DE-42 (EE) AMNA, FATIMA, SHUMAILA, WASEEM

BREAST CANCER DETECTION FROM MAMMOGRAMS USING DEEP LEARNING

COLLEGE OF
ELECTRICAL AND MECHANICAL ENGINEERING NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY RAWALPINDI
2024

COLLEGE OF ELECTRICAL AND MECHANICAL ENGINEERING

DE-42 EE
PROJECT REPORT

BREAST CANCER DETECTION FROM MAMMOGRAMS USING DEEP LEARNING

Submitted to the Department of Electrical Engineering
in partial fulfillment of the requirements
for the degree of
Bachelor of Engineering
in
Electrical
2024

Submitted by

NS Amna Muneer

NS Fatima Zafar

NS Shumaila Naveed

NS Waseem Arif

CERTIFICATE OF APPROVAL

It is to certify that the project **"BREAST CANCER DETECTION FROM MAMMOGRAMS USING DEEP LERANING"** was done by **NS Fatima Zafar**, **NS Amna Muneer**, **NS Shumaila Naveed**, and **NS Waseem Arif** under the supervision of **Assistant Professor Sobia Hayee**.

Submission: This project was submitted to the College of Electrical and Mechanical Engineering, National University of Sciences and Technology, Pakistan, as part of the requirement for the degree of Bachelor of Electrical Engineering.

Students:

1. **Amna Muneer**

NUST ID: 338890 Signature: ______________________

2. **Fatima Zafar**

NUST ID: 347638 Signature: ______________________

3. **Shumaila Naveed**

NUST ID: 348076 Signature: ______________________

4. **Waseem Arif**

NUST ID: 342321 Signature: ______________________

Approved By:

Project Supervisor: **Sobia Hayee**

Signature: ______________________ Date: __________________________

Head of Department: **Dr. Qasim Umar Khan**

Signature: ______________________ Date: __________________________

DECLARATION

We declare that no part of this project thesis has been submitted in support of an application for another degree or qualification. We have not submitted this thesis to any other university or educational institution. We are totally liable for any disciplinary action taken against us based on the nature of the proved offence, including the revocation of our degree.

Students:

1. **Amna Muneer**

NUST ID: 338890 Signature: ______________________

2. **Fatima Zafar**

NUST ID: 347638 Signature: ______________________

3. **Shumaila Naveed**

NUST ID: 348076 Signature: ______________________

4. **Waseem Arif**

NUST ID: 342321 Signature: ______________________

ACKNOWLEDGEMENTS

First of all, we would like to thank Allah Almighty who gave us the courage to pursue our ambition for our final year project and who helped us in our struggles to achieve our goals. Secondly, we would like to thank our project supervisor Sobia Hayee who guided us in selecting our final year project and helped us throughout our final year in achieving certain milestones for our project. Without her guidance we would not have been able to complete this project on time. Finally, we would like to thank our parents who prayed for us and who have high hopes for our future.

ABSTRACT

Breast Cancer detection with mammography, which is a critical task in medical imaging serves as a primary screening tool. Deep Learning techniques have shown significant improvement in recent years for improving efficiency and accuracy in Breast Cancer detection from mammograms. Our project has proposed a deep leaning model for breast cancer detection using mammogram images by deploying Convolutional Neural Networks (CNN) for feature extraction and Image classification. The model has been trained on large number of datasets containing mammogram images which have been annotated by the expert Radiologists. Experimental results show the accuracy and effectiveness of our model by detecting abnormalities in the respective regions of mammogram images which are indicators of Breast Cancer. This method has shown significantly improved performance as compared to traditional methods of Breast cancer detection. The proposed methodology has shown importance in the medical field by aiding radiologists in early detection and diagnosis and ultimately improving patient outcomes.

SUSTAINABLE DEVELOPMENT GOALS

Goal 3 – GOOD HEALTH AND WELL-BEING:

Our project aims to provide early diagnostics by detecting breast cancer which will help in promoting good health and well-being.

Figure 1 SDG Goal 3

Goal 9 – INDUSTRY, INNOVATION AND INFRASTRUCTURE

Our system detects breast cancer detection from mammograms using deep learning which is an advancement for the healthcare industry. The mobile application we designed helps the telemedicine industry in providing easier healthcare

