB multiplies gradually in the pulmonary and this characterizes the primary infection. Typical medical sorts in terms of symptoms and signs connected with lively lung TB are contained fever, cough, weight loss, hemoptysis, night sweats, fatigue, and chest pain [3].

Globally TB reports showed that there were an expected 10 million occurrence cases and 12 million widespread cases of TB worldwide. Generally, 90% of the pollution happened in adults; of this 9% remained people living with HIV (72% in Africa). About 26% occurrence of TB cases happened in Africa and 23% of the world's population are predictable to have a latent TB infection and are consequently at risk of going active TB throughout their lifetime [3]. The same becomes even more challenging when TB is associated with other diseases like COVID-19.

Chest CXR imaging is advantageous either for screening or for diagnosing Tuberculosis. We have the choice to take the chest x-ray in three dimensions: anterior-posterior, posterior-anterior, or lateral position. After taking the chest x-ray, we have to check: cavitation or consolidation, millinery mottling, pleural effusion for active TB and fibro nodular scarring, traction bronchiectasis, calcified lymph node, for the past TB infection history or in the TB treatment regimen. that can focus on points that are invisible to the human eye to support and change the manual TB screening model is proposed to detect the TB (especially at the latent stage) in the CXRs using digital image processing with the help of textural feature extraction method which is GLCM [4]. The application of AI CADx for automatic detection in medicine is becoming an interesting tool for physicians [5]. Beyond the solution proposal, it was implemented using the three available datasets and evaluated the model accordingly.

The development of digital chest radiography (CXR) increased the acceptance of x-ray imaging for several motives like low implementation cost, and medical price and became perceived as part of repetitive patient care.

Computer vision is related to image processing in the sense that the computer vision front-end is comprised of image processing techniques such as noise reduction, whitening, or image enhancement. Machine learning on the other side is flexible as it can be used in either 20 computer vision or image processing. For example, the main goal of computer vision is to extract meaningful information from images or videos whether a certain object is present or not in a particular scene and it is quite more complex than others. Similarly, machine learning is to optimize or enhance differentiable parameters so that a certain cost or loss function is minimized and the goal of image processing is to enhance or compress image or video information at pixel-wise operations.

MobileNetV2s are based on a streamlined architecture that uses depth-wise separable convolutions to build lightweight deep neural networks. We introduce two simple global hyper-parameters that efficiently trade-off between latency and accuracy. These hyper-parameters allow the model builder to choose the right sized model for their Model based on the constraints of the problem. We present extensive experiments on resource and accuracy tradeoffs and show strong performance compared to other popular models on Image Net classification. We then demonstrate the effectiveness of Mobile Nets across a wide range of applications and use cases including object detection, fine-grain classification, face attributes, and large-scale geo-localization [6].

MobileNetV2 is a model which does the same convolution as done by CNN to filter images but in a different way than those done by the previous CNN. It uses the idea of Depth convolution and point convolution which is different from the normal convolution as done by normal CNNs. This increases the efficiency of CNN to predict images and hence they can be able to compete in the mobile systems as well. Since these ways of convolution reduce the comparison and recognition time a lot, and it provides a better response in a very short time and hence we are using them as our image recognition model [6].

In the 21 century, most businesses are using machine learning and deep learning to automate their process, and decision-making, increase efficiency in disease detection, etc. One way to evaluate model efficiency is accuracy**.** The higher the accuracy, the more efficient the model is. It's therefore essential to increase the accuracy by optimizing the model; by applying loss functions [6].

We would say that an unbiased coin has high entropy because the uncertainty of the value of X is the highest possible. Finding out the value of the coin flip tells you lots of information you didn't already know. A biased coin has low entropy because the uncertainty of X is the lowest possible. Finding out the value of the coin flip doesn't tell you anything, because you already knew that it would land heads. The unit for entropy is bits. This can be a bit tough to understand at first [6].

Imagine that you have a communication channel with your friend to tell them the outcome of a given coin flip. Intuitively, for a fair coin, you will always need one bit to tell them the result of the coin flip (heads or tails? 1 or 0?). In the case of a biased coin, you won't need any bits at all. Because the coin will always be heading, your friend will always know the outcome of any number of samples [6].

1.2. Statement of Problem

Tuberculosis (Tb) is a contagious disease that is a main reason of ill health and one of the leading causes of death worldwide. Until the coronavirus (COVID-19) pandemic, Tuberculosis was the leading cause of death from a particular infectious agent, ranking above HIV/AIDS. Without treatment, the death rate from Tb disease is high (about 50%) [7]. with currently-recommended treatments (a 4–6 months course of anti-Tb drugs), about 85% of people can be cured. some countries have already reduced their burden of Tb disease to fewer than 10 cases and less than one death per 100,000 population per year [8]. Despite notable progress during the millennium development goals era globally, TB ranks to be the top killer infectious diseases worldwide. In 2016, there were 6.3 million new TB cases and 1.3 million TB deaths. In Ethiopia, Tuberculosis is still a main public health problem. The country is still among the 22 high TB burden countries with high number of missed and infectious TB cases in the community [9].

The research work on image processing and classification done on various domains to increase the efficiency of the manual based works with the development of machine learning and deep learning models. Image processing for disease classification done using machine learning and deep learning to solve the shortage of professionals and misdiagnosis of diseases from image and get little enough accuracy [10]. The speed of classification and detection of an image is an issue in various machine learning and deep learning algorithms.

Pulmonary tuberculosis (TB) is an infectious disease caused by Mycobacterium tuberculosis. Exposure to infection, being born in a nation with an endemic disease, and HIV infection are important risk factors. Cough, fever, and weight loss may be symptoms. If pulmonary TB is suspected, the patient should be isolated, a chest x-ray should be taken, three sputum samples should be taken for acid-fast bacilli smear and culture, and at least one respiratory specimen should undergo a nucleic acid amplification test. It is strongly advised to use directly observed therapy, which is especially appropriate for populations where adherence cannot be anticipated. In order to stop the spread of infectious TB, early detection and application of appropriate therapy are essential.

The lack of human resources for appropriate therapy weakens TB screening programs, especially in TB endemic countries. In addition to this poor judgment, inadequate skill, and diagnostic errors can directly cause patient death and harm. Making an appropriate decision and automatic detection of pulmonary tuberculosis disease can be used as a consistent alternative to more refined and technologically demanding techniques [11]. So that, a new approach of detection of pulmonary tuberculosis disease image with MobileNetV2 is required to improve the speed of detection of an image with Pulmonary Tuberculosis disease infected [12].

Therefore, our research filled the gap of detection with high accuracy and speed by using MobileNetV2 and entropy-based on feature fusion. To overcome all the above issues, acquiring mass detection capabilities, improving the detection, and development of a relatively accurate and automatic model for pulmonary tuberculosis disease detection is required. To the best of our knowledge, there is no a study conducted for automatic detection of pulmonary tuberculosis with MobileNetV2 and entropy-based feature fusion.

1.3. Research Questions

At the end of the research work, the following question will answer.

- ✓ How to apply entropy-based feature fusion strategy for the detection of pulmonary tuberculosis disease using MobileNet?
- ✓ How to develop MobileNet model with entropy-based feature fusion strategy for detection of pulmonary tuberculosis disease?
- ✓ To what extent the proposed model performs for the detection of pulmonary tuberculosis

disease?

- ✓ To what extent the proposed MobileNet model with entropy-based feature fusion strategy outperforms the state-of-the-art models for pulmonary tuberculosis disease detection?

1.4. Objective of the study

1.4.1. General Objective

The general objective of the study is to design and develop a model for automatic detection of Pulmonary Tuberculosis with MobileNetV2 and Entropy based feature fusion strategy.

1.4.2. Specific Objective

To achieve the general objective, these specific objectives are addressed.

- ✓ Review different literature on TB disease detection.
- ✓ Collect X-ray image dataset from pulmonary tuberculosis infected image and non-infected image.
- ✓ Develop MobileNet model with entropy-based feature fusion strategy.
- ✓ Train MobileNet model with the labeled training dataset.
- ✓ Evaluate the performance of MobileNet model with the testing dataset.

1.5. Scope and delimitation of the study

The research work attempted on automatic pulmonary tuberculosis disease detection in CXR images with MobileNet and entropy-based feature fusion strategy. However, the research does not provide other detection mechanisms such as skin tests, sputum tests, and patients historical or oral events. The research also does not provide the severity of the disease and the impact of being infected with the disease. The process of treatment and recommendation of the disease is also not assessed with this research work.

1.6. Significance of the study

The mainstream of tuberculosis finding techniques usually require human mediation to upkeep in the clarification of their results. A challenge conventional microscopy that is the gold standard method of tuberculosis diagnosis has produced slight success as the degree of accuracy for disease detection reported remains low. In this research work, the development of Pulmonary Tuberculosis disease detection using MobileNetV2 and Entropy-Based Feature Fusion Strategy techniques are carried out for effective and efficient diagnosis and treatment of the disease.

This research work provides a research output for Pulmonary Tuberculosis detection researchers in the development of Pulmonary Tuberculosis detection systems from chest x-ray images; the model is helpful to work with embedded device such as smartphone and other hand-held devices; the model is also helpful to work with embedded device such as smartphone and other hand-held devices; and the study helps to initiates researchers to do pulmonary tuberculosis detections with MobileNetV2 and Entropy-Based Feature Fusion Strategy

1.7. Organization of the Research

The thesis is structured into five different chapters. The first chapter presents a primary introduction to this study. It offers the general structure included in this study. It provides enough background information to help the reader understand the reason behind the study and what the researcher plans to accomplish by carrying out the research. The second chapter studies an explanation of pulmonary tuberculosis which will be selected for demonstrating the proposed methodology, MobileNetV2 and Entropy based feature fusion techniques that are used in this research, and present reviews of previous work related to the study topic with specific reference to the research objectives. It presents summaries from books, journals, and collected works that are helpful in accomplishing this work. The third chapter presents the methodology of the new proposed work with the use of MobileNetV2 and entropy-based feature fusion strategy. The fourth chapter presents the preparation dataset for the experiment and the result of training and testing models with the collected dataset. Comparison between different models performed on this chapter. The fifth chapter presents conclusion on this research work, contribution to the scientific community and future work for researchers.

Chapter Two

Literature Review

2.1. Introduction

The use of a digital computer to run an algorithm on digital images is called digital image processing. Digital image processing has significant benefits over analog image processing as a subfield or area of digital signal processing. It permits the application of a considerably wider variety of algorithms to the input data and can prevent issues like the accumulating noise and distortion during processing. Digital image processing can be described as a multidimensional system because images can exist in at least two dimensions and possibly more. The type of image that is composed of pixels is a digital image. It consists of written texts, images, and artwork. Any image machine, including a digital camera, was used to capture it. It is used to define the quality of the image because it is composed of pixels. The quality of the image is assessed using the pixel value. The value of a pixel is expressed in binary code. The value of the pixel per inch can be used to determine resolution. TIFF, GIF, JPEG, and BMP are some of the file formats that are used to store information on a computer.

A digital image is an image used for image classification which are exist in different forms. Some of them are discussed as follow.

I. Binary images

Binary images are the simplest images category and contains on two values. These values are categorized as black and white or 0 and 1. A binary image is discussed as a 1-bit image since it takings only 1 binary digit to characterize apiece of pixel. These categories of images are often used in applications where the only information necessary for common shape or summary, for instance optical character recognition (OCR).

II. Gray-scale images

These types of images are known as monochrome (one-color) images. It includes gray-level information, no color information. The number of bits used for apiece of pixel decides the amount of various gray levels accessible. The usual gray-scale image holds 8bits/pixel data, which allows us to have 256 different gray levels.

III. Color images

Color images can be demonstrated as three-band monochrome image data, where apiece of band data relates to a different color. The real information stored in the digital image data is the gray-level information in each spectral band.

This color images are characterized as red, green, and blue (RGB images). Using the 8-bit monochrome standard as a model, the equivalent color image would have 24-bits/pixel (8-bits for each of the three-color bands red, green, and blue).

IV. Multispectral images

Multispectral images normally have information outside the normal human perceptual range. This may include infrared, ultraviolet, X-ray, acoustic, or radar data. These are not images in the normal sense since the information represented is not directly visible by the human system.

2.2. Overview of Pulmonary Tuberculosis

Tuberculosis (TB) is one of the key worldwide health threats principals to illness and death [13, 14]. It is spread from person to person by lung droplets. While some people develop active tuberculosis disease after infection, almost all tuberculosis infections are asymptomatic and remain latent. Latent tuberculosis infection (LTBI) itself improvements to active disease in around 5% to 10% of infected persons. The rate of progression is much greater in immune compromised individuals. The expected 2 billion people living with LTBI represent a vast reservoir of potential cases of tuberculosis around the world. This reservoir of LTBI is therefore a major barrier to the ultimate control and elimination of tuberculosis. It has been expected that around one-third of world's people can be affected with TB and above 95% patients died in developing countries [15]. In general, TB affects the lungs, but additional parts of the body can also be affected [16]. The correct symptom of active TB are a long term cough with blood holding sputum, fever, night sweats, and weight loss [17]. Overall of 1.6 million people died since TB in 2021 (with 187 000 people with HIV). Worldwide, TB is the 13th leading reason of death and the second leading infectious killer next to COVID-19 (above HIV/AIDS) [18].

According to WHO report in global tuberculosis report in 2021, globally the total number of people newly diagnosed with TB, and testified in general governments, released from 7.1 million in 2019 to 5.8 million in 2020. Sixteen countries accounted for 93% of this decrease with India (41%), Indonesia (14%) and the Philippines (12%) the worst affected [19].

Ethiopia is one of the 30 high TB; TB/HIV and MDR TB load countries with an expected TB occurrence of 164 per 100,000 population of drug vulnerable TB [20]. In addition, the problem of MDR-TB in Ethiopia is increasing. Pulmonary tuberculosis exchange inquiries are infrequently and incompatibly accepted in resource limited locations of low and middle income countries like Ethiopia [21]. Studies showed in Ethiopia on the occurrence and related factors of PTB amongst household contacts were very restricted. So, it is essential to classify the causes of which give to TB infection and general status of PTB amongst the household contacts of smears positive TB in the district [22].

The recommended plan to control TB in low income countries containing Ethiopia, where 95% of the TB cases occur, is to detect and promptly treat smear-positive cases [23]. It is known that late diagnosis results in more widespread disease, more problems and tips to a higher death [24]. To prevent extra spread of infection amongst families and communities, early discovery of PTB cases is very significant in the TB control. Now, screening for TB in Ethiopia is limited to patients presenting with cough lasting at least for 2 weeks [25].

The latest technique to detect tuberculosis is molecular examine with a shorter improvement time [26]. The molecular method is the GeneXpert MTB/RIF examine based on measureable real time PCR to detect the presence of M. tuberculosis in less than 2 hours. GeneXpert MTB/RIF has several benefits, and it uses a closed magnification system that possibly decreases cross reaction infection. It can be easily used in source restricted settings. So, it can replace the microscopic tests because it can detect bacteria in smear-negative. According to World Health Organization, GeneXpert can be used as a preliminary in patients with suspected pulmonary tuberculosis.

2.3. Pulmonary Tuberculosis Detection

The task of Pulmonary Tuberculosis disease detection is to uncover physical signs when a patient poses challenging difficulties by relating medical knowledge, judgment, and experience frequently with the aid of computer-based systems. Such systems are commonly known as computer-aided detection (CAD). A CAD (Computer-Aided Detection and Diagnosis) system is a class of computer systems its objective is to support in the detection and/or diagnosis of diseases [27]. The aim of CAD systems is to improve the accuracy of radiologists with a decrease of time in the analysis of images. CAD systems are divided into two sets: Computer-Aided Detection (CADe) systems and Computer-Aided Diagnosis (CADx) systems. CADe are systems geared for the place of lesions in medical images. Also, CADx systems execute the classification of the lesions [28].

Medical imaging methods can deliver thorough images of human structure. The character of medical imaging methods to generate visual images of the body's interior in order to make an accurate diagnosis [29]. Creating visual representations of the interior body is good because what the clinician can see of the patient's exterior is insufficient to make accurate diagnoses. Moreover, one of the main reasons for developing CAD is to help physicians to avoid medical faults because manual interpretation and analysis are tedious and prone to error. It is often required in addition to a medical history and physical examinations. Medical imaging techniques are therefore useful for the production of images of structures in the human body. These images reveal details about structural, functional organs and tissues. The most common medical imaging methods that have become increasingly important are radiography.

2.3.1. Radiography

Radiography (X-ray imaging) is the most widely used imaging technique due to its easy availability and comparatively low cost for the assessment of pulmonary Tuberculosis. The mandate for, and accessibility of, CXR images may be qualified to their cost-effectiveness and low radioactivity dose, joint with a reasonable sensitivity to a wide variety of pathologies [30]. The CXR is often the first imaging training learned and remains central to screening, diagnosis, and management of a broad range of conditions [31].

Chest X-rays might be classified into three main categories, according to the position and orientation of the patient relative to the X-ray source and detector panel: poster anterior, anteroposterior, lateral. The poster anterior (PA) and anteroposterior (AP) views are both measured as frontal, with the X-ray source positioned to the back or front of the patient respectively. The AP image is usually acquired from patients in the horizontal position, while the patient is usually standing straight for the PA image gaining. The lateral image is usually acquired in combination with a PA image, and projects the X-ray from one side of the patient to the other, normally from right to left [30].

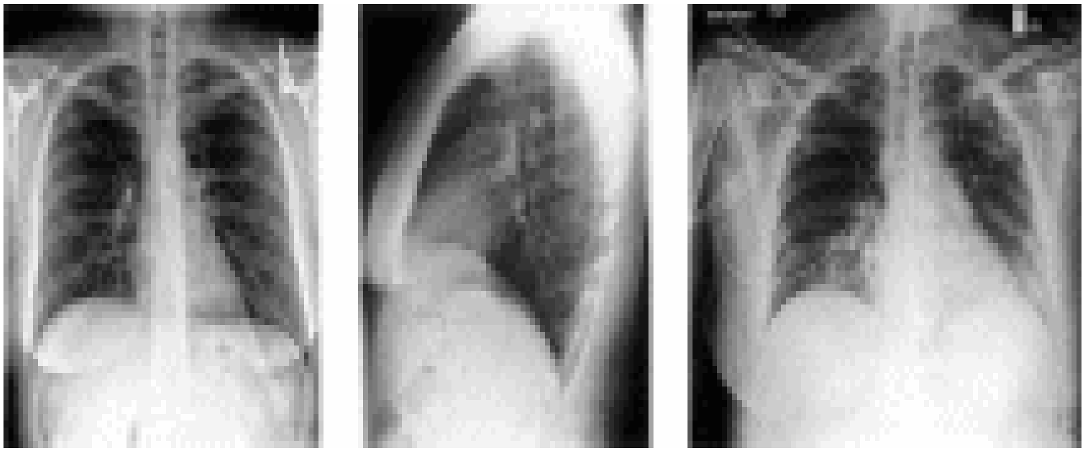

Figure 2.1. Posterior-anterior (PA), lateral and Anterior-posterior (AP) chest radiograph respectively.

The clarification of the chest radiograph can be challenging due to the superimposition of anatomical structures beside the forecast way [30]. This consequence can make it very challenging to detect abnormalities in specific places (for example, a nodule posterior to the heart in a frontal CXR), to detect small or subtle abnormalities, or to accurately distinguish between different pathological patterns. For these reasons, radiologists typically show high inter-observer variability in their analysis of CXR images [32, 33, 34].

2.4. Artificial Intelligence (AI)

Artificial intelligence (AI) is the replication of human intelligence functions by machines, particularly computer systems; expert systems, natural language processing, speech recognition, and machine vision are some examples of specific AI applications; AI requires a foundation of specialized hardware and software for the creation and training of machine learning algorithms; Python, R, Java, C++, and Julia all offer characteristics that are well-liked by AI engineers [35].

A vast volume of labeled training data is typically ingested by AI systems, which then examine the data for correlations and patterns before employing these patterns to forecast future states. By studying millions of instances, an image recognition tool can learn to recognize and describe objects in photographs, just as a chatbot that is given examples of text can learn to produce lifelike dialogues with people. Generative AI approaches are able to produce realistic text, graphics, music, and other media.

The potential for AI to alter how we live, work, and play makes it significant. Automation of human jobs including customer service, lead creation, fraud detection, and quality control has been successfully applied in business. AI is capable of several tasks considerably more effectively than humans. AI technologies frequently finish work fast and with very few mistakes, especially when it comes to repeated, detail-oriented activities, like reviewing a large number of legal papers to verify key fields are filled in correctly. AI can provide businesses with operational insights they may not have known about due to the enormous data sets it can process [35].

2.5. Overview of Machine Learning

Learning as a broad procedure is almost obtaining different, or changing existing, behaviors, values, awareness, skills, or preferences. Behaviorism, Cognitivism, Constructivism, Experientialism and Social Learning define the theory of personal learning, i.e. how humans learn [35]. Machines depend on data contrary to what arises obviously to persons: education from knowledge. Machine Learning (ML) is a type of artificial intelligence that allows computers to reflect and study on themselves. It is all about creating computers adjust their activities in order to improve the actions to achieve more accuracy, where accuracy is measured in terms of the quantity of times the selected actions results correctly. Machine learning is also known as a multi-disciplinary field having a wide-range of study areas supporting its reality.

Artificial Intelligence is a broad term for one part of computer science designed at machines to simulate human reasoning in real-life conditions. Machine Learning is a subset of an Artificial Intelligence focused on how to make computers study on their own without the need for hand coded guidelines. It allows computers to acquire from given samples. The more it studies, the enhanced it becomes. As the rapid development of technology continues, we start to see machines that can learn any intelligent task that a human can. A more recent survey on ML and observer common in machine learning applications in various fields such as data mining, face recognition, the Internet of Things, and the medical field at an unprecedented rate [36]. In this period, performing obstinately releasing its control in wide change of requests. Generally, an extremely productive change is taking place in the artificial intelligence background from efforts to form computer programs to solve health problems using automation and other progressive technology.

Basically, machine learning contains regression, classification, and clustering. Regression and classification tasks are supervised learning that a model is learned from the labeled examples in the training dataset. But, classification refers to the task where each instance/example is classified into one of the predefined categories. Whereas, Regression is used to forecast numerical data values for examples. Though, both classification and regression tasks build a model that can forecast the output of hidden instances by observing a set of class-labeled data. Whereas clustering is unsupervised learning because the input examples (training data) are not class labeled. This unsupervised learning algorithms are constructed on the intrinsic characters of the unlabeled data [37]. Though, despite much of the Machine Learning task research efforts up to now, there are still stimulating difficulties that need to be tackled. Some of these challenges include scalability, algorithm selection, and optimization techniques.

2.6. Entropy-based Feature Fusion strategy

Entropy-based feature fusion strategy is one the feature fusion strategy which is helpful to extract the relevant features of the image. In order for the suggested model to use as many features as feasible for the subsequent classification, this feature fusion combines feature vectors of training images retrieved from shared weight network layer with feature vectors made up of other numerical data. Numerical values are used to describe both the features derived from numerical data and the features extracted using image processing methods. $X_t = (X_{t1}, X_{t2} \ldots X_{tn}) R^n$ is the feature vector that was extracted from the image, and R_n stands for an n-dimensional vector.

It is a vector representation of the feature that the shared weight network layer extracted. Assume that the characteristics derived from numerical data are represented by the notation $X_e = (X_{e1}, X_{e2}, \ldots, X_{em})$ Rm, where Rm is an m-dimensional vector. Concatenating X_e and X_t results in the feature fusion, which is represented by the (m+n)-dimensional vector X_f. The following formula results in the feature fusion [38]:

$$\left(X_f = X_t \oplus X_e\right) = (X_{t1}, X_{t2}, \ldots., X_{tn}, X_{en}, X_{e2}, \ldots, X_{em}),\ X_f \in X_n + m \text{...............Eq(1).}$$

Where the elements $(X_{t1}, X_{t2}, \ldots, X_{tn})$ of X_t and the elements $(X_{e1}, X_{e2}, \ldots,$ xem) of Xe construct a new vector $(X_{t1}, X_{t2}, \ldots, X_{tn}, X_{e1}, X_{e2}, \ldots, X_{em})$ to express the fused feature vector X_f.

2.7. Deep Learning

Machine learning enables software applications to more accurately predict outcomes without being explicitly programmed. Machine learning algorithms use historical data as input to predict new output values. This approach has become significantly more effective as the amount of training data has increased. A subset of machine learning, deep learning is based on an understanding of the structure of the brain. Deep learning, utilizes the structure of artificial neural networks, is the basis for recent advances in AI networks [39].

A Deep learning (DL) is a subcategory of machine learning (ML) that give services in multiple layers to mine together with higher and lower-level information from input (i.e., images, numerical value, categorical values). The existing deep learning models are built on artificial neural networks (ANN), particularly convolutional neural networks (CNN), which may be combined with other deep learning models, with generative models, deep belief networks [39]. Individually apiece of level in deep learning acquires in changing its input data to subsequent layers while learning separate qualities of data. Example, the raw input may be a pixel matrix in image recognition applications, and the first layers may detect the image's edges. On the other hand, the second layer will construct and encode the nose and eyes, and the third layer may recognize the face by merging all of the information gathered from the previous two layers [40].

2.7.1. Deep Learning in Healthcare

This section seeks to measure the far-reaching influences of deep learning in healthcare. The combination of the rapid advances in the field of deep learning with socially impactful projects can influence the vast areas of our lives. One might be wondering where deep learning fits in the world of healthcare. Healthcare and medicine attitude to benefit greatly from deep learning because of the pure volume of data being generated as well as the increasing propagation of medical strategies and digital record systems.

This is strengthen wondering where deep learning fits in the world of healthcare. To attach the potential of deep learning to facilitate radiologists in accelerating the speed of the detection process. Its main focus was to investigate the applicability of deep learning concept to detect Pulmonary Tuberculosis in chest X-ray images. The outstanding lesson when analyzing all the research efforts is the wide variety of fields of deep learning is involved. Deep learning appears everywhere powering everything from virus-related screening to medication discovery [41]. There are, however, several obstacles to conveying advances in deep learning to the scientific setting, including the problem of simplification, restrictions in the accessible training data, lack of interpretability, and lack standards for reproducibility of publications [42].

2.7.2. Deep Learning with Computer Vision

A computer vision is an interdisciplinary area that includes all features of image and video processing that can be used in artificial visual systems for automatic scene analysis, interpretation, and understanding. The past history of computer vision can be drawn back to the 1960s unlike traditional image processing. The huge growth of cutting-edge approaches, algorithms, papers, and the broadly deliberated form of autonomous vehicles brought computer vision into public awareness [43]. Nowadays, developments in several fields of knowledge advantage since advances in computer vision. Within the model changes of artificial intelligence through different fields of science. For instance, computer vision is supporting autonomous automobiles to figure out where the extra cars and pedestrians are, so as to escape them. Nowadays, advances in many fields of knowledge benefit from advances in computer vision. The fundamental computer vision knowledge's have huge areas of uses such as remote sensing, medical image processing, precision agriculture, satellite image, defense, and various [44].

A Computer vision is therefore quite appealing in a widespread in different fields. The use of computer science to medicinal imaging represents one of the greatest promises of computer vision. As an example, imagine a radiologist who would be assisted by software allowing to detect anomalies on a radiograph that managed to extract relevant information about certain bone structures present in an x-ray image. The time saved for the radiologist and the gain in safety for the patient can be enormous. As a research area, there is a growing body of literature that recognizes the importance of computer vision in the field of different healthcare.

2.7.3. Conventional Neural Network (CNN)

Convolutional neural network (CNN) is one of the most popular and used of deep learning networks [45, 46]. The key benefit of CNN related to its antecedents is that it automatically identifies the significant features without any human control which made it the most used. Therefore, we have dug in deep with CNN by presenting the main components of it. Furthermore, we have elaborated in detail the most common CNN architectures, starting with the AlexNet network and ending with the High-Resolution network (HR.Net) [47].

A Conventional Neural Network is a special technique of image recognition, and a very real network with forward feedback [48]. The key objective of the CNN is to classify two-dimensional graphics. Its network arrangement is extremely invariant to translation, scaling, slanting or other forms of deformation. The reason why CNN has these characteristics is that CNN focuses on different kinds of features at each level [49]. It is a special family of deep learning algorithm and help for rich feature extractor that are used in image classification and object detection, object segmentation, image and video recognition, recommendation system and many other advanced tasks.

By means of conventional neural networks for images is not very practical since you have to deal with millions of multiplications of very large size matrices. One way to avoid computational complexity is to use 2D convolutions. The following figure 2.2 represents a typical CNN architecture.

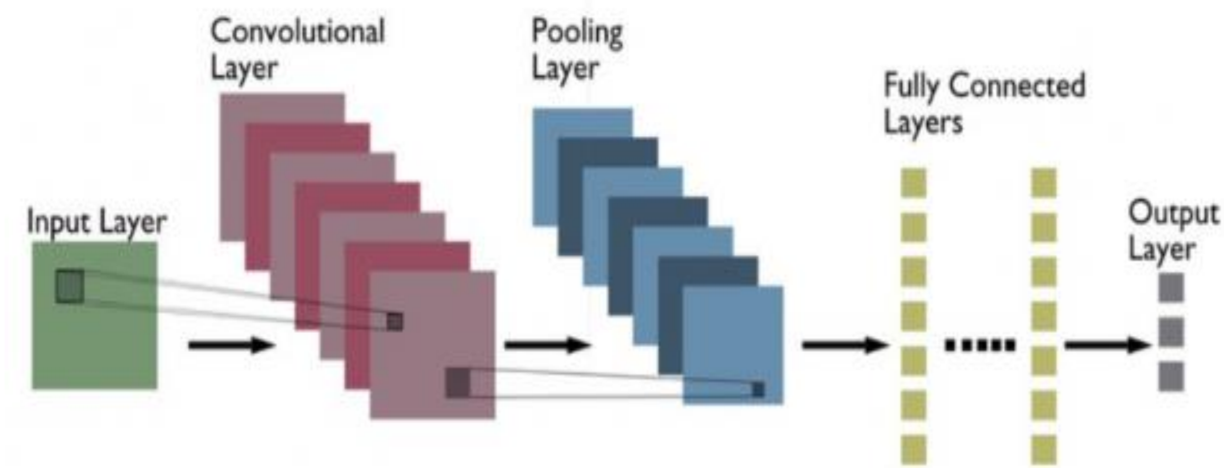

Figure 2.2. Basic Convolutional Neural Network Architecture [50].

- **Convolutional layer**: They are prepared since a set of filters (similarly known as kernels) that are functional to an input image [50]. The yield of the convolutional layer is a feature map, which is an illustration of the input image with the filters practical. Convolutional layers can be loaded to generate more difficult models, which can study more complicated features from images.
- **Pooling layer**: these layers are a type of convolutional layer used in deep learning. Pooling layers decrease the three-dimensional size of the input, making it easier to process and requiring less memory. It also supports to reduce the total of parameters and makes training faster. The two types of pooling are: max and average pooling. Max pooling takings the maximum value since each feature map, whereas, average pooling takes the average value. This layers are usually used next to convolutional layers in order to reduce the size of the input before it is served into a completely connected layer.
- **Fully connected layer**: It is one of the basic types of layers in convolutional neural

network. As we understand from its name, in this layer each neuron in a fully connected layer is fully connected to every other neuron in the previous layer. This layers are basically used towards the end of a CNN when the goal is to take the features learned by the pervious layers and use them to make predictions.

2.7.3.1. Types of CNN Architectures.

There are several types of architecture used in different purpose. Some of them are discussed as follow.

i. LeNet

LeNet is the first conventional neural network architecture. It was established in 1998 by Yann LeCun, Corinna Cortes, and Christopher Burges for handwritten digit recognition problems. It is one of the first successful CNN. It is also one the initial and most widely used CNN architecture and has been successfully applied to tasks such as handwritten digit recognition. The LeNet architecture contains of various conventional and pooling layers, followed through a fully connected layer. This model has five convolution layers followed by two fully connected layers. And it is the beginning of CNNs in deep learning for computer vision problems. Though, LeNet might not train well due to the vanishing gradients problem. To solve this issue, a shortcut connection layer known as max-pooling is used between convolutional layers to reduce the spatial size of images which helps prevent overfitting and allows CNNs to train more effectively. The figure below represents LeNet-5 architecture.

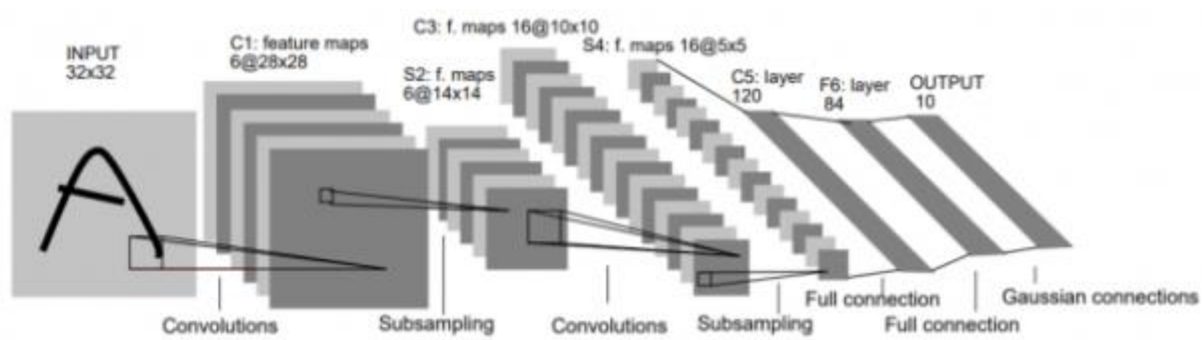

Figure 2.3. LeNet-5 Architecture [50]

This LeNet CNN is a modest yet great model that has remained used for several tasks like handwritten digit recognition, traffic sign recognition, and face detection.

ii. AlexNet

AlexNet network needed a much-related architecture to LeNet, however was deeper, bigger, and contained Convolutional Layers loaded on top of each other. It was the primary large-scale CNN and was used to win the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012. This architecture was considered to be used with large-scale image datasets and it accomplished state-of the-art results at the time of its publication.

iii. GoogLeNet

GoogLeNet is the CNN architecture used by Google to win ILSVRC 2014 classification task. It has been presented to have a particularly reduced error rate in assessment with earlier winners AlexNet and ZF-Net. It accomplishes deeper design by using a number of distinct methods, with 1×1 convolution and global average pooling. GoogleNet CNN architecture is computationally expensive. Practically applications of GoogLeNet CNN architecture contain Street View House Number (SVHN) digit recognition task, which is repeatedly used as a alternative for roadside object detection.

iv. VGGNet

It is the CNN architecture that was established by Karen Simonyan, Andrew Zisserman et al. at Oxford University. It is a 16 layer CNN containing up to 95 million parameters and trained on above one billion images (1000 classes). It need large input images of 224 x 224-pixel size that it has 4096 convolutional features. This type of CNN with large filter are expensive to train and require many data. So, GoogLeNet work better than VGGNet for most image classification tasks where input images have a size between 100 x 100-pixel and 350 x 350 pixels.