Figure 2 SDG Goal 9

TABLE OF CONTENTS

CERTIFICATE OF APPROVAL……………………………………………………………… 3

DECLARATION……………………………………………………………………..…...…4

ACKNOWLEDGEMENTS……………………………………………………….......….5

ABSTRACT……………………………………………………………………………….6

SUSTAINABLE DEVELOPMENT GOALS……………………………………………....7

TABLE OF CONTENTS………………………………………………………………....8

LIST OF FIGURES……………………………………………………………………...12

Chapter 1 – INTRODUCTION…………………..………………………………….……..14

1.1 Understanding breast cancer: An overview………………………………..…....14

1.2 Introduction to deep learning……………………………………………….…..…14

1.2.1 Application for deep learning in medical imaging…………………….……………14

1.2.2 Motivation for deep learnig in mammograpghy……………………………..……....15

1.3 Procedure for breast cancer detection using deep learning……………..….…………15

1.3.1 Data Acquisition……………………………………………………...…….………15

1.3.2 Training and Model development………………………………….……….….15

1.3.3 Performance evaluation and validation……………………..………......….15

1.3.4 Real-time application.. 15

1.4 Our aim for this project 15

Chapter 2 – LITERATURE REVIEW 16

2.1. Background of breast cancer detection from mammograms using deep learning model .. 16

2.1.1 Breast cancer diagnostic techniques... 16

2.1.2 Early approaches and overview of machine learning algorithms..................................... 17

2.1.3 Deep learning approaches 17

2.1.4 Breast cancer detection through convolutional neural network....................................... 18

2.1.5 Data collection and selection……………………………………………………....18

2.1.6 Model selection 19

2.1.7 Training 19
2.1.8 Evaluation and testing 19
2.1.9 Real time prediction 19
2.1.10 Challenges 19
2.2 Literature Review 20
2.2.1 Previous work on breast diagnostic techniques 20
2.2.2 Machine learning algorithms in medical images 20
2.2.3 Deep learning approaches 21
2.2.4 Previous work on CNN architecture (Vgg-16 & ResNet-50) 21
2.2.5 Implementation on Raspberry pi 22
2.2.6 Development of app 22
2.2.7 Conclusion 22
Chapter 3- Methodology 23
3.1 Introduction 23
3.1.1 Proposed methodology 23
3.2. Convoloutional Neural Network architecture 23
3.2.1 Training the neural network 24
3.2.1.1 Convolutional layer 24
3.2.1.2 Pooling layer 25
3.2.1.3 Fully connected layer 26
3.3 Model selection 26
3.3.1 ResNet-50 26
3.3.2. Vgg-16 27
3.3.3. Google-net 28
3.3.4 Alexnet 29
3.3.5 Final model selection 29
3.4 Overview of Dataset 30
3.4.1. Dataset of breast mammography (DBM) 30
3.4.2 Curated breast imaging subset DDSM dataset (CBIS-DDSM) 31

3.4.3 Radiological society of north america (RSNA) ... 31
3.4.4 King abdul azuz university mammogram dataset (KAUMDS) ... 31
3.5 Final datset and model selection ... 32
3.6 Vgg-16 architecture ... 36
3.7 Softaware implementation of selected model ... 37
3.8 Classifier performance index ... 38
3.9 Raspberry Pi ... 39
3.9.1 Specifications ... 39
3.9.2 Raspberry Pi setup ... 40
3.9.3 Hardware implementation ... 40
3.10 Mobile App development… ... 41
3.10.1 Mobile App development stages ... 42
3.10.2 Our Mobile App (MammoDetect) ... 43.
3.11 Software components ... 45
3.11.1 Software applications used ... 45
Chapter 4- Results ... 49
4.1 Introduction ... 41
4.2 Summary of findings ... 41
4.3 Hardware Results on Raspberry pi ... 51
4.4 Mobile Application results ... 52
Chapter-5 Conclusion and Future works ... 53
5.1 Conclusion ... 53
5.2 Future work suggestions ... 53
5.2.1 Integrating cloud computing ... 53.
5.2.2 Collaboration with research labs ... 54
5.2.3 Partnerships with diagnostic centres ... 54
5.2.4 Industry collaboration ... 54
5.2.5 Telemedicine and remote diagnostic ... 54

REFERENCES ……………………………………………………………………………….55

APPENDIX A: Sustainable Development Goals…………………………………………………58

APPENDIX B: Complex Engineering Problem…………………………………………………..59

APPENDIX C: CODE……………………………………………………………………...…61

APPENDIX D: PLAGIARISM REPORT………………………………………………...…66

LIST OF FIGURES

Figure 1 SDG Goal 3..7

Figure 2 SDG Goal 9..7

Figure 3 Breast mammograpghy screening...16

Figure 4 Branches of machine learning………………………………………………..17

Figure 5 Illustration of how CNN performs…………………………………..……24

Figure 6 Splitting dataset into training & validation. 24

Figure 7 Convolution layer scheme. 25

Figure 8 Pooling layer scheme. 25

Figure 9 Fully connected layer scheme. 26

Figure 10 ResNet-50 Architecture 27

Figure 11 Vgg-16 Architecture 28

Figure 12 Google-net Architecture. 29

Figure 13 AlexNet Architecture...29

Figure 14 Vgg-16 on RSNA dataset...32

Figure 15 ResNet-50 on RSNA dataset. 33

Figure 16 Vgg-16 on DBM 34

Figure 17 ResNet-50 on DBM 35

Figure 18 Model summary. 38

Figure 19 Basic framework of Confusion matrix...38

Figure 20 Specifications of Raspberry Pi...40

Figure 21 Hardware Methodology. 41

Figure 22 Mobile App development stages…………………………………………….43

Figure 23 MammoDetect design…………………………………………………….44

Figure 24 Google Collab logo……………………………………………………………46

Figure 25 Thonny IDE logo…………………………………………………………….46

Figure 26 TensorFlow logo……………………………………………………………47

Figure 27 NumPy logo…………………………………………………………………47

Figure 28 Keras logo……………………………………………………………………48

Figure 29 Android Studio logo…………………………………………………………48

Figure 30 Training & validation accuracy vs training & validation loss………………………50

Figure 31 Final confusion matrix performance……………………………..…………….50

Figure 32 Raspberry pi integrated with LCD display…………………………………………51

Figure 33 LCD display showing results…………………………………………………….51

Figure 34 MammoDetect Results…………………………………………………………..52

Chapter 1 – INTRODUCTION

1.1. Understanding Breast Cancer: An Overview

Breast cancer is a type of cancer that starts in the cells of the breast. It appears when breast cells grow abnormally and uncontrollably, forming a mass of tissue, also called a tumor. This tumor can be cancerous (malignant) or non-cancerous (benign). A non-cancerous tumor is made up of cells that look like normal cells and don't invade nearby tissues or spread out to other parts of the body. A cancerous cell is spread to other parts of the body, such as lymph nodes, bones, liver, or lungs, through a process called metastasis.

Breast cancer has a significant impact on public health globally. In Pakistan, one in nine women have a risk of being diagnosed with breast cancer during their lifetime. Especially in rural areas of Pakistan, due to lack of awareness and early diagnostics, there is high mortality rate among women who are diagnosed with this type of cancer.

Mammography plays an important role in breast cancer detection. It is basically a screening tool that uses X-rays to diagnose and locate tumors in the breast. Regular mammography screenings help in identifying any abnormality in breast tissue that can help detect breast cancer at an early stage when it is most treatable.

1.2. Introduction to deep learning

Deep learning is a subset of machine learning that is inspired by the human brain structure, specifically the interconnection of neurons, known as the neural network. A neural network consists of nodes or artificial neurons that receive input, performs some computation on it and then passes it to the next layer of neurons. There are multiple layers in a neural network that allows it to extract complex features from the data making deep learning algorithm a powerful tool in various fields.