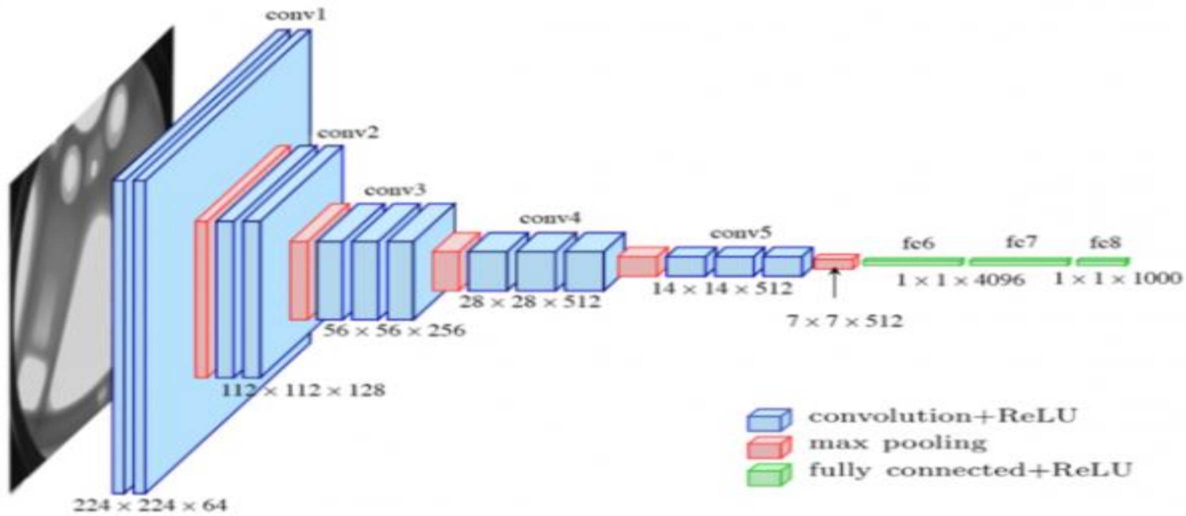

Figure 2.4. The VGG16 network architecture [45].

v. **MobileNetV2s**

These are a CNNs that can be appropriate on a mobile device to categorize images or sense objects by low latency. MobileNetV2s have been established by Andrew G Trillion et al. They are generally very small CNN architectures that makes them easy to run in real time using embedded devices like smartphones and drones. The architecture is also flexible so it has been tested on CNNs with 100-300 layers and it still works better than other architectures like VGGNet.

2.8. Related Work

There are different related research works are done using different techniques. In order to support our research work and strengthen our research gap, we reviewed some related research works. The study work on detection of tuberculosis disease using image processing technique [51]. In order to detect a disease, it takes the characteristics of the image that are used as a classification attribute extracted with KERAS. This research used three classification methods. These are support vector machines (SVMs) that is a supervised learning model with associated algorithms that analyze the data and recognize patterns. The second technique is based on logistic regression (LR), that is a classification machine learning algorithm that is used to predict the probability of a categorical dependent variable that is dichotomous; it contains data that can be classified in one of two possible categories (dead or alive, sick or healthy, yes or no, and so on). The last techniques is based on the nearest neighbors (KNN, K-Neighbors Classifier), that is an algorithm based on supervised type instances of machine learning. Finally, this study had better achievement of correctly detected a disease with accuracy of above 85%.

The research done on detecting tuberculosis in chest x-ray images using convolutional neural network [52]. The research was designed and deploy a consistent and well-organized system to categorize several TB manifestations. For this work, the study evaluates various CNN architectures and training parameters for TB X-ray image dataset and it uses the transfer learning methods in chest X-ray images. The transfer learning helps to store the learned knowledge from one domain and apply it to another different related domain to avoid training from the scratch to minimalize time. Total number of dataset used are 4701 images. Among those 453 of them are labeled as normal and 4248 are labeled as abnormal that contain various TB manifestations. It also used the AlexNet and GoogLeNet architecture for image classification. Finally, the model which is correctly detecting the disease with an accuracy of 85.68%. The research uses CNN feature selection techniques for the development of the model. However, the research did not explored the effect of entropy based-feature fusion strategy for the selection of features.

In COVID-19 detection from chest x-ray images using feature fusion and deep learning done for diagnosis of disease using chest X-ray images [53]. The main problem is the coronavirus feast quickly everywhere in the world, supposed to be initiated from Wuhan in China and is accountable for a huge number of deaths. Detection of the COVID-19 using chest x-ray image through accurate diagnosis, particularly for the case with no obvious symptoms, may decrease the patient's death rate. The feature extracted using histogram-oriented gradient (HOG) and convolutional neural network (CNN) from x-ray images where fused to develop the classification model through training by CNN (VGGNat). The HOG techniques was used to extract a feature vector from x-ray COVID-19 dataset and CNN method also extract another feature vector from the same images to fused both feature and used as an input to train classification model. In the feature extraction method used separately HOG techniques and CNN method accuracy are 87.34% and 93.64% respectively. The proposed feature fusion technique (98.36%) provided higher accuracy than the individual feature extraction methods, such as HOG (87.34%) or CNN (93.64%).

Si-Yuan Lu study on a classification method for brain MRI via MobileNetV2 and feedforward network with random weights [54]. The main problem addressed in the paper, the difficulty of diagnosis of brain disease to easily get treatment and control of the patient condition. There are also difficulty of verifying doctors' decision and improvement of diagnosis accuracy. Researcher have proposed their schemes for brain MRI classification mainly focused on binary classification using classifier training and feature learning. A pre-trained MobileNetV2 was exploited for image representation generation. The methods accomplished good classification performance in assessment with state-of-the art approaches.

In [55] paper work lung disease detection in chest x-ray images using transfer learning. The main problem is difficult to detect lung disease. The accurate detection is essential to achieve overall control of the disease and greatly increase the chance of medical treatment. The study propose two CNN model that rely on transfer learning to classify and detect the presence of pneumonia from a collection of chest x-ray images belong to four classes of normal, bacteria, pneumonia (viral), or COVID-19. It used VGGNet-16 and MobileNetV2 classification methods. The two proposed improved versions of VGGNet-16 and MobileNetV2 have been exposed to have near-perfect accuracy rates of 80% for VGGNet-16 and 82% for MobileNetV2.

CHAPTER THREE

METHODOLOGY

3.1. Introduction

The study's main objective is to create a MobileNetV2 model utilizing a Convolutional Neural Network to identify the disease of Pulmonary Tuberculosis. With the necessary ethical clearance, we included both normal and pulmonary tuberculosis-infected images in the collected dataset. The fundamental steps needed for model development are listed below. Image collection from medical institutions such primary care clinics, specialist hospitals, high clinics, and medium clinics; image preprocessing; segmentation; feature extraction; and, finally, image detection. Different techniques can be applied to detect disease on an image. We used MobileNetV2 which is one of CNN architecture and an efficient method for studying image detection.

3.2. System architecture

In order to learn characteristics of pulmonary tuberculosis, the proposed model was developed using the MobileNetV2 network. Based on how well the model's value hyperparameters worked, the hyperparameters were chosen. Segmentation was used in order to separate the Region Of Interest (ROI) from the other backgrounds. So that the developed TBDMobileNetV2 model can detect the disease associated with Pulmonary Tuberculosis in the provided image. For the analysis of our data, we used the following process.

First, our model tried to preprocess the input labeled image such as denoise the image, image scaling to equal sized image, convert RGB image to grayscale, segment to boost the quality of the image for subsequent processing. The second phase of our model attempted to use the Entropy based feature fusion strategy. The third phase attempted to split the dataset into training, validation, and testing. The fourth phase attempted to build base-pretrained MobileNetV2 model. The fifth phase attempted on modifying the base-pretrained MobileNetV2 model. The sixth phase attempted on train and evaluate MobileNetV2 model with our dataset. The final phase attempted on testing the developed model with testing dataset. So that infected and normal results can be obtained.

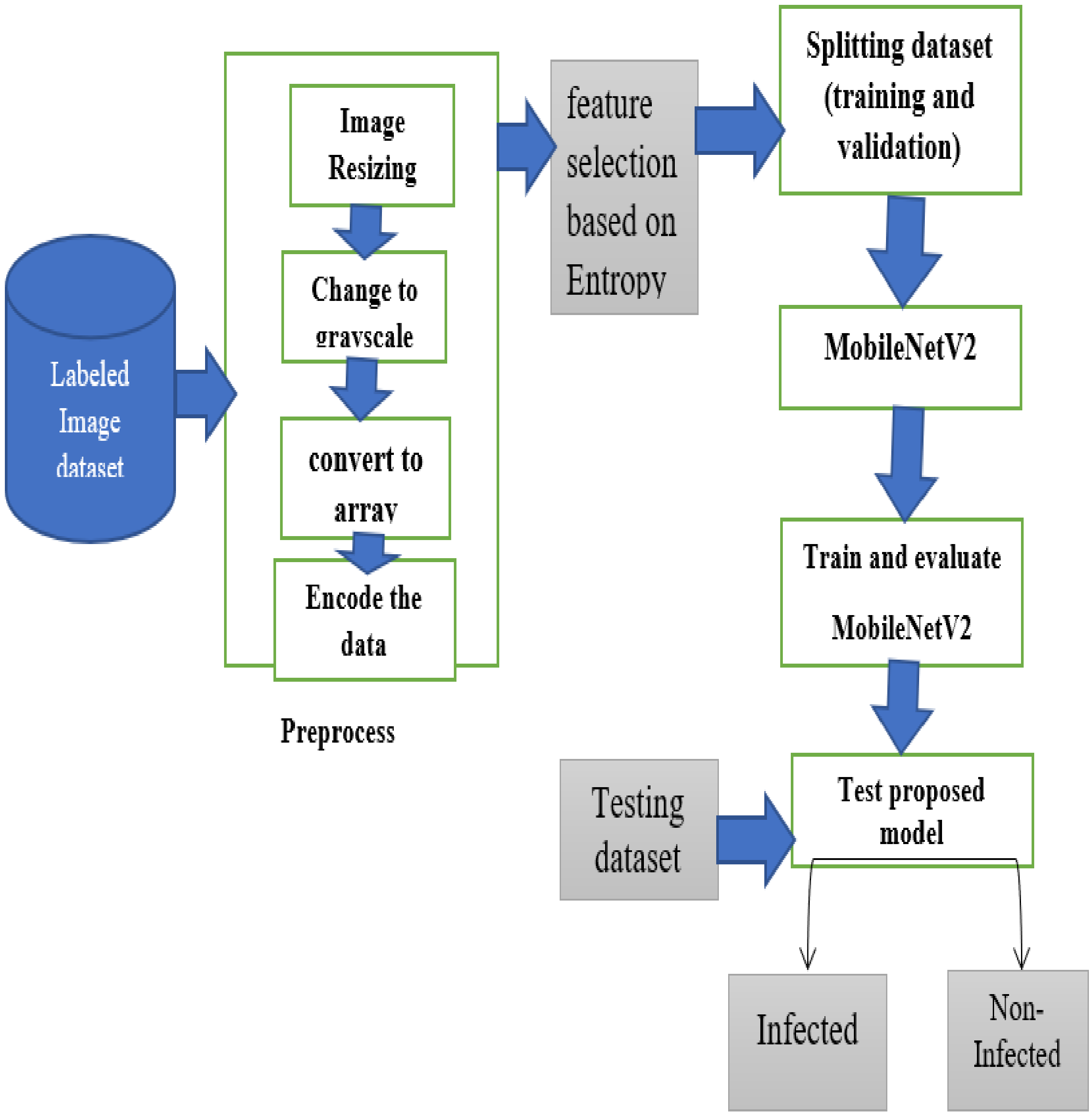

Figure 3.1. Proposed TBDMobileNetV2 architecture for automatic detection of Pulmonary Tuberculosis disease.

3.2.1. Preprocessing

The given input image must be processed in order to make it ready for our MobileNetV2 model to use for training and testing. This procedure is known as image preprocessing. In order to extract the significant features from an image, significant operations are carried out on the image. The result is an improved and high-quality image.

The preprocessing phase includes image resizing to (224 x 224), background noise removal, and conversion of the image into arrays using NumPy and Kera's. The recent models developed [56] took an image of size (224 x 224) and our MobileNetV2 model also takes an image of size (224 x 224) that have improved accuracy.

We used OpenCV (Open-source Computer Vision) for image resizing and conversion of an image into arrays with NumPy. The original input image is RGB that was captured by x-ray which is a combination of primary colors (Red, Green, Blue). Because of the values of RGB color is between 0 and 255; it is a challenge to implement with RGB color images. So that it is required to convert RGB color image into a range of 0 and 1which makes easy to implement in many applications. OpenCV bicubic interpolation can perform many tasks for image data preprocessing. Some important information will probably lose due to interpolation methods are approximation algorithms. So that interpolation works for estimation of unknown values with known data. So that the more known the previous value the more accurate prediction of pixels.

We decided the shape of the input image in the preprocessing hence, the ordering of the input shape affect the developed model. So that if the input shape is ordered by channels first, the depth of the volume must be placed first; if the input shape is ordered by channels last, the depth of the input volume should be place next to the height and the width at the third coordinate. However, our MobileNetV2 model works without the consideration of the order of the input shape.

The image can be divided into pertinent parts to help distinguish between the background and foreground, and the region of interest (ROI) can then be retrieved from the backdrop using segmentation from the preprocessing library. Pixel-based images typically have identical color, intensity, and texture patterns throughout. However, even though they share the same properties, the different regions of pixels differ dramatically. Therefore, segmentation can lessen loss while simultaneously increasing the accuracy of object detection in an image. In the suggested

MobileNetV2 model, we employed segmentation by classification, where the goal of classification is to divide data points into a number of groups based on similar data points are grouped in the same group.

3.2.2. Entropy-based feature fusion strategy

Entropy measures the degree of uncertainty or randomness in a collected dataset and assesses the information content of the picture. The quality of an Pulmonary Tuberculosis infected and normal image improves with information content. After the quantization phase, entropy encoding, a form of lossless compression, is applied to the image. With the least amount of memory needed for storage or transmission, it makes it possible to depict dataset images more effectively. Entropy in our dataset is referred to as the corresponding intensities to which individual pixels can adapt. The entropy value is utilized in the quantitative analysis and evaluation of picture details since it allows for a more accurate comparison of the image details [57]. The input preprocessed dataset texture can be described using entropy, a statistical measure of randomness. Entropy is a measure of the unpredictability or disorder of the data that machine learning algorithms are analyzing. Entropy is the machine learning metric that evaluates the unpredictability or impurity in the system, to put it another way.

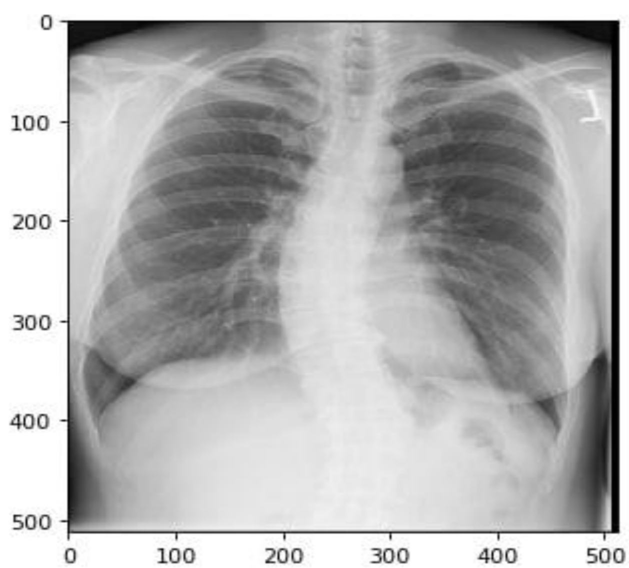

Figure 3.2: Normal image of the the dataset

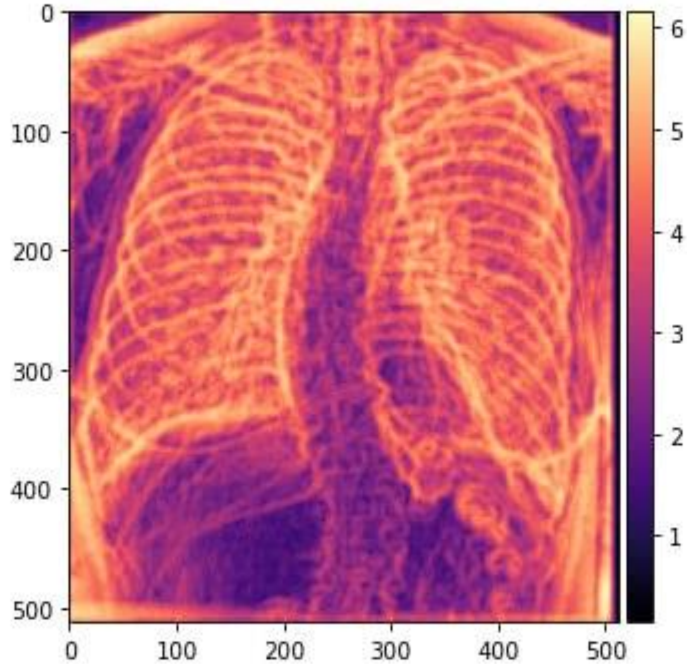

Figure 3.3: Image with entropy-based feature selection

3.2.3. Splitting the dataset

The collected dataset is classified into training, validation, and testing dataset. Training data is the type of data on which MobileNetV2 model is built. Users supply the model with input data that corresponds to the expected output. The model repeatedly evaluates the data to learn more about its behavior and tune itself to serve its intended purpose. Validation data is present during training. Validation data inserts new data into the model that has not been previously evaluated. Validation data provides a first test against unseen data, allowing users to assess how well the model is making predictions based on new data. Not all data users use validation data, but it can provide useful information for tuning hyperparameters that affect how the model evaluates the data. Test data is used after the model is built to reconfirm that it can make accurate predictions. The training and validation data contain labels to monitor the model's performance metrics, the test data should be labeled. Test data provides a final, real-world validation of an unseen data set that confirms that our MobileNet model is training effectively.

The validation dataset is a sample of data retained during model training and used to estimate model features while optimizing model hyperparameters. The validation dataset is different than the test dataset. The test dataset is also retained from model training, but is instead used to unbiasedly estimate the features of the final adjusted model when comparing or selecting between final models.

We collected Pulmonary Tuberculosis disease images for the training, validation and testing of the model. So that 900 x-ray images collected from Jimma University medical center for model development. For training, validation, and testing the dataset split ratio is 80% for training and 10% for validation and 10% testing. The train-test split performed after the image preprocessing and entropy-based feature selection technique. The various parameters used in the model is helpful to improve the system performance with the training dataset. The later testing dataset is used to evaluate the performance of the system.

Phase 1: Training phase: The most crucial phase of TBDMobileNetV2 is training. To detect patterns and generate predictions, we feed our TBDMobileNetV2 model and the prepared data during training. As a result, the model gains knowledge from the data to complete the given task. The model improves in prediction over time with training. The training data is the subset of original data that is used to train the TBDMobileNetV2 model.,

Phase 2: Validation phase: a sample of data withheld from the training of our model is referred to as a validation dataset and is used to measure the model's skill while adjusting its hyperparameters. Model validation is a step that comes after model training and involves comparing the trained model to a test set of data. The machine can determine whether or not it is performing well at accurately forecasting future data-out-of-sample during the machine validation phase by creating feedback loops.

Phase 3: Testing phase: Testing data is used to check the accuracy of the model. We need unknown data to test our TBDMobileNetV2 model after it has been constructed (using our training data). We can use this data, which is referred to as testing data, to assess the effectiveness and development of the training of our algorithms and to modify or optimize them for better outcomes. The main goal of the test phase is to establish the readiness for deployment of the model that was created, acquired, and preliminary tested during the development phase.

Images that are different from the training datasets fed into the learned model to observe how well the developed models responds to different datasets. After building our model, testing datasets are fed into the developed model for the evaluation of the performance.

3.2.4. Build Base-Pretrained MobileNetV2

An improved version of MobilenetV1 is MobilenetV2. This increases its effectiveness and potency. Because of the simplified and smaller models, MobileNetV2 models run more quickly. The second version of the Mobilenet models is known as MobilenetV2. The deep neural network's parameter count is noticeably smaller. Deep neural networks as a result get lighter. Due to its small weight, it works best in mobile and embedded systems. We used MobileNetV2 version for the development of the model. The small number of parameters in deep neural networks are employed to build our model [58].

3.2.5. Modify MobileNetV2 model

MobileNetV2 has an output layer consisting of 1000 neurons corresponding to 1000 trained categories. However, we want to separate the output into two categories for detection. So that the model is modified based on the expected output and the nature of the collected dataset. A dataset with TB infected and non-infected is our datasets for detection. A type of convolutional neural network models known as MobileNetV2s are compact, low-latency, and low-power models that can be utilized for classification, detection, and other typical tasks. So that we used MobileNetV2 CNN architecture for our Pulmonary Tuberculosis model development and for the detection of the disease. These are regarded as excellent deep learning models for usage on mobile devices because to their modest size. A potent detection model called MobileNetV2 can achieve cutting-edge performance by using transfer learning for our model development. A novel type of convolutional layer known as Depth wise separable convolution is used in the considerably quicker and smaller CNN architecture known as MobileNet [58].

3.2.6. Train MobileNetV2 model

The annotated normal and Tuberculosis disease-infected images are used for the training and evaluation of the model. Our TBDMobileNetV2 model developed with the modified MobileNetV2 and the prepared dataset for training. So that the TBDMobileNetV2 model is helpful to detect the Pulmonary Tuberculosis disease after it obtained the required experience from training. The training process used the annotated training dataset to acquire the necessary learning experience and to obtain the desired model.

3.2.7. Test proposed model

Testing of the model is required to evaluate the performance of the proposed MobileNet model. The total labeled dataset collected from Jimma University medical center classified into training, validation, and testing dataset. The testing dataset is helpful to know how much the developed model performs based on the learned experience from the training dataset. So that testing of the developed model refers to the task of extracting information based on the learned experience for the given Pulmonary Tuberculosis infected and normal image.

The input image detected and returned as Pulmonary Tuberculosis disease infected or not based on the nature of the image and the learned experience of the model. So that the result of detection for the image is infected with Pulmonary Tuberculosis or non-infected. Infected image of the result of testing with the proposed model is returned based on the performance of the developed model. Infected image is one of the outputs returned from the developed model. Non-infected images are the other results of the developed model based on the training dataset and the model.

CHAPTER FOUR

EXPERIMENTAL RESULTS AND DISCUSSION

4.1.Introduction

In this part the experimental assessment findings of the proposed model that has been undertaken for the diagnosis of Pulmonary Tuberculosis disease. The findings of the experimental assessment demonstrate that our model detects the disease of pulmonary tuberculosis with improved accuracy. The accuracy returned using Pulmonary Tuberculosis MobileNetV2 (TBDMobileNetV2). The experimental assessments allow for a verification of the developed model functionality, effectiveness, and accuracy. To this end, loss/accuracy curve, and assessment metrics outcomes is used to highlight the accuracy of our model.

4.2.Dataset preparation

We collected the X-ray image dataset from Jimma University Medical center. When we collect dataset, some challenges are faced. Some of them are, unavailability of well captured x-ray image, difficulty of communicating with health sector to get x-ray dataset, other related problems are challenging for get our dataset easily. The domain experts annotated the collected X-ray image dataset into Pulmonary tuberculosis infected and non-infected. After we collected the dataset from the specified medical center, the collected data fed to TBDMobileNetV2 for model development. Our image is in PNG format since it offers excellent compression and is a lossless file type. So that PNG has excellent compression as a result. Most papers worked on image processing followed 80% to 10% to 10% for training, validation, and testing respectively. From 900 image data 80% of the dataset used for training 10% for validations and 10% used for testing.

Table 4.1: Dataset preparation

Class	Original collected image	Image format
Pulmonary Tuberculosis Infected	400	PNG
Normal	500	PNG

The figure 4.1 depicts normal and pulmonary tuberculosis infected image. We observed the difference between the two infected and normal images that we collected from Jimma University medical center and annotated by domain experts.

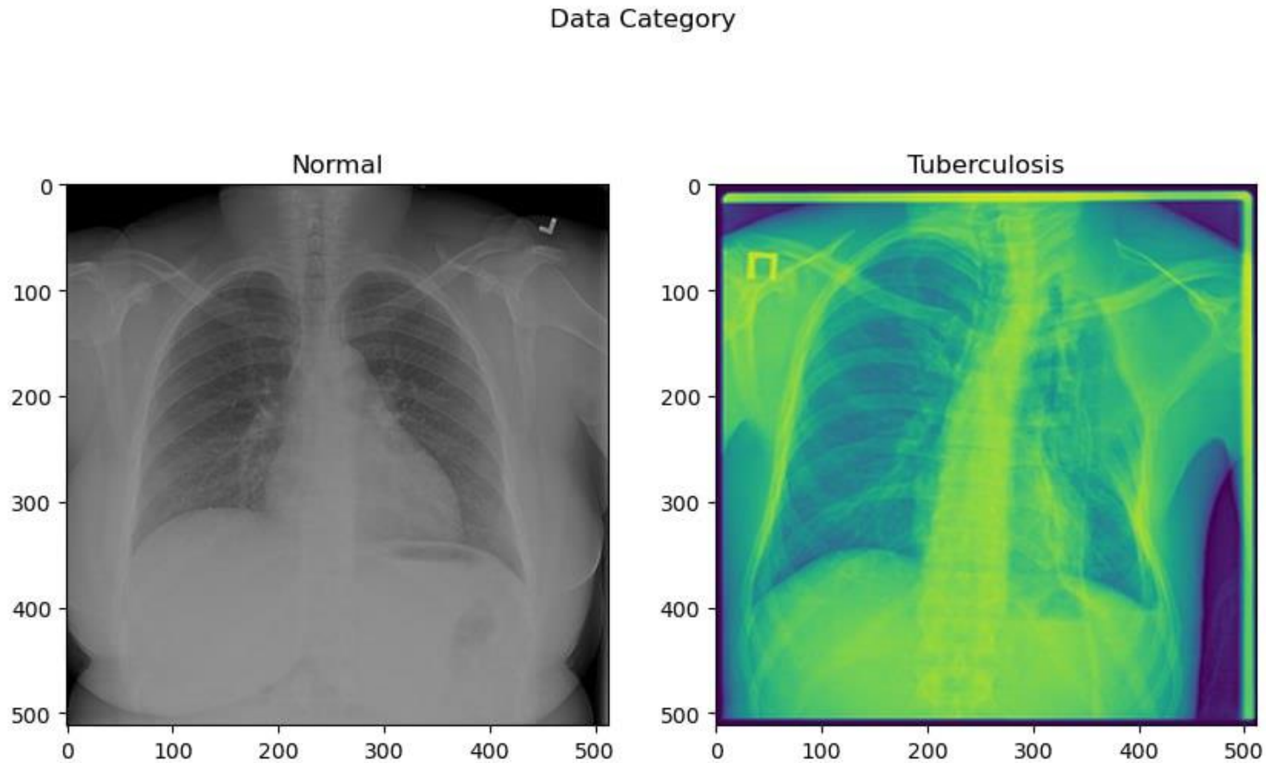

Figure 4.1: Normal and Tuberculosis infected image

4.3. Implementation environment

Several open-source libraries are used in the implementation of the pulmonary tuberculosis illness detection model. For this work with thousands of open-source packages and libraries, we chose Anaconda3 2019 64 bit. Anaconda is designed to make package administration and development easier and it is a distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analysis, etc.). We utilized the Jupyter Notebook editor, which is a free and open-source scientific environment created by and for scientists, engineers, and data analysts and developed in Python specifically for Python.

TensorFlow was used for numerical computation and large-scale machine learning; Numpy was used for working with arrays; Pandas was used for data analysis and manipulation; Matplotlib was used to create graphics; Python programming language, version 3.7, was used for scientific computing; Scikit-Learn, a machine learning library for Python, was used. We used the free source Keras (with TensorFlow as the backend) and Tensor Flow library. The CNN library Keras is open-source and written in Python.

It is easily running on top of Tensor Flow. TensorFlow is a representative math library and is also used for scientific computing tenders such as neural networks [59].

Another free software library for deep learning and machine learning dataflow programming is called Tensor Flow. The Google brain team developed it so that Google users throughout the world may utilize and analyze it [59]. It is a foundational package that enables the creation of deep learning models instantly. Tensors are the typical data representation in deep learning. An array with several dimensions called a tensor is an extension of matrices and vectors. A Python linking library called OpenCV-Python is used to solve computer vision issues. This library contains the CV2 module, which is used to read images, resize images, convert RGB to grayscale, etc for efficient processing of the model.

The main attempt of this work is detection of the use of MobileNetV2 model with entropy feature fusion strategy for the detection of Pulmonary Tuberculosis disease. We also used VGGNet, AlexNet model in addition to our attempt for comparison of results.

```
In [23]: base_model.summary()

Model: "mobilenetv2_1.00_224"
__________________________________________________________________________________________________
 Layer (type)                   Output Shape         Param #     Connected to
==================================================================================================
 input_2 (InputLayer)           [(None, None, None,  0           []
                                 3)]

 Conv1 (Conv2D)                 (None, None, None,   864         ['input_2[0][0]']
                                32)

 bn_Conv1 (BatchNormalization)  (None, None, None,   128         ['Conv1[0][0]']
                                32)

 Conv1_relu (ReLU)              (None, None, None,   0           ['bn_Conv1[0][0]']
                                32)

 expanded_conv_depthwise (Depth  (None, None, None,   288        ['Conv1_relu[0][0]']
 wiseConv2D)                    32)
```

```
 block_16_project_BN (BatchNorm  (None, None, None,   1280       ['block_16_project[0][0]']
 alization)                     320)

 Conv_1 (Conv2D)                (None, None, None,   409600      ['block_16_project_BN[0][0]']
                                1280)

 Conv_1_bn (BatchNormalization)  (None, None, None,   5120       ['Conv_1[0][0]']
                                1280)

 out_relu (ReLU)                (None, None, None,   0           ['Conv_1_bn[0][0]']
                                1280)

==================================================================================================
Total params: 2,257,984
Trainable params: 2,223,872
Non-trainable params: 34,112
```

Figure 4.2: Model summary for the model TBDMobileNetV2

4.4. Result of new proposed TBDMobileNetV2 Model during validation phase

As demonstrated in figure 4.3 below, our model, TBDMobileNetV2, achieves 95.93% training accuracy and 96.11% validation accuracy. When the model is originally trained using batch normalization and dropout, this detection accuracy is attained. This outcome was likewise attained by resizing the input image to 224*224 pixels. Our loss curve is more stable when batch normalization is used in our network after each activation layer.

The preparation of TBDMobileNetV2 model use various parameters that influence the model performance. Therefore, we tried to optimize the value of the parameters to come up with the best performance of the model. We used a batch size of 32 hence, when the batch size minimized, the training time of the epoch becomes increased. Additionally, we used epoch 50, optimizer Adam for the TBD model development. The following figure depicts the result summary of TBDMobileNetV2 model with our dataset.

```
22/22 [==============================] - 9s 421ms/step - loss: 0.1011 - accuracy: 0.9622 - val_loss: 0.0717 - val_accuracy: 0.9722
Epoch 38/50
22/22 [==============================] - 9s 406ms/step - loss: 0.1028 - accuracy: 0.9651 - val_loss: 0.0819 - val_accuracy: 0.9722
Epoch 39/50
22/22 [==============================] - 10s 443ms/step - loss: 0.0997 - accuracy: 0.9608 - val_loss: 0.0723 - val_accuracy: 0.9722
Epoch 40/50
22/22 [==============================] - 10s 433ms/step - loss: 0.1202 - accuracy: 0.9578 - val_loss: 0.0977 - val_accuracy: 0.9611
Epoch 41/50
22/22 [==============================] - 9s 403ms/step - loss: 0.1141 - accuracy: 0.9622 - val_loss: 0.0821 - val_accuracy: 0.9667
Epoch 42/50
22/22 [==============================] - 9s 403ms/step - loss: 0.1009 - accuracy: 0.9695 - val_loss: 0.0591 - val_accuracy: 0.9833
Epoch 43/50
22/22 [==============================] - 10s 456ms/step - loss: 0.1282 - accuracy: 0.9477 - val_loss: 0.1316 - val_accuracy: 0.9611
Epoch 44/50
22/22 [==============================] - 10s 455ms/step - loss: 0.0877 - accuracy: 0.9782 - val_loss: 0.0625 - val_accuracy: 0.9778
Epoch 45/50
22/22 [==============================] - 9s 398ms/step - loss: 0.0917 - accuracy: 0.9680 - val_loss: 0.0639 - val_accuracy: 0.9778
Epoch 46/50
22/22 [==============================] - 9s 400ms/step - loss: 0.1012 - accuracy: 0.9564 - val_loss: 0.0682 - val_accuracy: 0.9778
Epoch 47/50
22/22 [==============================] - 10s 443ms/step - loss: 0.0887 - accuracy: 0.9673 - val_loss: 0.0668 - val_accuracy: 0.9778
Epoch 48/50
22/22 [==============================] - 9s 388ms/step - loss: 0.0929 - accuracy: 0.9680 - val_loss: 0.0654 - val_accuracy: 0.9778
Epoch 49/50
22/22 [==============================] - 9s 406ms/step - loss: 0.0951 - accuracy: 0.9535 - val_loss: 0.1082 - val_accuracy: 0.9500
Epoch 50/50
22/22 [==============================] - 10s 450ms/step - loss: 0.1135 - accuracy: 0.9593 - val_loss: 0.0968 - val_accuracy: 0.9611
```

```
[INFO] evaluating network...
6/6 [==============================] - 4s 667ms/step
              precision    recall  f1-score   support

      Normal       0.94      0.99      0.97       100
Tuberculosis       0.99      0.93      0.95        80

    accuracy                           0.96       180
   macro avg       0.96      0.96      0.96       180
weighted avg       0.96      0.96      0.96       180
```

Figure 4.3: Accuracy and loss summary of validation and training TBDMobileNetV2 model

As demonstrated on figure 4.3, our model, TBDMobileNetV2, achieves 95.93% training accuracy and 96.11% validation accuracy. When the model is originally trained using batch normalization and dropout, this detection accuracy is obtained. This outcome was likewise attained by resizing the input image to 224*224 pixels. Our loss curve is more stable when batch normalization is used in our network after each activation layer [59].