1.2.1. Application for deep learning in medical imaging

With advancements in deep learning, medical imaging has become much easier to diagnose results. Deep learning algorithms particularly the neural networks have shown great results in medical image analysis. Preprocessing techniques on images like image segmentation, feature extraction, classification and diagnostics all can be done by deep learning models.

1.2.2. Motivation for deep learning in mammography

Deep learning has emerged as a significant tool in medical diagnostics. For breast cancer detection, mammography plays a critical role in diagnostics and detection. A mammogram result can be challenging when detecting cancer due to the complexity of breast tissue and the subtle nature of abnormalities that could indicate cancer. Radiologists have been detecting breast cancer from mammogram images, but it can be time-consuming and with advancements in technology, deep learning can be used to predict results with greater accuracy.

1.3. Procedure for Breast cancer detection using Deep learning

Breast cancer can be detected using a deep learning model by following a procedure. Here is an overview of the general procedure:

1.3.1. Data Acquisition

Mammogram dataset is required for breast cancer detection. The dataset should be diverse and accurate so that deep learning model can correctly learn and extract useful features from it.

1.3.2. Training and Model Development

After data acquisition, the dataset is split into training and testing sets and then deep learning models like CNN can be applied to the train set so that the model can learn its features.

1.3.3. Performance Evaluation and Validation

Lastly model accuracy and performance can be evaluated using different performance metrics. Predictions for the test dataset can be cross verified with the actual mammogram labels and accuracy can be further improved with different optimization techniques.

1.3.4. Real-time Application

With real-time mammogram image that is acquired from an actual patient, we can verify the model’s prediction.

1.4. Our aim for this project

Our objective for this project is to develop a device that can detect breast cancer from given mammogram images using deep learning. This device is then deployed onto raspberry pi such that we have a portable device that can be used in hospitals and medical centers where a radiologist is not present. A mobile application is also developed that will be useful in remote areas for early diagnostics and treatment.

Chapter 2 – BACKGROUND AND LITERATURE REVIEW

2.1. Background of Breast Cancer Detection from Mammograms using Deep Learning Models

This section provides a detailed background of breast cancer detection from mammograms using deep learning model, highlighting key research contributions and advancements.

2.1.1. Breast Cancer Diagnostic Techniques

MRI(Magnetic Resonance imaging) [1] that uses non-invasive imaging techniques, ultrasound [2] that uses sound waves, Biopsy [3] that removes breast tissues and then examines under microscope and thermography [4] that detects cancer by measuring the changes in the temperature of the breast tissues are of moderate success due to certain limitations. Higher costs, operator-dependent techniques and expertise of sonographer, painful and dangerous process, and producing more false positives effecting the reliability respectively are some of the drawbacks [5]of these techniques.

Figure 3 Breast mammography screening

Mammography [6] is proven to be an advanced and effective diagnostic technique even at a very early stage, even where doctor and patient are unable to detect. Also, it is a non-invasive technique [7] that does not require needles or any type of anesthesia along with its wide availability that ultimately reduces diagnostic test cost. Mammography utilizes low dose X-rays.

2.1.2. Early Approaches and Overview of Machine learning algorithms

Early studies on breast cancer detection focused on Computer-Aided Detection (CAD) systems [8] , classifiers such as support vector machines (SVMs) [9] and other machine learning algorithms [10], although maximum accuracy achieved by these models varied depending on factors such as characteristics of datasets, model architecture, and certain evaluation metrics, but major drawback when these classifiers are applied to imagery datasets is that they extract handcrafted features manually from mammograms. [11] While these approaches achieved moderate success, they often faced numerous challenges because these traditional methods may not extract relevant features due to manual feature extraction method, leading to wrong predictions on new unseen data that is given to the model.

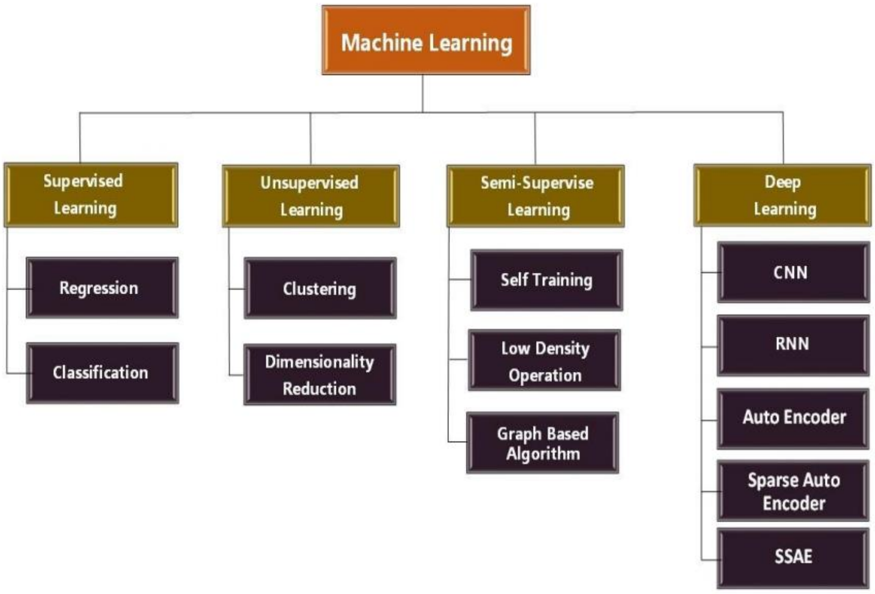

Figure 4 Branches of machine learning

2.1.3. Deep Learning Approaches

Deep learning [12] is the type of machine learning. The advent of deep learning revolutionized the field of breast cancer detection. Deep learning models such as Convolutional Neural Networks (CNNs) [13] and Recurrent Neural Networks (RNNs) [14] have shown remarkable performance in various computer vision tasks with CNN especially in detecting tumors in breasts from mammograms and Histopathological images.

CNN-based architectures [15] have been used to automatically learn and extract discriminative features directly from raw images. These deep CNN architectures such as VGG-16 [16] , ResNet [17], AlexNet [18] and Inception [19], have depicted superior performance in medical images and diagnostic tasks, surpassing traditional handcrafted feature-based approaches. Transfer learning techniques, where pre-trained CNN models are fine-tuned on large datasets called ImageNet [20], have also been effective in addressing data scarcity and improving performance.