We tried different activation functions like sigmoid, Tanager, and ReLu and among the available activation function Sigmoid outperforms the other. Based on the model result, the above figure 4.3 depicts the evaluation of the model performance using different evaluation matrices such as precision, recall, and F1-score.

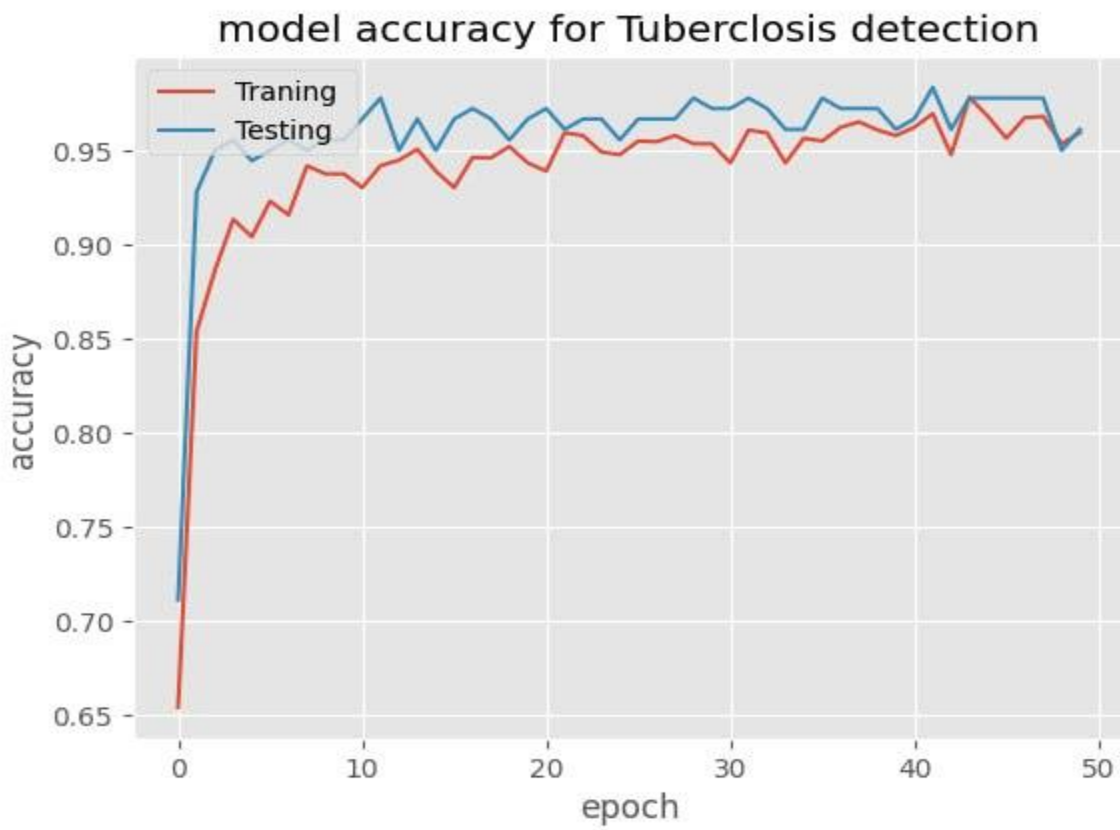

Figure 4.4: Accuracy of TBDMobileNetV2 model during training and validation

One of the most common curves used to understand neural network progress is the accuracy curve hence, it describes the difference between training and validation curve. So that the difference between training and validation accuracy can be used to detect overfitting. We observed a gap between training and validation accuracy happened from the first epoch to around 10 which indicates the presence of overfitting. At the starting epoch of the model the gap between the validation and testing accuracy is expected. From epoch 10 to the end up and down of the curve observed with the noise movement of the training curve as shown on above figure. Jumping up and down would be possible if the testing dataset contained fewer samples than the training dataset [1].

A model with strong generalization capabilities can perform well on both test and training sets of data. On the other side, overfitting also happens when a model succeeds on test data from epoch 1 to epoch 20, but fails on training data, creating a generalization gap. Data augmentation, regularization, and dropout are often used techniques to avoid overfitting when developing neural networks. Finally, even good models will likely still exhibit some overfitting. If we attempt to close this gap between training and test errors, we will likely cripple our model and prevent it from learning at all [1].

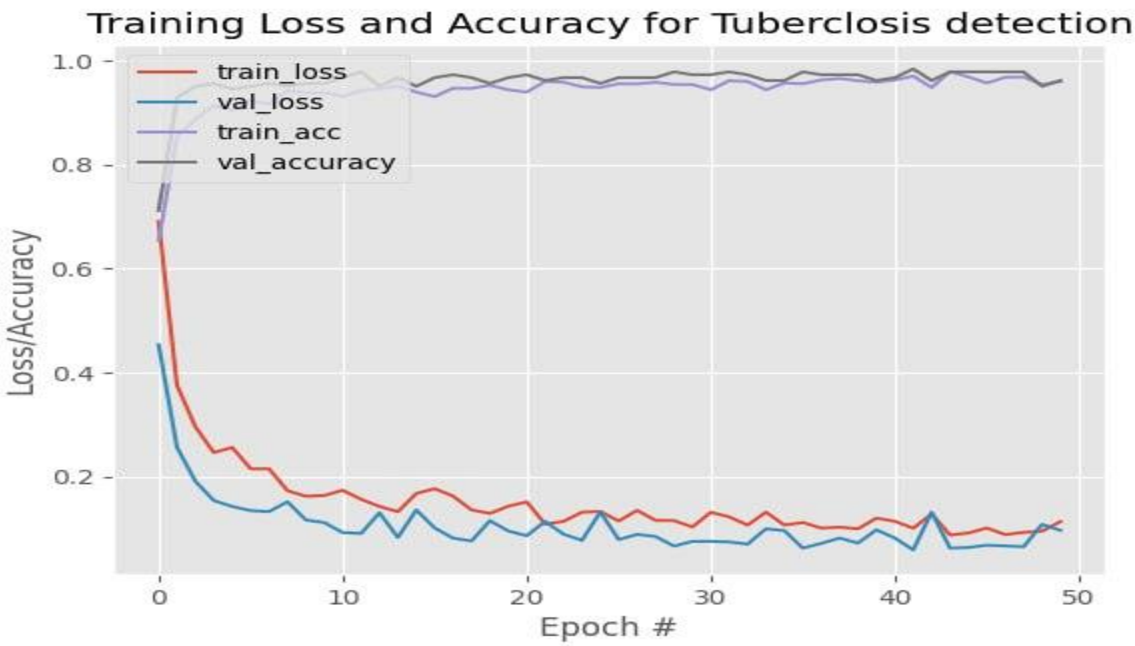

Figure 4.5: Pulmonary Tuberculosis detection loss and accuracy in validation and training phase

The validation loss shows correctly and incorrectly classified samples. Loss is going up for incorrectly classified samples and going down for correctly classified samples. A loss is the penalty for a false estimation. In other words, loss is a statistic that indicates how accurately the model predicted a particular input. The loss curve is close to zero if the model's forecast is accurate; otherwise, it is greater. Typically, an epoch's earliest batches suffer from higher losses than its last epoch. From the start of the epoch until about epoch 10, as seen in Figure 4.5 above, the loss of training is noticeably greater than the loss of validation. This happened as a result of regularization approaches being applied during the training phase as opposed to the testing phase. These regularization strategies are crucial for training accuracy to surpass testing accuracy [1].

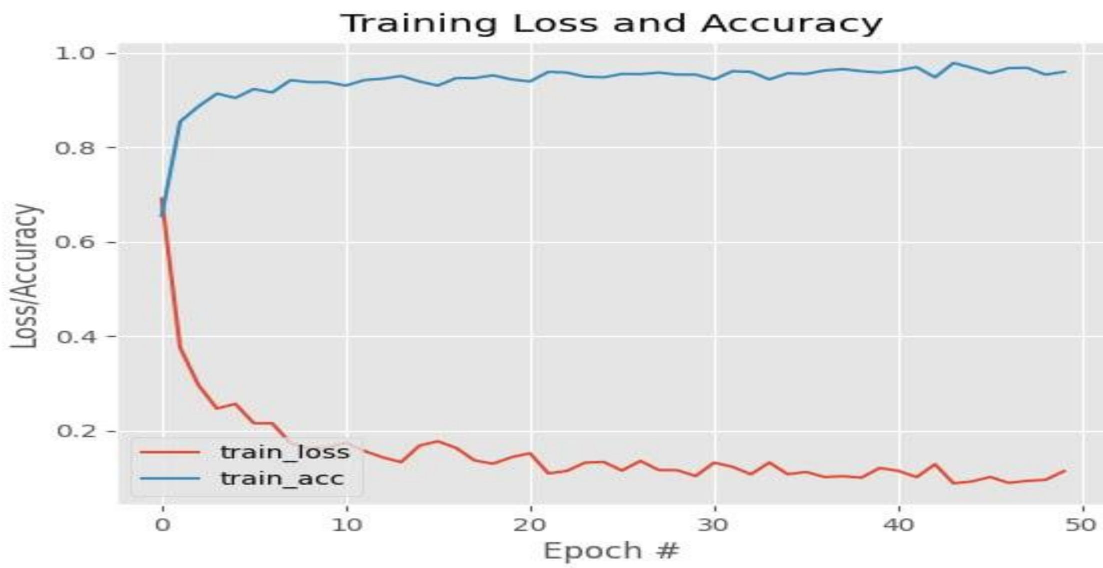

Figure 4.6: Pulmonary Tuberculosis training accuracy and loss graph

The diagram below shows the confusion matrix result of TBDMobileNetV2. We tested the developed model with unseen dataset. Based on the following confusion matrix figure, from 180 test images 175 of the unseen dataset detected correctly and the remaining 5 images detected incorrectly with TBDMobileNetV2 model.

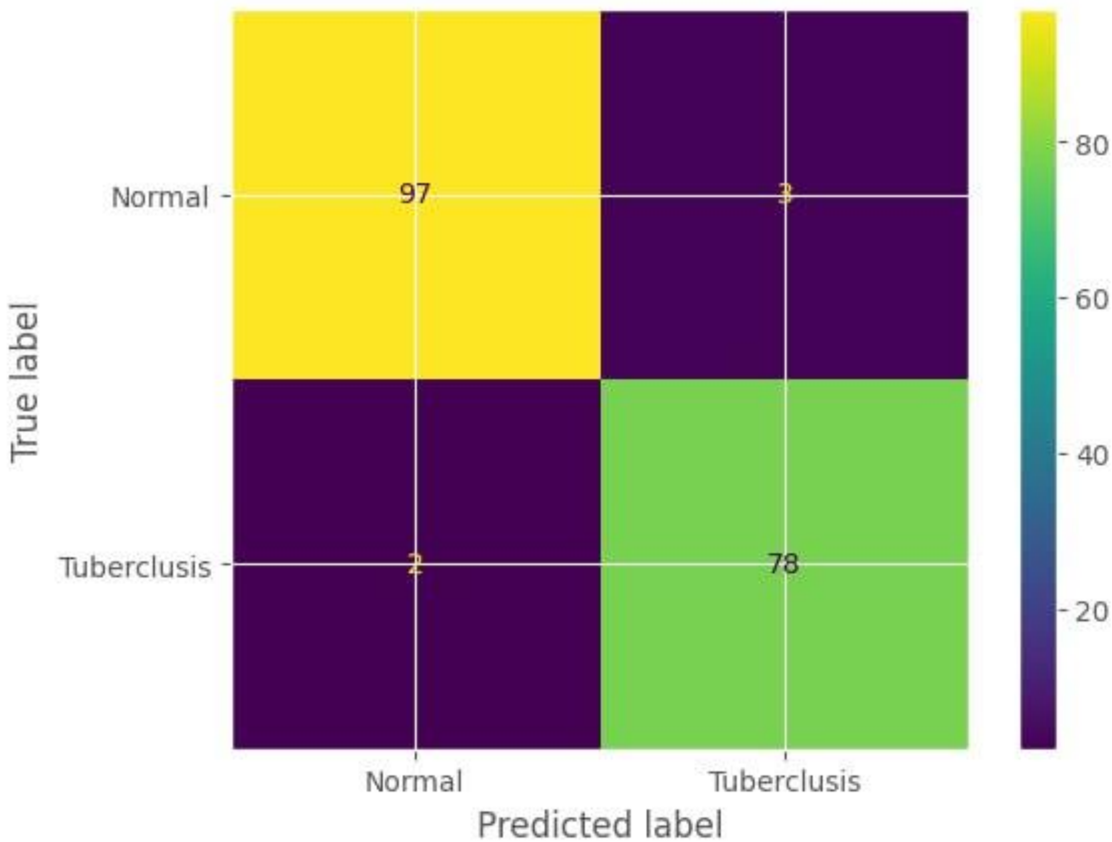

Figure 4.7: Confusion matrix result for TBDMobileNetV2 model

4.5. Parameters involved in the proposed model

We used various hyperparameters to select the best value of the hyperparameters and to develop the outperformed model. So that we changed the hyperparameter values till we obtained the outperformed model. Some hyper-parameters, i.e., variables with the power to determine the network structure for optimizing the training result, are considered while training the model. The following are the major parameters that must be considered when constructing the TBDMobileNetV2 model.

- **Number of epochs**: how many times the system reads the given dataset.
- **Number of batch size:** specific parts of the dataset go to be seen by the model at once.
- **Number of iterations:** The number of batches required to calculate the dataset in one epoch.
- **Pooling size:** reduces the original input image or feature maps during the convolution operation.
- **Stride size:** by what distance does the kernel move forward and downward to convolve with the images.

- **Zero padding:** Adding zero from all sides of the image in a matrix to protect the size of the original input image or feature map.
- **Kernel size**: The number of filters applied to the image to generate feature maps.
- **Dropout rate**: techniques to regularize the training to overcome over fitting in the fully connected layer.

4.6. Hyperparameters

When the train model has finished running, our classifier has 98.49 percent accuracy, which means that our model can detect 98 percent of diseases. The following table provide information like the number of epochs, optimization algorithms, image size, and so forth. The best results were finally obtained with a dropout of 0.49 and a learning rate of 0.001, with accuracy of 98.49 percent. The following table shows that, the summary of the values of various hyper parameters used in the TBDMobileNetV2 neural network model and scores higher accuracy during experimentations.

The network structure for a training result that is optimized is determined by hyperparameters. By selecting the value of the batch size and other parameters, these groupings of hyperparameters are established for the network based on several training data sets. These parameters control how many times the model reads the entire data set, as well as the batch size, which decides which portions of all datasets are highlighted at once. The additional groups of hyperparameters are as follows.

Table 4.2: TBDMobileNetV2 hyperparameters

Hyper parameter	**Value**
Activation	**Sigmoid**
Striding	**1**
Padding	**Same**
Kernel size	**3x3**
Batch size	**32**
Learning rate	**0.001**
Dropout	**0.49**

Epoch	**20**
Optimizer	**Adam**

Optimizers

Optimizers are techniques or algorithms for decreasing loss by altering a neural network's properties, such as weights and learning rate. Using Adam as an optimizer, we created a TBDMobileNetV2 model. The model parameters are updated because it is crucial for frequent updates of the model parameters and because it converges more quickly once the loss for each training example has been computed. Because model parameters are continuously updated, loss functions exhibit significant variances and volatility at various volumes.

Learning rate: In training the CNN model, the learning rate is a crucial factor. The step size that should be properly taken into account during training is the learning rate, which expedites the training process. On the other hand, it can be tricky to choose the learning rate's value. For instance, if you increase the learning rate, the network can start to considerably differ rather than integrate. On the other hand, if you lower the learning rate, it will take longer for the network to converge. In addition, it is susceptible to becoming caught in a local minimum. Typically, this issue is resolved by slowing the rate of learning while under training [60].