CNN-based models have been employed to datasets. These models have been successful in recognizing lesions in mammograms.

2.1.4. Breast Cancer Detection Through Convolutional Neural Networks

Advancements in machine learning techniques and deep learning models have fueled the development in medical images in recent years. Both 2-dimensional and 3-dimensional structures of organs are necessary to detect what is normal and what is abnormal. CNN [21] is effective on imagery tasks while maintaining these relationships. CNN is a supervised machine learning algorithm which utilizes large training datasets. CNN preserves the spatial relationships while filtering input images. The research initiated with training and evaluating on two best models VGG-16 and ResNet-50 using four datasets and then finally selecting the best model with effective dataset. The best trained model is then deployed on hardware Raspberry Pi [22] and integrated on App. [23]

The key steps involved in breast cancer detection from mammograms using deep learning models are listed below.

2.1.5 Data Collection and Selection

The availability of labeled datasets plays a crucial role in the development and evaluation of breast cancer detection from mammograms using deep learning models. Several benchmark datasets have been widely used in the field taken from Kaggle. The first step is to select the dataset from various sources such as medical centers, hospitals, diagnostic centers, or datasets used in research papers. We collected four datasets from Kaggle for training our model. DBM (Dataset of Breast Mammography) [24], CBIS-DDSM (Curated Breast Imaging Subset DDSM Dataset) [25] , RSNA (Radiological Society of North America) [26] and KAUMDS (King Abdul Aziz University mammogram dataset) [27] from Kaggle due to better visuals and they were labeled already as benign and malignant.

2.1.6. Model Selection

CNN architectures [28] such as VGG-16, AlexNet and Resnet-50 are used widely for image classification purposes, but we selected two architectures VGG-16 and ResNet-50 because through analysis of various research papers we came to know that best image classification and detection of breast tumors are resulted through these two CNN architectures with correct real-life predictions. [29]

2.1.7. Training

Four datasets are used for training the models [30] VGG-16 and Resnet-50 and then evaluating certain performance matrices including accuracy. Highest accuracy of 97 percent with correct predictions were evaluated on DBM (Dataset of Breast Mammography) utilizing VGG-16. [31] After selecting final dataset pre-processing techniques were applied to the data for augmentation and then final selected model VGG-16 is used for training the labeled DSM dataset.

2.1.8 Evaluation and Testing

Evaluation metrics commonly used in breast cancer detection include accuracy, precision, and confusion matrix and are often employed to assess the generalization performance of models. These models are typically trained and tested on labeled datasets. The testing phase involves giving new unseen data to model for correct predictions.

2.1.9 Real-Time Predictions

Real life mammograms of three patients are taken from hospital and tested on google collab in testing phase. Correct real-life predictions provide basis to deploy it into hardware and integrate it into app.

2.1.10. Challenges

Despite significant progress, breast cancer detection from mammograms using deep learning model still faces several challenges [32]. Variations in image quality and breast density remain areas of concern. Efforts are being made to develop robust algorithms that are invariant to such variations and can generalize well across different mammograms.

Another challenge lies in curated datasets as they do not generalize well on real world images and new unseen data. This occurs due to variations in equipment, patient's demographics, and imagery protocols.

Additionally, ethical considerations related to patient's privacy and data security need to be carefully addressed. Ensuring fairness and transparency in mammography alongside patients' ID and personal information is essential to prevent potential misuse and unintended consequences.

2.2. Literature Review

This section includes Previous Published works on breast cancer detection from mammograms using deep learning model, it represents various recent research work and contributions done in the field of breast cancer detection from mammograms using deep learning.

2.2.1. Previous Works on Breast Diagnostic Techniques

The research paper written by D. Surya Gowri et al [33] depicts different types of methods used for breast cancer detection including mammograms which is an ensemble term covering CAD (Computer Aided Design), digital mammograms, and breast tomosynthesis. Other than this, PET and MRI scans are used. Among these, the paper emphasizes more on detection through digital mammograms. It also demonstrates various datasets available for use, namely - DDSM (Digital Database for Screening Mammograms), UCI Machine Learning Repository [34] and the Mini-MIAS Database [35]. The primary aim of this paper, however, focuses on how mammograms are effective in detecting breast cancer lesions.

2.2.2. Machine Learning Algorithms in Medical Images

Thesis for master's degree in Telecommunications and Networks by university of Buea titled as "Machine Learning-Based Breast Cancer Detection" provides the efficient ways to detect breast cancer using Machine learning involving artificial intelligence methods like Convolutional Neural Networks with 97% accuracy results. Moreover, App is also developed for user-friendliness, in order to detect cancer even at home instead of coming all the way to diagnostic centers. This method paves way to technological advancement and revolutionizes the way we live.

Rodriguez-Ruiz et al. evaluated an Artificial Intelligent (AI) system against radiologists in the detection of breast cancer using digital mammograms. The results acquired proved that the AI system was able to detect breast cancer far more accurately than radiologists. Dhahri et al. [36] compared the performance of KNN, SVM, Decision Trees (DT), Random Forest (RF), AdaBoost, Gradient Boosting (GB), Gaussian Naïve Bayes (GNB), Linear Discriminant Analysis (LDA), quadratic discriminant analysis (QDA), linear regression, and extra trees classifier, while the features were selected using Genetic Programming (GP) optimization. Khan et al. [37] adopted pretrained CNNs (GoogLeNet, VGGNet, and ResNet), which were fed into a fully connected network layer for the classification of malignant and benign cells using average pooling classification, which achieved a 97.52% accuracy on two breast microscopic image datasets.

2.2.3. Deep Learning Approaches

The foundation of detection of breast cancer using deep leaning models can be traced back to the pioneering work of researchers from the Massachusetts Institute of Technology (MIT) and Massachusetts General Hospital (MGH). Their study, [38] titled "Deep Learning for Breast Cancer Screening Using Mammograms," introduced a deep learning model called "DeepBreast." This model was trained on a large dataset of mammograms and the highest accuracy was obtained. This research aids Radiologists for better understanding of breast cancer screening and diagnostics.

“A Deep Learning Mammography-based Model for Improved Breast Cancer Risk Prediction” [39] published in 2019 by Radiological Society of North America (RSNA) demonstrated proven results in accurate predictions and 95% confidence interval [CI].

2.2.4. Previous Works on CNN architecture VGG-16 and ResNet-50

For the feature extraction, training and testing purpose, we used the VGG16 model. This is a CNN architecture comprising of sixteen layers. Accompanied by convolutional layers and the final SoftMax layer to perform classification. This architecture and its configurations have been explained by Simonyan & Zisserman 2016 [40]. The VGG16 model is denoted by configuration D in this paper. This is a pre trained network on the ImageNet [41] database which consists of 1.2 million images over 1000 classes.

“Fine tuning deep learning models for breast tumor classification” [42],this paper proposes an approach to detect benign and malignant Breast Tumors using histopathology images from the BreakHis [43] dataset. The main stages involve preprocessing, which encompasses image resizing, data partitioning (training and testing sets), followed by data augmentation techniques and evaluating accuracies up to 91 percent using ResNet-50 architecture.

2.2.5. Implementation on Raspberry Pi

“Classification of Mammography Images Using Deep-CNN based Feature Ensemble Approach and its Implementation on a Low-Cost Raspberry” [44]. This paper describes the Procedure for deploying the already trained DL mammography image classification model on a Raspberry Pi 3B. The deployment of trained DL models on low-cost, low, and effective-computational devices opens an opportunity for researchers to develop faster and cost-effective Detection systems for disease diagnosis.

2.2.6. Development of App

Thesis for master’s degree in Telecommunications and Networks by university of Buea titled as “Machine Learning-Based Breast Cancer Detection” [45] provides the framework and explains the stages to make the mobile and Web application. These steps divide the development process into tasks that can then be assigned, completed, monitored, and measured. It outlines essential tasks that are required to put together and make a mobile application.

2.2.7. Conclusion

Investigation of the relevant literature helps in analyzing and sorting out different deep learning models and machine learning algorithms and techniques that could be exploited in detecting breast cancer. After reviewing all techniques, deep learning models and datasets, their performance matrices including accuracy, the number of times they appeared in a journal, the optimum deep learning technique that our paper selected for breast cancer detection is, Convolutional Neural Networks (CNN), specifically VGG-16 with DBM dataset to train our dataset and to make correct prediction of detecting lesions and recognizing as benign or malignant. For the proposed methodology, the CNN architecture and functioning shall be further explained in Chapter 3.

Chapter 3 – METHODOLOGY

3.1. Introduction:

This section summarizes the various methods and techniques adopted to implement the project. It provides an in-depth look at the specific dataset used, highlights the complexity of selecting and training a model in a software environment, software and libraries used in this regard, and highlights several implications important for understanding the program. By providing an overview of these features, this section aims to provide clarity and promote reproducibility to ensure that project processes and outcomes are clear and understandable.

3.1.1. Proposed Methodology:

The aim of our project is to employ the deep learning model which uses Convolutional Neural Networks (CNN), to detect Breast Cancer from mammograms by extraction key features and meaningful information from images and then classifying them into benign and malignant categories respectively. After the model is trained on the large number of datasets it is being deployed onto Raspberry pi for portability. In the end the mobile app has been deployed for the ease of users which will enable then to input mammogram images and receive instant diagnostic outcome.

3.2. Convolutional Neural Network Architecture:

Convolutional Neural Network is a deep learning network that plays a vital role in the field of medical image analysis. They have been used in our project due to its previous works in related field and ability to detect abnormalities in medical images such as mammograms. The architecture of CNN is designed in such a way that it focuses on visual processing mechanisms of human brain. It consists of multiple layers for extracting meaningful information from the images. Each layer has its own specific feature. Here in our project, we have implied transfer learning by using the pre-trained model of CNN from ImageNet. CNN was preferred over ANN algorithms because here we are using digital images so it is much easy to deal with then using CNN.

CNN consists of 3 types of layers. These layers are convolutional layer, pooling layer and fully connected layer. Stacking up all these layers forms the CNN architecture which is shown in figure.

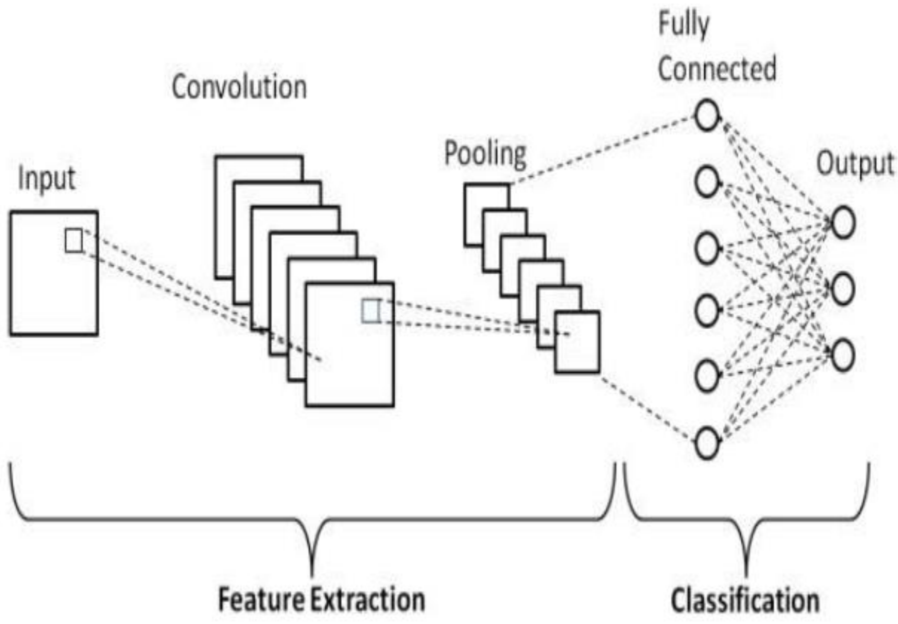

Figure 5 Illustration of how CNN performs.