A generalization gap in the learning process is anticipated when a model's performance on training data differs from its performance on brand-new or untested test data. As depicted in figure 4.3 above, the generalization gap between the training and validation accuracy curves increased dramatically between the first epoch and about epoch 2.5, however it decreased significantly between the start of epoch 2.5 and epoch 5. The accuracy of validation degrades from epoch 15 to epoch 20. The model will oscillate up and down as additional epoch are added as the learning rate increases. This viewpoint leads us to the conclusion that selecting the learning rate is one of the most challenging tasks because a value that is too small may cause a drawn-out and stuck training process, while a value that is too large may produce a learning rate with an inadequate set of weights or an unstable training process. The optimal learning rate, which yields good convergence, processing, and accuracy, is therefore between 0.001 and 0.0001.

A good fit model's testing loss is initially high, but it gradually decreases as more training examples are added and the learning curve flattens out. We can also see that the training and validation losses converged after there were enough training samples included. The losses during validation and training are essentially the same or overlap.

Starting with a complete dataset and running it through a neural network is not adequate and is absurd. The same neural network must be fed the entire dataset again. Although the epoch varies depending on the dataset, you can argue that the diversity of your data determines the number of epochs. According to figure 4.3 above, the accuracy of the TBDMobileNetV2 increases marginally as the epoch number increases.

Smaller batch sizes, on the other hand, have been demonstrated to have a faster convergence to good solutions. The greater the batch size results in the bad generalization gap since the model can't travel far enough in a fair number of training epochs. Smaller batch sizes enable the model to begin learning before viewing all of the data, which logically explains this. The iteration size is tiny in larger batches; therefore, convergence is difficult inside the smaller iteration. On the other hand, the larger the batch size does not converge as quickly as low batch size.

According to [61] since these hyper-parameters will influence the network's performance during its time to convergence, many of them must be modified in order to properly train the TBDMobileNetV2 to be able to recognize images of the input data. As a result, the batch size, or the number of images used to train the network in each epoch, is one of the most crucial hyperparameters to modify. The network takes too long to reach convergence when this hyperparameter is set too high, and accuracy is not increased.

4.7.Evaluation of the proposed TBDMobileNetV2 model

The performance of TBDMobileNetV2 model achieved 96 % of precision, 96 % recall, and also 96 % of f1-score as depicted on figure 4.3. The model TBDMobileNetV2 can be evaluated to assess its performance to detect the Pulmonary Tuberculosis infected image. We evaluated the model with the following performance measurements.

- Model accuracy: a model with higher accuracy is better for the detection of results.
- Model loss: a model with a lower loss value is better for the detection when taking new images that are not part of the training and testing datasets.
- Model size: a model with a smaller size is better at processing the dataset because it has fewer parameters to learn. The size of a model is determined by its parameters, not by the amount of training data.

- Time it takes to train the model: for classifying the testing data, the model with the fastest (shortest) time is preferred.

The performance of the TBDMobileNetV2 model can be influenced by a number of characteristics during model creation. In order to achieve the optimum, we varied the values of those parameters. Ultimately, we used batch size of 32, epoch 20, the optimizer "Adam," which is the best and most appropriate optimizer because it combines Adgrad and RMSProp, and grid search mechanisms to select this optimizer with 0.001 learning rate and dropout of 0.49 to reduce model overfitting in the final analysis.

4.8. Comparison with AlexNet Model using our Datasets

AlexNet is one the powerful model that has a capability of achieving high accuracy on very challenging datasets. AlexNet is one of CNN architecture used for any object detection task for various applications in computer vision domain of Artificial Intelligence problems. So that the performance of the AlexNet model achieves in our dataset is 85.61 percent accuracy for training and 83.89 percent accuracy for validation. As described on the figure 4.8 and figure 4.9 less performance of the model observed with our dataset. The lower performance of AlexNet model as compared to our TBDMobileNetV2 is due to a challenge that a model is hard to be applied to high resolution image and it is not deep enough as compared to other models. So that most often other models suppressed the AlexNet model in image classification and detection.

```
[INFO] training head...
Epoch 1/50
22/22 [==============================] - 114s 418ms/step - loss: 1.8607 - accuracy: 0.7273 - val_loss: 2.1255 - val_accuracy: 0.5556
Epoch 2/50
22/22 [==============================] - 8s 377ms/step - loss: 0.5340 - accuracy: 0.8256 - val_loss: 1.9078 - val_accuracy: 0.5500
Epoch 3/50
22/22 [==============================] - 8s 382ms/step - loss: 0.5016 - accuracy: 0.8474 - val_loss: 1.0931 - val_accuracy: 0.5667
Epoch 4/50
22/22 [==============================] - 7s 316ms/step - loss: 0.5030 - accuracy: 0.8241 - val_loss: 0.3612 - val_accuracy: 0.8889
Epoch 5/50
22/22 [==============================] - 8s 381ms/step - loss: 0.5138 - accuracy: 0.8110 - val_loss: 0.5333 - val_accuracy: 0.7611
Epoch 6/50
22/22 [==============================] - 7s 314ms/step - loss: 0.4265 - accuracy: 0.8503 - val_loss: 0.6124 - val_accuracy: 0.7611
Epoch 7/50
22/22 [==============================] - 7s 315ms/step - loss: 0.3721 - accuracy: 0.8735 - val_loss: 0.4003 - val_accuracy: 0.8111
Epoch 8/50
22/22 [==============================] - 8s 370ms/step - loss: 0.4636 - accuracy: 0.8241 - val_loss: 0.2737 - val_accuracy: 0.9000
Epoch 9/50
22/22 [==============================] - 8s 379ms/step - loss: 0.3819 - accuracy: 0.8648 - val_loss: 0.2326 - val_accuracy: 0.9111
Epoch 10/50
22/22 [==============================] - 7s 313ms/step - loss: 0.4113 - accuracy: 0.8561 - val_loss: 0.6337 - val_accuracy: 0.8389
```

Figure 4.8: AlexNet Model summary

```
Evaluating Alexnet model...
6/6 [==============================] - 1s 166ms/step
              precision    recall  f1-score   support

      Normal       0.97      0.86      0.91       100
Tuberculosis       0.85      0.96      0.90        80

    accuracy                           0.91       180
   macro avg       0.91      0.91      0.91       180
weighted avg       0.91      0.91      0.91       180
```

Figure 4.9: Classification accuracy of AlexNet model

The figure 4.10 below depicts the training accuracy and training loss of AlexNet model with our dataset. The loss curve described the rate of correctly and incorrectly classified datasets. High loss is observed on the figure that lowers the performance of the model evaluation. The loss curve is relatively far from zero up to epoch 10 and the curve is relatively near to zero from epoch 10 to the end.

Loss is going up for incorrectly classified samples and going down for correctly classified samples. A loss is the penalty for a false estimation. In other words, loss is a statistic that indicates how accurately the model predicted a particular input. The loss curve is close to zero if the model's forecast is accurate; otherwise, it is greater. Typically, an epoch's earliest batches suffer from higher losses than its last epoch. From the start of the epoch until about epoch 10, as seen in Figure 4.10 below, the loss of training is noticeably greater than the loss of validation. This happened as a result of regularization approaches being applied during the training phase as opposed to the testing phase. These regularization strategies are crucial for training accuracy to surpass testing accuracy.

Figure 4.10: training accuracy and training loss of AlexNet Model

As shown in the training loss and accuracy curve in figure 4.11 below, the training accuracy was higher than validation accuracy throughout the curve. We observed the small overfitting gap between the training and validation accuracy. The validation loss is initially high around the first consecutive epoch and the decrement of the loss observed quickly. However, the validation loss oscillates up and down throughout the curve and did not become stable. We also observed that the training loss was very small and stable throughout the curve. So that from the loss perspective, overfitting is still occurred in AlexNet model with our Pulmonary Tuberculosis disease detection.

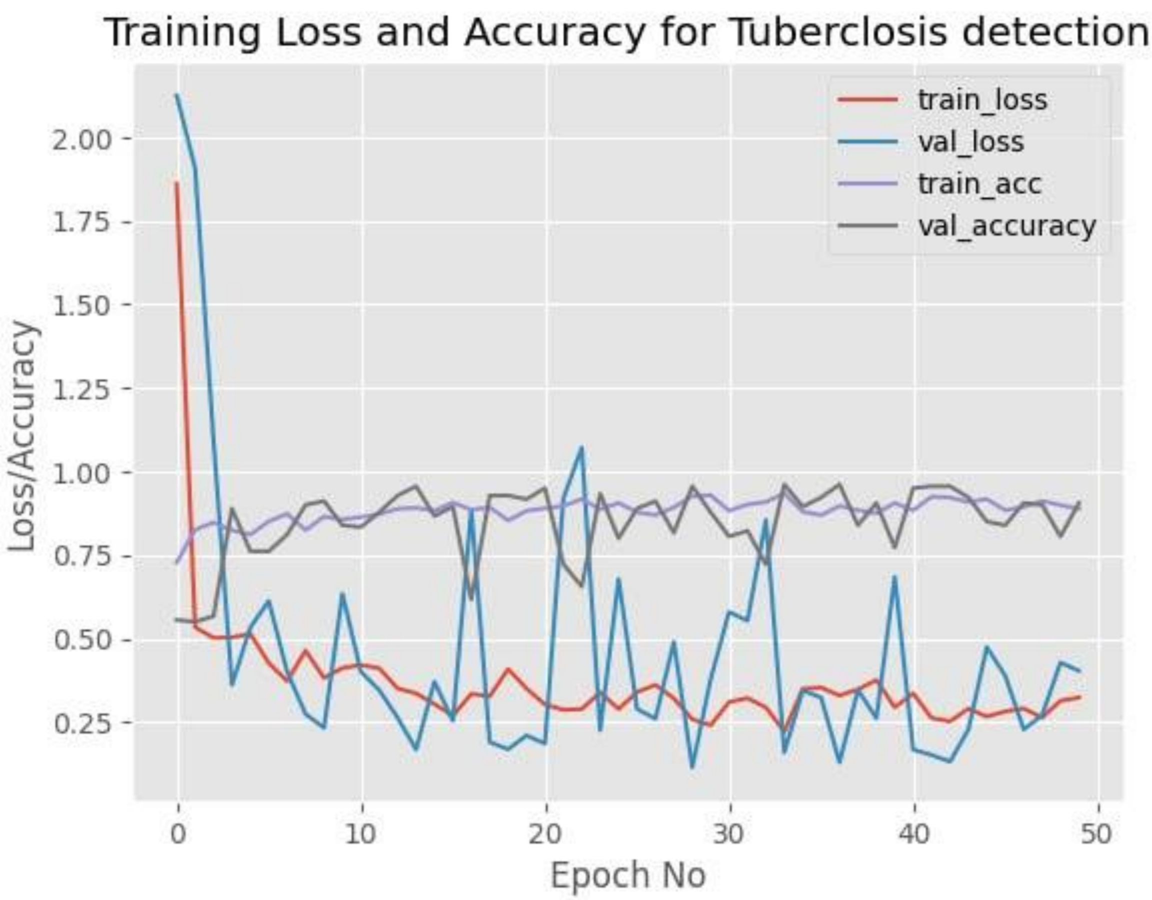

Figure 4.11: Training loss and accuracy for Tuberculosis detection

The figure below describes the training and testing accuracy of the AlexNet model for Pulmonary Tuberculosis disease detection. The graph describes the difference between the training and testing curve. Based on the graph, high overfitting observed between the training and testing graph. Around the first epoch large difference between training and testing accuracy observed. Jumping up and down of the testing accuracy observed on the figure.

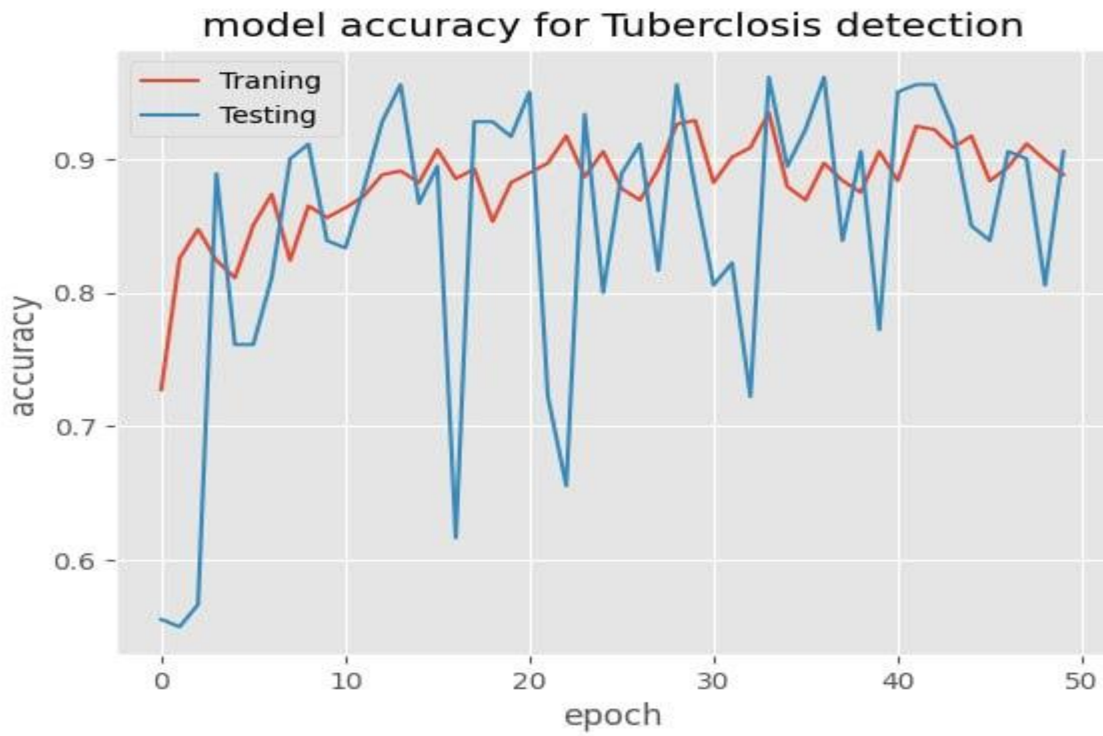

Figure 4.12: AlexNet model training and testing accuracy for Tuberculosis detection

The following figure 4.13 depicts the confusion matrix of AlexNet that describes the number of datasets that are correctly and incorrectly classified with the model. Besides, from the given dataset for testing 175 of the samples classified correctly and the remaining 5 samples classified incorrectly as observed on the figure.

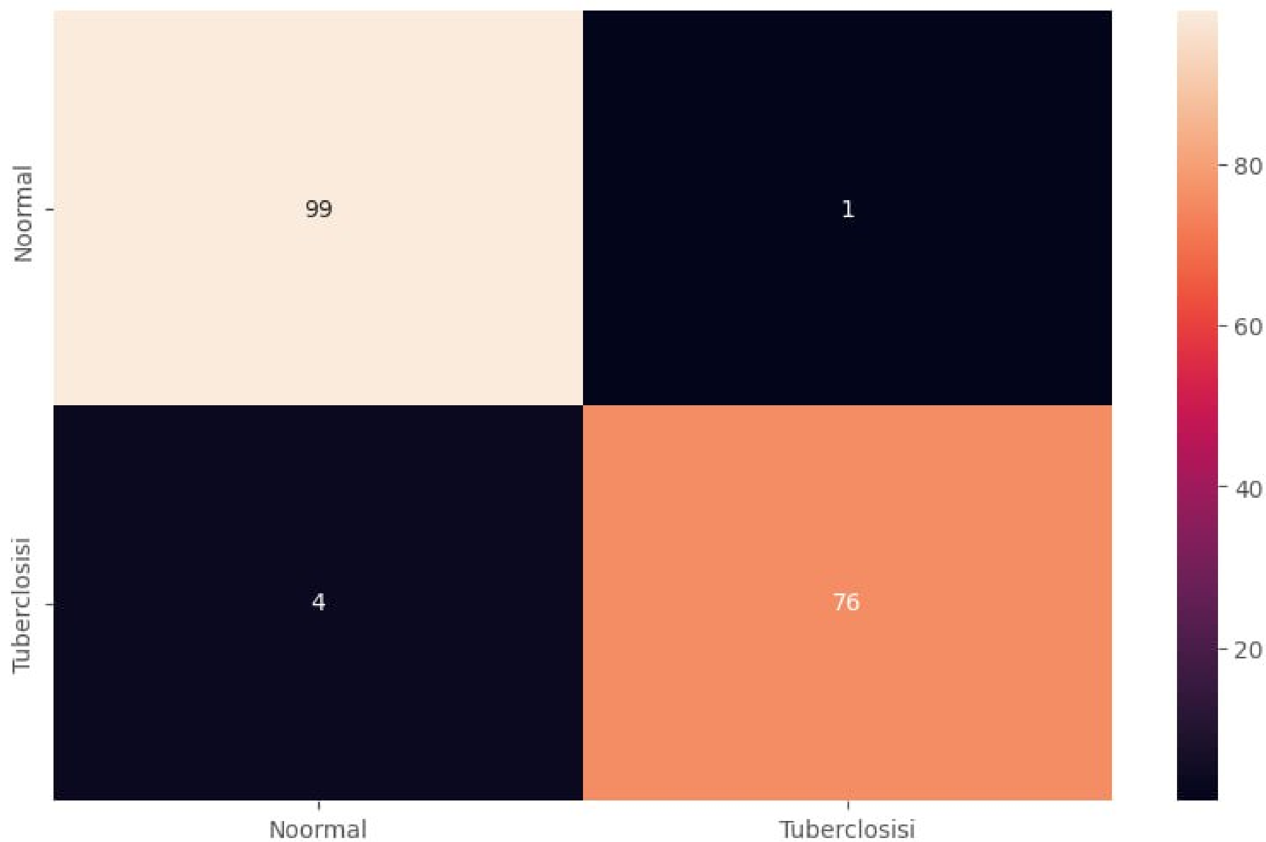

Figure 4.13: confusion matrix of AlexNet model for Pulmonary Tuberculosis disease detection