3.2.1 Training the Neural Network:

The pre-trained CNN model which we are using is then trained on our dataset of mammograms in colab using TensorFlow and Keras modules. The classification is performed by splitting our dataset into training and validation sets i.e., 80% training and 20% validation.

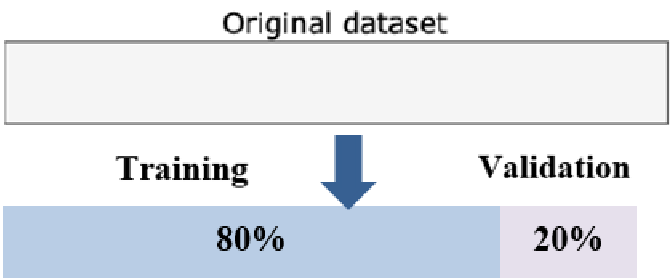

Figure 6 Splitting dataset into training & validation.

Then in the end we plotted the graphs of training and validation accuracy and loss in order to evaluate the classification performed by our model in terms of benign and malignant. These graphs are explained in the next chapter.

3.2.1.1 Convolutional Layer:

Figure-5 is showing the convolution operation. Convolutional layer is the first layer of CNN whose task is to perform feature extraction by slider the filter over input image. The output of this convolution layer is the element wise product of filters in the image and and their sum for every sliding action. The output layer which is also known as feature map refers to curves, sharp edges, textures etc. in the images.

In the networks that involve more convolutional layers, all the generic features are extracted in the initial layers while the complex parts are removed as the networks becomes deeper.

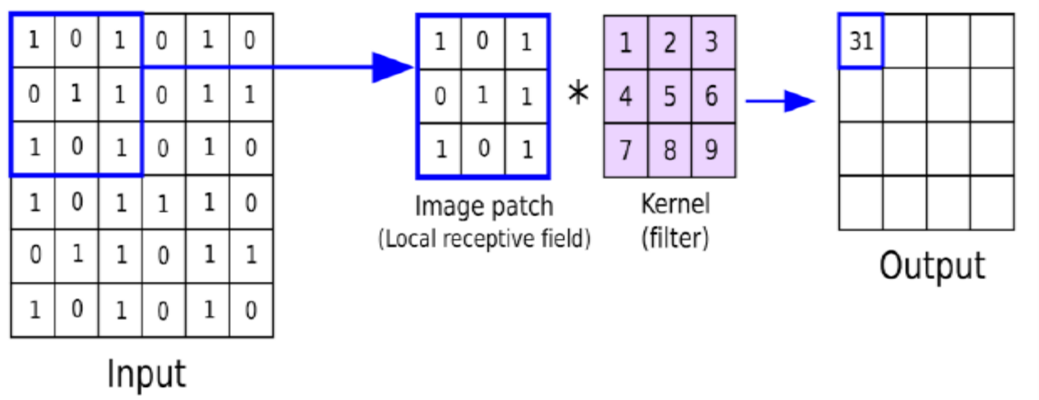

Figure 7 Convolution layer scheme.

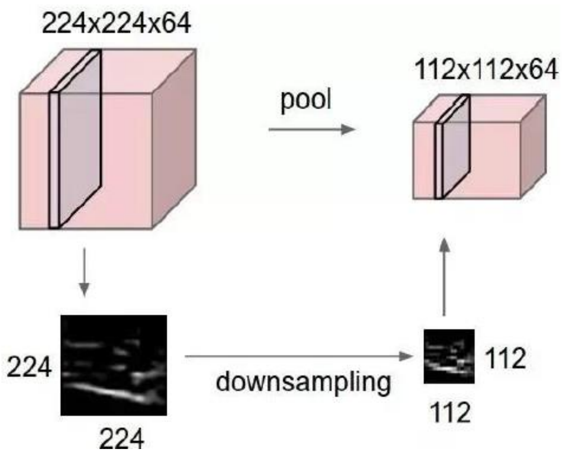

Figure 8 Pooling layer scheme.

3.2.1.2 Pooling Layer:

Figure shows the working scheme of pooling layer. This layer's purpose is to decrease the number of trainable parameters by reducing the spatial size of the image and as a result computational cost also decreases. The image depth is not changed because pooling is performed independently in each depth dimension. The most common pooling method is max pooling whereas feature map provides the most significant element as an input. In the end max pooling is performed to reduce the dimensions of output image to great extent while retaining the useful information.

3.2.1.3 Fully Connected Layer:

Functioning of fully connected layer is shown in figure. Output is determined by the last few layers which are fully connected layers. The input of the fully connected layer is the output of the pooling layer which is flattened into one dimensional vector. The number of neurons in the output layers are same as the number of categories or class labels we are having in our problem for classification i.e., benign and malignant thus associating features to the particular label.

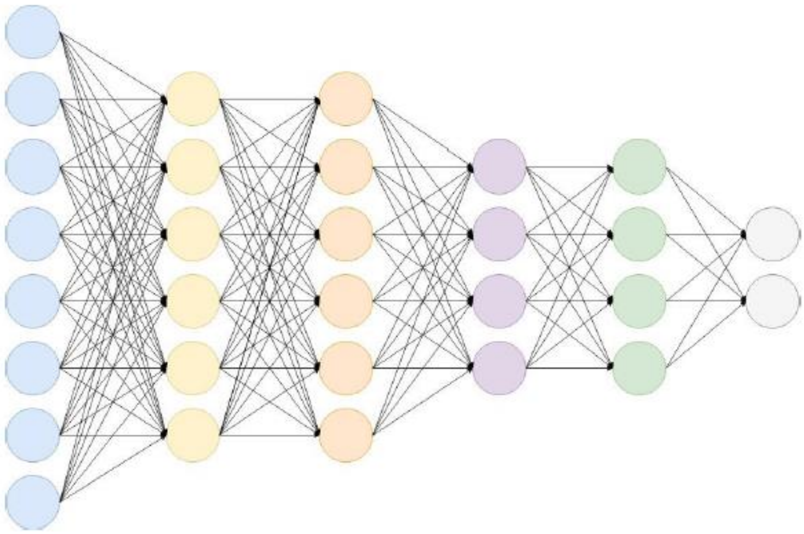

Figure 9 Fully connected layer scheme.