4.9. Comparison with VGGNet Model using our Datasets

The performance of VGGNet model is shown in figure 4.14 and figure 4.15 below. As depicted on figure 4.14 below, the VGGNet model has a training accuracy of 97.82 % and a testing accuracy of 73.33 % with our dataset. The accuracy of VGGNet model is lower than TBDMobileNetV2 model that obtains a testing accuracy of 96.11 %.

```
[INFO] training head...
Epoch 1/50
22/22 [==============================] - 12s 402ms/step - loss: 0.4483 - accuracy: 0.8038 - val_loss: 0.4069 - val_accuracy: 0.7944
Epoch 2/50
22/22 [==============================] - 9s 422ms/step - loss: 0.2123 - accuracy: 0.9244 - val_loss: 0.3519 - val_accuracy: 0.8278
Epoch 3/50
22/22 [==============================] - 10s 438ms/step - loss: 0.1586 - accuracy: 0.9404 - val_loss: 0.2554 - val_accuracy: 0.8778
Epoch 4/50
22/22 [==============================] - 7s 338ms/step - loss: 0.1339 - accuracy: 0.9564 - val_loss: 0.2479 - val_accuracy: 0.8778
Epoch 5/50
22/22 [==============================] - 8s 347ms/step - loss: 0.1363 - accuracy: 0.9477 - val_loss: 0.5163 - val_accuracy: 0.7333
Epoch 6/50
22/22 [==============================] - 9s 418ms/step - loss: 0.1129 - accuracy: 0.9666 - val_loss: 0.4133 - val_accuracy: 0.7944
Epoch 7/50
22/22 [==============================] - 9s 419ms/step - loss: 0.0824 - accuracy: 0.9797 - val_loss: 0.4567 - val_accuracy: 0.7778
Epoch 8/50
22/22 [==============================] - 7s 338ms/step - loss: 0.0936 - accuracy: 0.9709 - val_loss: 0.1495 - val_accuracy: 0.9389
Epoch 9/50
22/22 [==============================] - 8s 339ms/step - loss: 0.0864 - accuracy: 0.9709 - val_loss: 0.2408 - val_accuracy: 0.9000
Epoch 10/50
22/22 [==============================] - 9s 413ms/step - loss: 0.0709 - accuracy: 0.9767 - val_loss: 0.1947 - val_accuracy: 0.9278
Epoch 11/50
22/22 [==============================] - 8s 346ms/step - loss: 0.0819 - accuracy: 0.9787 - val_loss: 0.3521 - val_accuracy: 0.8333
Epoch 12/50
22/22 [==============================] - 8s 344ms/step - loss: 0.0627 - accuracy: 0.9782 - val_loss: 0.3283 - val_accuracy: 0.8500
Epoch 13/50
22/22 [==============================] - 9s 415ms/step - loss: 0.0828 - accuracy: 0.9695 - val_loss: 0.3229 - val_accuracy: 0.8500
Epoch 14/50
22/22 [==============================] - 8s 353ms/step - loss: 0.0527 - accuracy: 0.9869 - val_loss: 0.3307 - val_accuracy: 0.8611
Epoch 15/50
22/22 [==============================] - 7s 331ms/step - loss: 0.0707 - accuracy: 0.9782 - val_loss: 0.6390 - val_accuracy: 0.7333
Epoch 16/50
```

Figure 4.14: Model summary for VGGNet

Based on figure 4.14, VGGNet model has slower performance to train as compared to the proposed TBDMobileNetV2 model for Pulmonary Tuberculosis disease detection. The evaluation matrices such as precision, recall, and F1-score with our dataset to detect Pulmonary Tuberculosis are described on the following figure.

```
[INFO] evaluating network...
6/6 [==============================] - 1s 46ms/step
              precision    recall  f1-score   support

      Normal       0.89      1.00      0.94       100
Tuberculosis       1.00      0.85      0.92        80

    accuracy                           0.93       180
   macro avg       0.95      0.93      0.93       180
weighted avg       0.94      0.93      0.93       180
```

Figure 4.15: Detection accuracy of VGGNet model

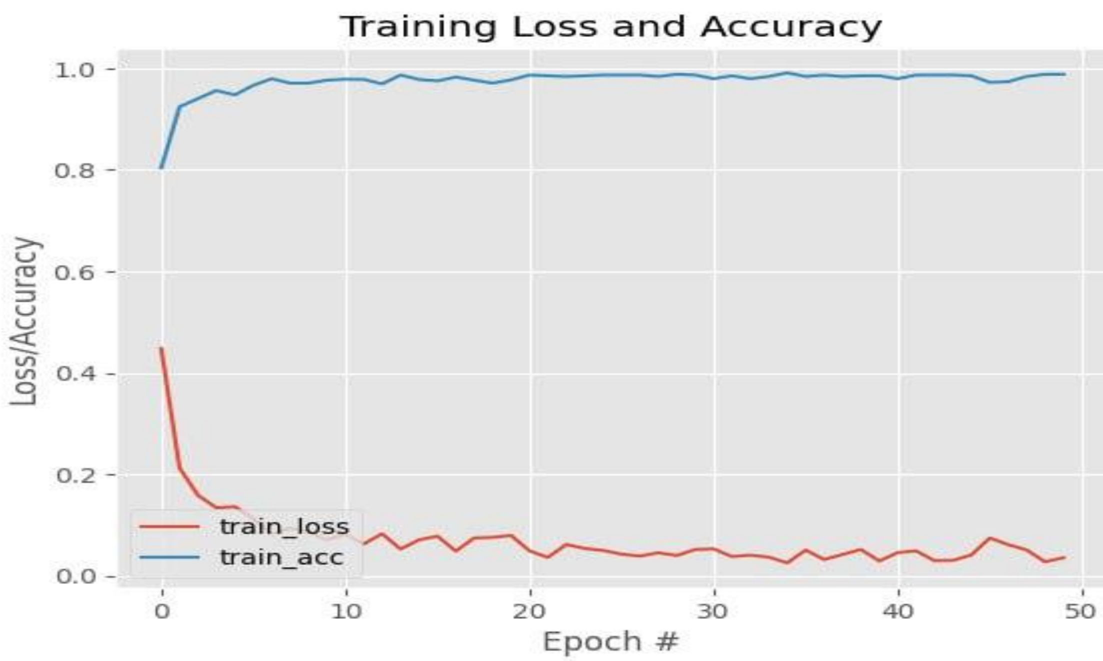

Figure 4.16: Training accuracy and loss for VGGNet model

The performance of the VGGNet model achieves 97.82 percent training and 73.33 percent validation accuracy on our data. As depicted on the figure 4.17 there is high overfitting happened on training and testing accuracy that results less performance of the model when tested with the unseen dataset. The result of VGGNet is described on figure 4.17 and we observed the fluctuation of the validation accuracy overtime. So that due to high overfitting happened on training and validation accuracy, there was larger generalization gap between the training and testing curves. As a result, this VGGNet model is unstable for the detection of Pulmonary Tuberculosis disease.

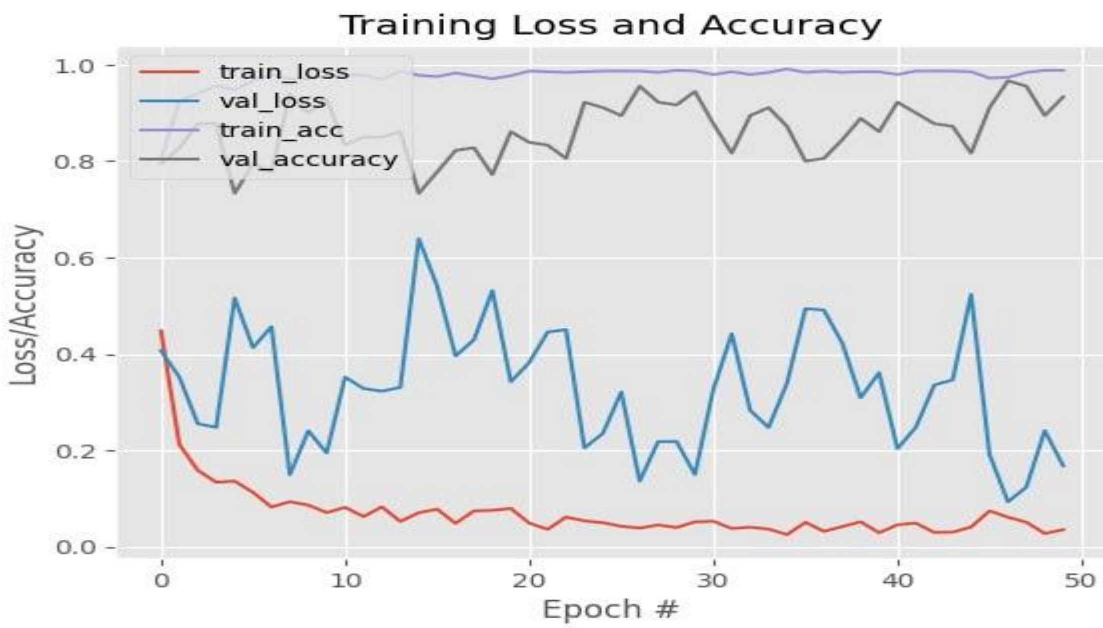

Figure 4.17: Training loss, validation loss, training accuracy, and validation accuracy of VGGNet model

The figure 4.18 below describes the training and testing accuracy of the VGGNet model for Pulmonary Tuberculosis disease detection. The graph describes the difference between the training and testing curve. Based on the graph, high overfitting observed between the training and testing graph. Around the first epoch small difference between training and testing accuracy observed. Jumping up and down of the testing accuracy observed on the figure.

Figure 4.18: VGGNet model accuracy

The diagram below shows the confusion matrix result of VGGNet model with our dataset. We tested the developed model with unseen dataset. Based on the following confusion matrix figure, from 180 test images 172 of the unseen dataset detected correctly and the remaining 8 images detected incorrectly with VGGNet model. The following figure describes the confusion matrix of the VGGNet model.

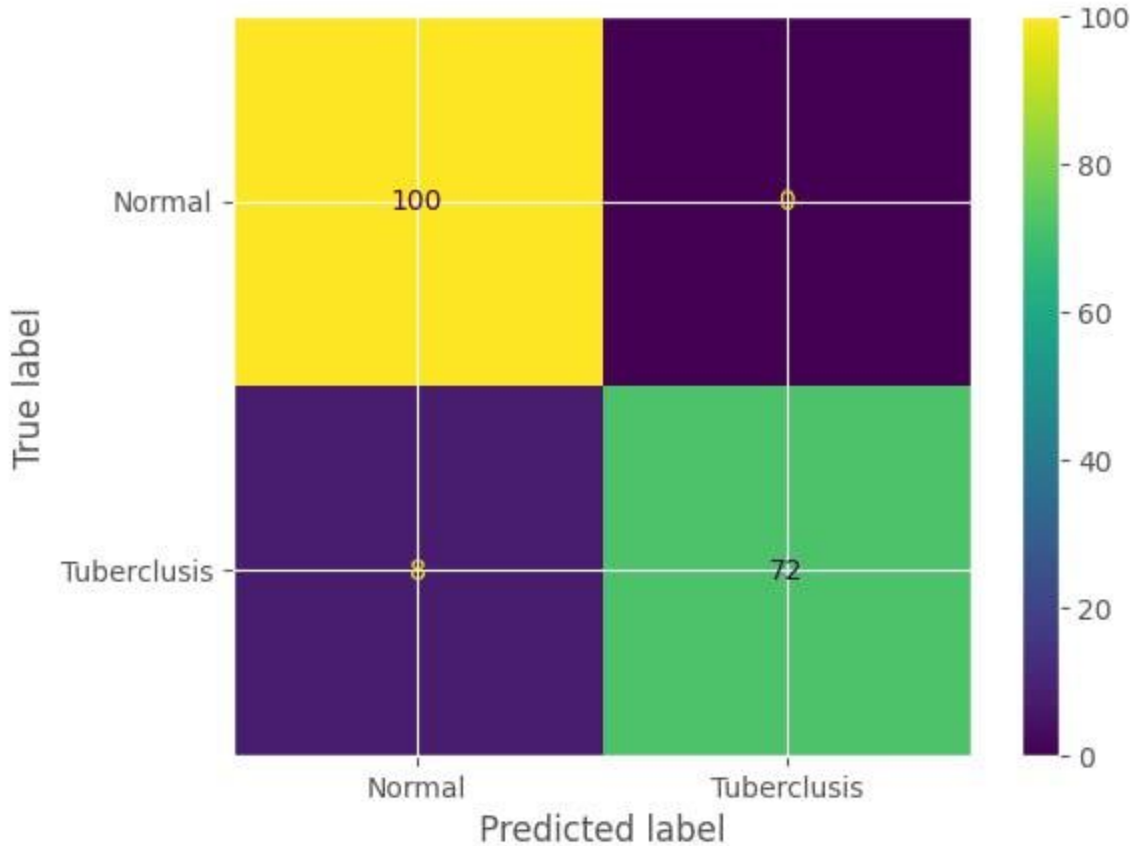

Figure 4.19: the confusion matrix of VGGNet model

4.10. Testing with unseen Datasets

The detection of Pulmonary Tuberculosis disease with TBDMobileNetV2 and unseen dataset described with the following figure 4.20. We used randomly selected dataset from anywhere. The collected unseen dataset is out of the dataset we collected from Jimma University medical center. The figure explores with unseen dataset there is high up and down testing accuracy observed. The graph describes the difference between training and testing accuracy. To some extent the training accuracy is stable however, the testing accuracy is oscillating high and low. So that the model performance decreases with unseen dataset.

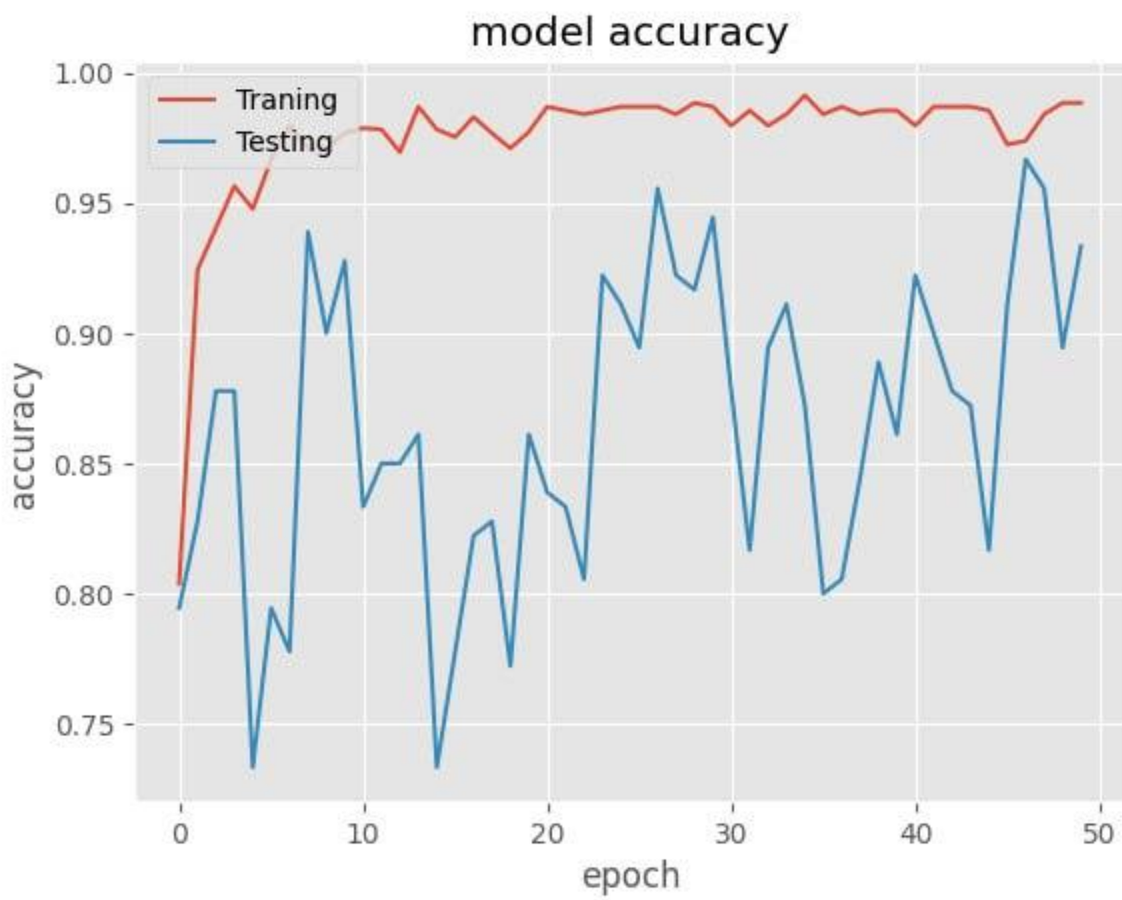

Figure 4.20: the confusion matrix of VGGNet model

4.11. Discussion

The bacteria Mycobacterium tuberculosis (MTB) is responsible for the dangerous infection known as pulmonary tuberculosis (TB), which affects the lungs but can also affect other organs. Isoniazid INH in combination with the three medicines rifampin, pyrazinamide, and ethambutol is the most often used treatment for active TB. Even though you might start feeling better just a few weeks after starting the medication, TB treatment takes far longer than treating other bacterial infections.

We can make the treatment of Pulmonary Tuberculosis easy with the use of our model TBDMobileNetV2 that increase the speed of diagnosis. Our model works 95.93% accurate to detect the existence of the disease using embedded device such as smart phones and other hand-held device. The accuracy of the model obtained from limited datasets that were collected from Jimma University Medical center. The parameters used for model development are selected manually with many trials. The validation accuracy of the model is 96.11% which is a measure of the model's skill while adjusting its hyperparameters.

The performance of TBDMobileNetV2 model for the detection of Pulmonary Tuberculosis disease has been analyzed with the use of the captured images and evaluation metrices results. We used Entropy feature fusion strategy and MobileNetV2 CNN architecture. We developed other models such as AlexNet and VGGNet to compare the result with the new proposed TBDMobileNetV2 model. These models trained, tested, and validated to detect Pulmonary Tuberculosis disease.

The main challenges on other models are overfitting and oscillation in training and validation loss or accuracy that lowers the performance of the model. Different testing accuracy of the result recorded at each iteration and it takes the average of the testing accuracy. So that different testing accuracy results observed using different hyperparameters for the models we developed.

Table 4.3: Summary of comparison of models

Model	Training accuracy	Validation accuracy
TBDMobileNetV2 model	95.93	96.11
AlexNet	85.61	83.39
VGGNet	97.82	73.33

Therefore, as described on the table 4.3, the newly proposed TBDMobileNetV2 model with Entropy based feature fusion strategy shows 95.93% training accuracy and 96.11% validation accuracy, which is the outperformed model as compared to the other developed models. The AlexNet model achieved 85.61 training accuracy and 83.39% validation accuracy which is lower than the newly proposed model. The VGGNet model achieved 97.82% training accuracy and 73.33% validation accuracy which is also lower than the proposed TBDMobileNetV2 model. So that our proposed model outperforms as compared to the other state-of-the-art models.

4.12. Summary

The detection of Pulmonary Tuberculosis using MobileNetV2 and Entropy based feature fusion is performed to detect the disease with the developed model TBDMobileNet.V2 The model development includes different phases such as dataset collection from health institutions, image preprocessing to resize, convert the RGB image into grayscale, and segment an image to separate ROI (region of interest) from the background and foreground; entropy-based feature selection;

split dataset for training and testing and finally for model development. Afterall, TBDMobileNetV2 obtained for the detection of Pulmonary Tuberculosis and the normal image. Other models are also developed to compare with the newly developed models. Based on the result we achieved, the newly developed model outperforms the other state-of-the-art models with 95.93 training accuracy and 96.11 validation accuracy.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

5.1. Conclusion

Mycobacterium tuberculosis (MTB) bacteria is responsible for the dangerous infection known as pulmonary tuberculosis (TB), which affects the lungs but can also affect other organs. The treatment of TB takes longer than other bacterial infection. NGO's and government are supporting the treatment of TB with funding the necessary materials for diagnosis and treatment due to its severity that consequences death on human beings.

This research work attempts to use MobileNetV2 architecture of CNN and Entropy based feature fusion strategy to detect Pulmonary Tuberculosis disease. We proposed a new model architecture MobileNetV2 based on entropy-based feature fusion strategy leading to an efficient model.

SO that, we collected x-ray image with pulmonary tuberculosis infected and on-infected from Jimma University medical center for model development and detection of the disease with MobileNetV2 based model. So that we developed TBDMobileNetV2 model with the collected dataset and various hyperparameters. We selected hyperparameters manually to obtain the best value of the hyperparameters and high performance of the model. We also developed other models such as AlexNet, VGGNet based on the state-of-the-art image processing models to compare results achieved.

Based on the result we achieved, the newly proposed TBDMobileNetV2 model with training accuracy of 95.93% and validation accuracy of 96.11% outperforms the other models.

5.2. Contribution

The main attempt of this research work is the detection of Pulmonary Tuberculosis disease using MobileNetV2 and entropy-based feature fusion strategy. So that we developed Tuberculosis Detection MobileNetV2 model (TBDMobileNetV2) model using MobileNetV2 and entropy-based feature fusion strategy. Besides, the following are the main contribution of our research work.

- We collected a dataset for our model development and this dataset is available for further research for the scientific community.
- We developed TBDMobileNetV2 model which helpful for the detection of Pulmonary Tuberculosis disease that minimizes the times of diagnosis by experts and for quick treatment of patients.

- We used entropy-based feature fusion strategy that improves the performance of the model which is better from CNN-based feature extraction technique.
- We compared the result of our model with other state-of-the-art image processing models such as AlexNet and VGGNet.

5.3. Recommendation

The detection of Pulmonary Tuberculosis disease with MobileNetV2 and entropy-based feature fusion strategy achieved a training accuracy of 95.93% and a validation accuracy of 96.11% with TBDMobileNetV2 model. Based on the result, we recommend the following for the scientific community.

- We developed a model with small dataset and we recommend to develop a model using huge amount of dataset, hence deep learning approaches require large amounts of dataset to enhance the model performance.
- We also recommend to enhance the value of the hyperparameters of MobileNetV2 model to improve the performance.
- This proposed model focused to classify infected and non-infected further researches could consider to perform to build a model that can classify species of pulmonary tuberculosis
- Adding extra more features to represent the image will undoubtedly increase the performance of the classification.
- If more labeled datasets are found it will lead to better result.

Reference

[1] M. Peakman and V. Diego , "Basic and clinical immunology.," *Elsevier Health Sciences,* 2009.

[2] WHO, "Global tuberculosis report," 2020.

[3] K. Erdem and M. Ozer , "Thymoquinone protects against the sepsis induced mortality, mesenteric hypoperfusion, aortic dysfunction and multiple organ damage in rats," *Pharmacological Reports,* 2017.

[4] D. Aspandi-Latif, S. Goldin and P. Rakwatin, "Texture Based Classification of High Resolution Remotely Sensed Imagery using Weber Local Descriptor," 2021.

[5] Yabsera, Erdaw; Erdaw, Tachbele, "Machine Learning Model Applied on Chest X-Ray Images Enables Automatic Detection of COVID-19 Cases with High Accuracy".

[6] R. Wambui, "Cross-Entropy Loss and Its Applications in Deep Learning," 2021.

[7] K. lönnroth, E. Jaramillo and B. Williams , "Tuberculosis: the role of risk factors and social determinants in: blas e & Kurup a (eds.), equity, social determinants and public health programmes.," 2010.

[8] "Global Tuberculosis report," WHO, 2022.

[9] "WHO. Global Tuberculosis Report. World Health Organization (WHO)," Geneva 2017, 2017.

[10] W. Wang and H. Yiyang , "A new image classification approach via improved MobileNet models with local receptive field expansion in shallow layers," *Computational Intelligence and Neuroscience,* 2020.

[11] R. Anatomy, "Manual of diagnostic imaging".

[12] A. G. Howard and Z. Menglong , "Mobilenets: Efficient convolutional neural networks for mobile vision applications," 2017.

[13] M. Raviglione and G. Sulis , "Tuberculosis 2015: Burden Challenges and Strategy for Control and Elimination Infect," vol. 8, pp. 65-70, 2016.

[14] A. A. Akosua and O.-A. Richard , "Tuberculosis—an overview," *Public Health and Emergency,* vol. 1, no. 37, 2016.

[15] G. Dolin GL, J. E. Mandell and B. Raphael , "Principles and Practice of Infectious

Diseases," *7th ed., Philadelphia, PA; Churchill Livingstone,* pp. 250-300, 2011.

[16] I. Campbell and O. Bah-Sow , "Pulmonary tuberculosis: diagnosis and treatment.," *Br Med J,,* vol. 332, no. 7551, pp. 1194-1197, 2006.

[17] World Health Organization, "Definitions and reporting framework for tuberculosis," *World Health Organization,* 2014.

[18] W. H. Organization, "Tuberculosis," 2022.

[19] WHO, "Global tuberculosis report 2021," 2021.

[20] W. H. Organization, "WHO, Global Tuberculosis report," 2018.

[21] W. Vision, "Technical Guideline for Tuberculosis (TB) and TB-HIV Program Implementation," World Vision, Uxbridge, UK, 2017.

[22] A. Adane and M. Damena, "Prevalence and Associated Factors of Tuberculosis among Adult Household Contacts of Smear Positive Pulmonary Tuberculosis Patients Treated in Public Health Facilities of Haramaya District, Oromia Region, Eastern Ethiopia," *Tuberculosis Research and Treatment,* 2020.

[23] S. Nimbarte , P. Deshmukh and A. Mehendale, "Effect of duration of cough (≥3 weeks vs≥2 weeks) on yield of sputum positive tuberculosis cases and laboratory load.," *Online J Health Allied Scs,* vol. 8, no. 2, p. 7, 2009.

[24] H. Ward , D. Marciniuk and P. Pahwa, "Extent of pulmonary tuberculosis in patients diagnosed by active compared to passive case finding," *Int J Tuberc Lung Dis,* vol. 8, no. 59, pp. 3-7, 2004.

[25] A. Negesse and M. Belay, "Prevalence of pulmonary tuberculosis among patients presenting with cough of any duration in Addis Ababa, Ethiopia," *Federal Ministry of Health, Addis Ababa, Ethiopia,* 2019.

[26] P. Oki Nugraha and N. H. Affan Yuniar , "Early detection of tuberculosis in patients with smear-negative pulmonary tuberculosis," *The Korean Journal of Internal Medicine,* vol. 37, no. 5, pp. 1088-1089, 2022.

[27] S. K. A, "review of computer-aided diagnosis in thoracic and colonic imaging.," *Quantitative Imag Med Surg.,* vol. 2, no. 3, pp. 163-176, 2012.

[28] F. Macedo and A. Giovani , "Computer-aided detection (CADe) and diagnosis (CADx)

system for lung cancer with likelihood of malignancy," 2016.

[29] Y. Zhang, Y. Zhang and Z. Dong, "Advanced Signal Processing Methods In Medical Imaging.," *IEEE Access Special Section Editorial,* vol. 6, pp. 61812-61818, 2018.

[30] Ç. Erdi and S. Ecem, "Deep learning for chest X-ray analysis: A survey," *Medical Image Analysis,* vol. 72, 2021.

[31] R. Suhail and F. David , "Interpretation of Plain Chest Roentgenogram, chest," vol. 141, no. 2, pp. 545-558, 2012.

[32] Q. L.G and K. A.G., "Detection of lung cancer on the chest radiograph: a study on observer performance," vol. 39, no. 2, pp. 111-116, 2001.

[33] Y. Balabanova, R. Coker and I. Fedorin, "Variability in interpretation of chest radiographs among russian clinicians and implications for screening programmes: observational study," vol. 331, no. 7513, pp. 379-382, 2005.

[34] M. Yong, "Interobserver variability in the interpretation of chest roentgenograms of patients with possible pneumonia," vol. 154 , no. 23, p. 2729, 1994.

[35] A. Jafar and N. Anand , "Machine Learning from Theory to Algorithms: An Overview," *Conference Series,* 2018.

[36] A. Tejpal and K. Kaur, "Machine Learning Survey," *Comput. Sci. Eng,* vol. 7, no. 2, p. 453–457, 2019.

[37] Al-Sahaf, Harith and Q. C. Ying Bi, "A survey on evolutionary machine learning," *ournal of the Royal Society of New Zealand,* vol. 49, no. 2, pp. 205-228, 2019.

[38] Z. Taohong , F. Suli and H. Junnan , "A Feature Fusion Method with Guided Training for Classification Tasks," 2021 .

[39] A. Md Manjurul and L. Shahana Akter , "Machine-Learning-Based Disease Diagnosis: A Comprehensive Review," *Healthcare,* vol. 10, no. 541, pp. 1-20, 2022.

[40] M. Ahsan, E. Alam, T. Trafalis and P. Huebner, "Deep MLP-CNN model using mixed-data to distinguish between COVID-19," vol. 12, no. 1526, 2020.

[41] Cai, Chenjing, W. Shiwei , X. Youjun and Z. Weilin , "Transfer learning for drug discovery," *Journal of Medicinal Chemistry,* vol. 63, no. 16, pp. 8683-8694, 2020.

[42] K. Kallianos, J. Mongan and S. Antani, " How far have we come? Artificial intelligence for

chest radiograph interpretation," *Clinical radiology,* vol. 74, no. 5, pp. 338-345, 2019.

[43] M. Hassaballah and K. M. Hosny, "Recent Advances in Computer Vision: Theories and Applications," *Springer International Publishing,* vol. 804, 2019.

[44] S. S. Chouhan, . U. P. Singh and . S. Jain, "Applications of Computer Vision in Plant Pathology: A Survey," *Comput. Methods Eng.,* vol. 27, no. 2, p. 611–632, 2020.

[45] A. Dhillon and G. Verma, "Convolutional neural network: a review of models, methodologies and applications to object detection.," *Progress in Artificial Intelligence,* vol. 9, no. 2, pp. 85-112, 2020.

[46] G. Yao , L. Tao and Z. Jiandan , "A review of convolutional-neural-network-based action recognition.," *Pattern Recognition Letter,* vol. 118, pp. 14-22, 2019.

[47] A. Laith and Z. Jinglan , "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," vol. 53 , 2021.

[48] J. Long , E. Shelhamer and T. Darrell , "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis & Machine Intelligenc,* vol. 39, no. 4, pp. 640-651, 2017.

[49] J. R. Uijlings J R , K. E. Sande and T. Gevers , "Selective Search for Object Recognition," *International Journal of Computer Vision,* vol. 104, no. 2, pp. 154-171, 2013.

[50] K. Ajitesh , "Different types of CNN Architectures Explained: Examples," in *Data, Data Science, Machine Learning, AI, BI, Blockchain*, 2018.

[51] A. Mohammad and A. Gharbi , "Detection of Tuberculosis Disease Using Image Processing Technique," *Mobile Information Systems,* pp. 1-7, 2021.

[52] L. Chang , C. Yu and A. Marlon , "Tx-Cnn: Detecting Tuberculosis In Chest X-Ray Images Using Convolutional Neural Network," *IEEE,* pp. 1-5, 2017.

[53] Nur-A-Alam , A. Mominul , Md and B. Abdul , "COVID-19 Detection from Chest X-Ray Images Using Feature Fusion and Deep Learning," vol. 21, no. 1480, pp. 1-30, 2021.

[54] L. Si-Yuan and W. Shui-Hua , "A Classification method for Brain MRI via MobileNet and Feedforward Network with Random Weights," *Pattern Recognition Letters,* vol. 140, pp. 252-260, 2020.

[55] C. Ines , E. Amira and K. Rafik , "Lung Disease Detection in Chest X-ray Images Using

Transfer Learning," 2022.

[56] L. Zhang et al.,, Increased risks of dental caries and periodontal disease in Chinese patients with inflammatory bowel disease,, vol. 70, Int. Dent. J.,, 2020, pp. pp. 227–236,.

[57] J. C. M. Román, J. L. V. Noguera, H. Legal-Ayala, D. P. Pinto-Roa, S. Gomez-Guerrero, and M. G. Torres, Entropy and contrast enhancement of infrared thermal images using the multiscale top-hat transform," Entropy,, vol. 21, 2019, p. pp. 1–19.

[58] A. G. Howard et al.,, MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, 2017.

[59] L. Rampasek and A. Goldenberg, "TensorFlow: Biology's Gateway to Deep Learning?, vol. 2, Cell Syst, 2016, pp. 12-14.

[60] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, Momentum， Nesterov accelerate， On the importance of initialization and momentum in deep learning, vol. 28, J. Mach. Learn. Res, 2013, p. 1139–1147.

[61] I. Kandel, M. Castelli, and A. Popovič, Comparative Study of First Order Optimizers for Image Classification Using Convolutional Neural Networks on Histopathology Images, vol. 6, J. Imaging, 2020.

[62] R. Irvin and C. Ciurea-Ilcus, "Chexpert: a large chest radiograph dataset with uncertainty labels and expert comparison," *AAAI Conference on Artificial Intelligence*, vol. 33, pp. 590-597, 2019.