This process is known as forward propagation. After this process the output obtained is comparted to the actual production for error generation. Backpropagation is performed on error to update weights and bias values. After this forward and backward propagation cycle one training is completed.

3.3 Model selection:

The first and main step in our project methodology was to choose the deep learning model for Breast Cancer detection. For this purpose, we had the four main models that are being commonly used. These four are given below:

3.3.1 ResNet-50:

ResNet-50 is a Convolutional Neural Network with a deeper architecture which aids it in learning complex patters and features. It consists of 50 layers and uses residual learning with shorter connections. It employs residual blocks, has skip connections and also employs both max pooling after every convolution block. It was introduced in 2015. ResNet was introduced with different number of layers like 50,101, 152 and much more but the best suited for this is ResNet-50 as it is commonly used as pre-trained model for transfer learning in medical image processing and also has been successfully applied in medical image analysis.

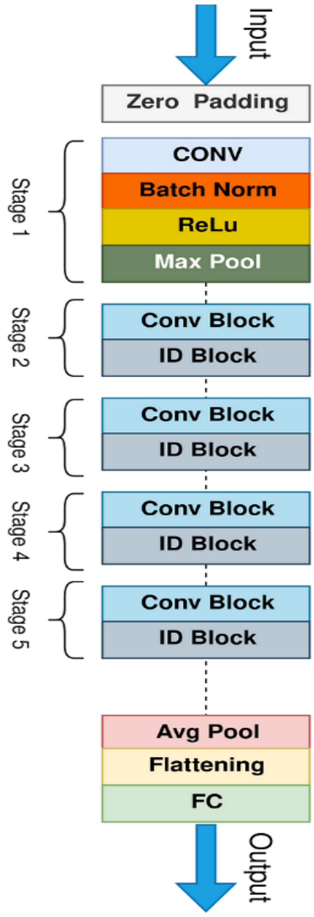

Figure 10 ResNet-50 Architecture

3.3.2. VGG-16:

VGG-16 is a most commonly used deep convolutional network which has simpler architecture comprising of convolution filters. It is also been effectively used in image classification tasks. It consists of 16-layers among which are 13 convolutional layers and 3 are fully connected layers. Each convolutional layer has 3*3 kernel size and is follower by max pooling layer. In this way it creates a deeper network with the large no of trainable parameters. It has shown remarkable performance in computer vision and serves as strong model baseline for image classification tasks.

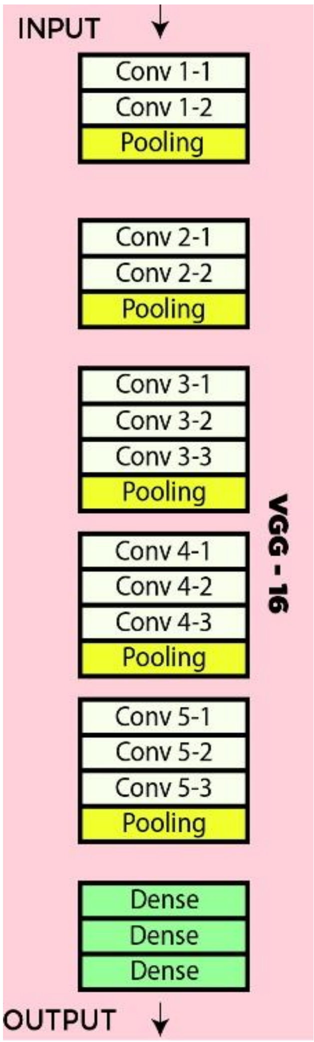

Figure 11 Vgg-16 Architecture

3.3.3. GoogleNet:

GoogleNet which is also known as inception-V1 is a Deep Convolutional Networks consisting of 22 layers developed by researchers at Google. It consisted of inception modules for efficient use of computational resources. This architecture reduced the number of parameters as compared to traditional neural networks. One of its disadvantages includes its complexity and computational costs as compared to ResNet-50 and VGG-16. Due to presence of multiple pathways, it can consume more memory during training especially when dealing with larger datasets.

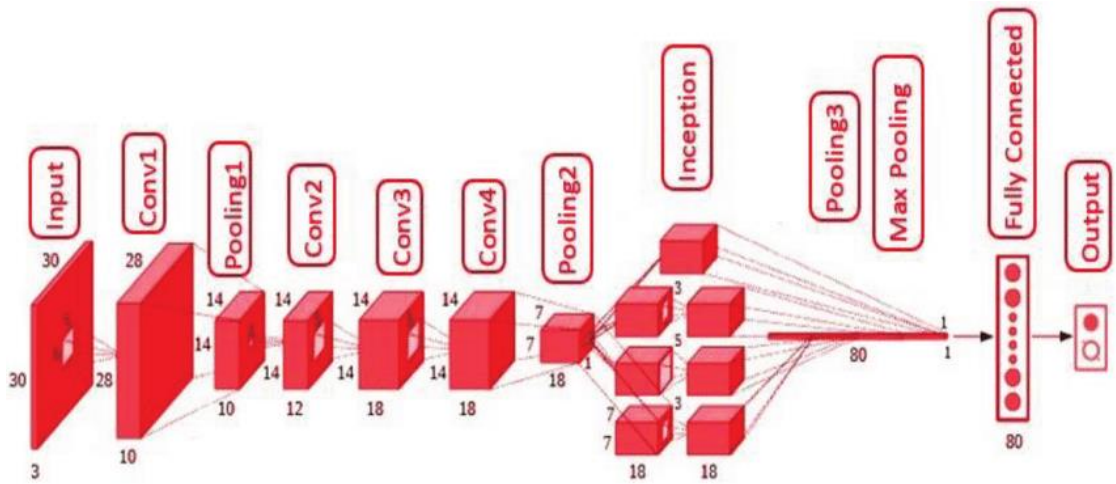

Figure 12 Google-net Architecture.

3.3.4. AlexNet:

AlexNet is a pioneering deep convolutional network which consisted of 8 convolutional layers, ReLu activation function for adding non-linearity and dropout regularization for providing overfitting. But it can be challenging task for deploying it in Real-time applications due to its complexity in neural network.

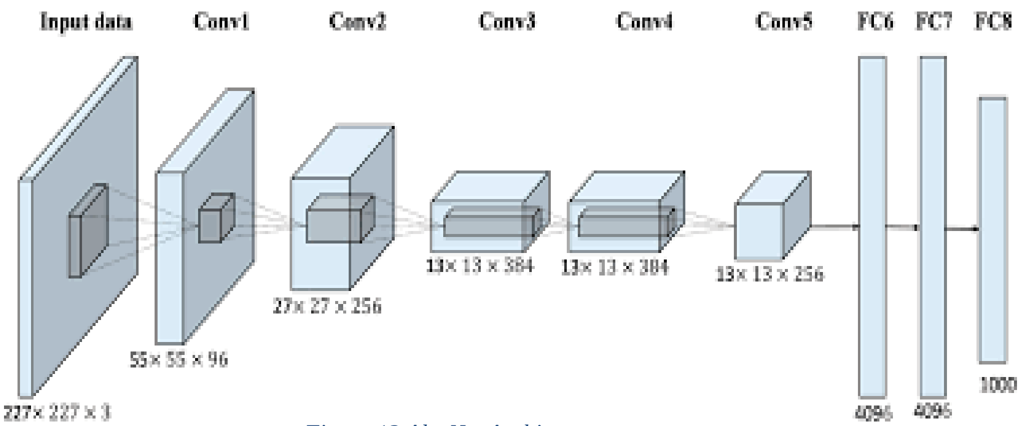

Figure 13 AlexNet Architecture

3.3.5. Final Model selection:

So, the final models which we selected for Breast cancer detection for our projects are ResNet-50 and VGG-16. This is because both play major role in medical image processing then the remain two architectures. VGG-16 comprised simple architecture and convolutional layers stacked with pooling layers for extracting meaningful and important information from medical images. Similarly, ResNet-50 with its deeper architecture and residual connections can focus more easily on mammograms images when dealing with complex features. Secondly, VGG-16 and ResNet-50 both are best suitable for transfer learning and have been pretrained on large datasets like ImageNet which can easily allow then to learn generic features from images. Whereas GoogleNet and AlexNet shallower architecture cannot provide better results for feature extraction and transfer learning especially when dealing with Breast Cancer detection in medical analysis. Due to these reasons, VGG-16 and ResNet-50 were the best suitable choices for using as the model for our project.

3.4 Overview of Dataset:

The datasets we had contained four distinct datasets of breast mammography, each with varying image counts either labelled or unlabelled and each with different and unique properties. These datasets have been obtained from Kaggle and ScienceDirect to cater to specific requirements and objectives within the project. The diversity in image count and properties across the datasets provides a comprehensive range of data for training, validation, and testing purposes, ensuring robustness and accuracy in the project's outcomes. Each of this dataset have been thoroughly checked and verified in order to proceed with them for training the model. Following is the brief overview of each and every dataset.

3.4.1 Dataset of Breast Mammography (DBM):

This dataset is the combination if 3 different datasets INbreast containing 7632 out of which 106 were original, MIAS containing 3816 images and 53 were original and DDSM has 13,128 out which 2188 were original. All of these images were labelled. Here these 3 subsets originally had lower image count so in order to enlarge the dataset data augmentation was performed i.e., rotation and flipping. Each image had different sizes so the images were resized to 227*227 pixels. These 3 subsets also comprised some unique properties. So, the overview of them is given below:

- **INBreast:**

  It contains high-resolution breast images from multiple scanners, including both raw and processed images. However, it's relatively smaller in size compared to DDSM. It is used for research purposes in breast cancer detection and analysis, but its size might limit certain types of analysis.

- **MIAS:**

This dataset consists of digitized mammograms with annotations provided by expert radiologists. It's smaller in size compared to DDSM but has detailed annotations. It is primarily used for testing and developing computer-aided diagnosis (CAD) systems due to its annotations and relatively smaller size.

- **DDSM:**

DDSM is a larger and more comprehensive dataset, containing a significant number of mammograms with detailed annotations for abnormalities like masses, calcifications, and lesions. It is widely used in the development and evaluation of algorithms for lesion detection, classification, and segmentation due to its extensive annotations and larger size.

3.4.2 Curated Breast Imaging Subset DDSM Dataset (CBIS-DDSM):

It's a curated subset of DDSM, providing a standardized version with enhancements. It contains a subset of images from DDSM, but it includes additional features such as standardized image formats and enhanced annotations. It contains curate and annotated images. The total image count is 2620. Annotations can include the information pointed out on the region of interest that aids in the evaluation of different machine learning algorithms. This dataset contained unlabeled images along with separate csv file that had the different patient ids each with unique id and other features relevant to each patient.

3.4.3 Radiological Society of North America (RSNA):

This dataset contained images that have been already used in the field of radiology and medical image processing. All of the images were labelled and obtained from regular screening. The total image count was 8013. It is diverse and well-curated dataset.

3.4.4 King Abdul Aziz University mammogram dataset (KAUMDS):

This dataset is the first digitized mammogram dataset for breast cancer in Saudi Arabia which is dependent on the Bi-Rads category. It is the labelled dataset but with the BI-RAD (Breast Imaging-Reporting and Data). It contained total 2378 images.

3.5 Final Dataset and Model selection:

After certain evaluations and conclusion, the most suitable datasets that we concluded with were DBM and RSNA. These datasets both contained labelled and the larger number of images. It was fine-tuned and clear images of mammograms. Now the next step was to choose one of these 2 datasets. We trained both datasets on the VGG-16 and ResNet-50 models using google colab in order to check their training and validation accuracies and loses so that we can choose one of two models and datasets. All the training and validation were performed with the ratio of 80% training and 20% validation. We concluded with different results which are summarized below:

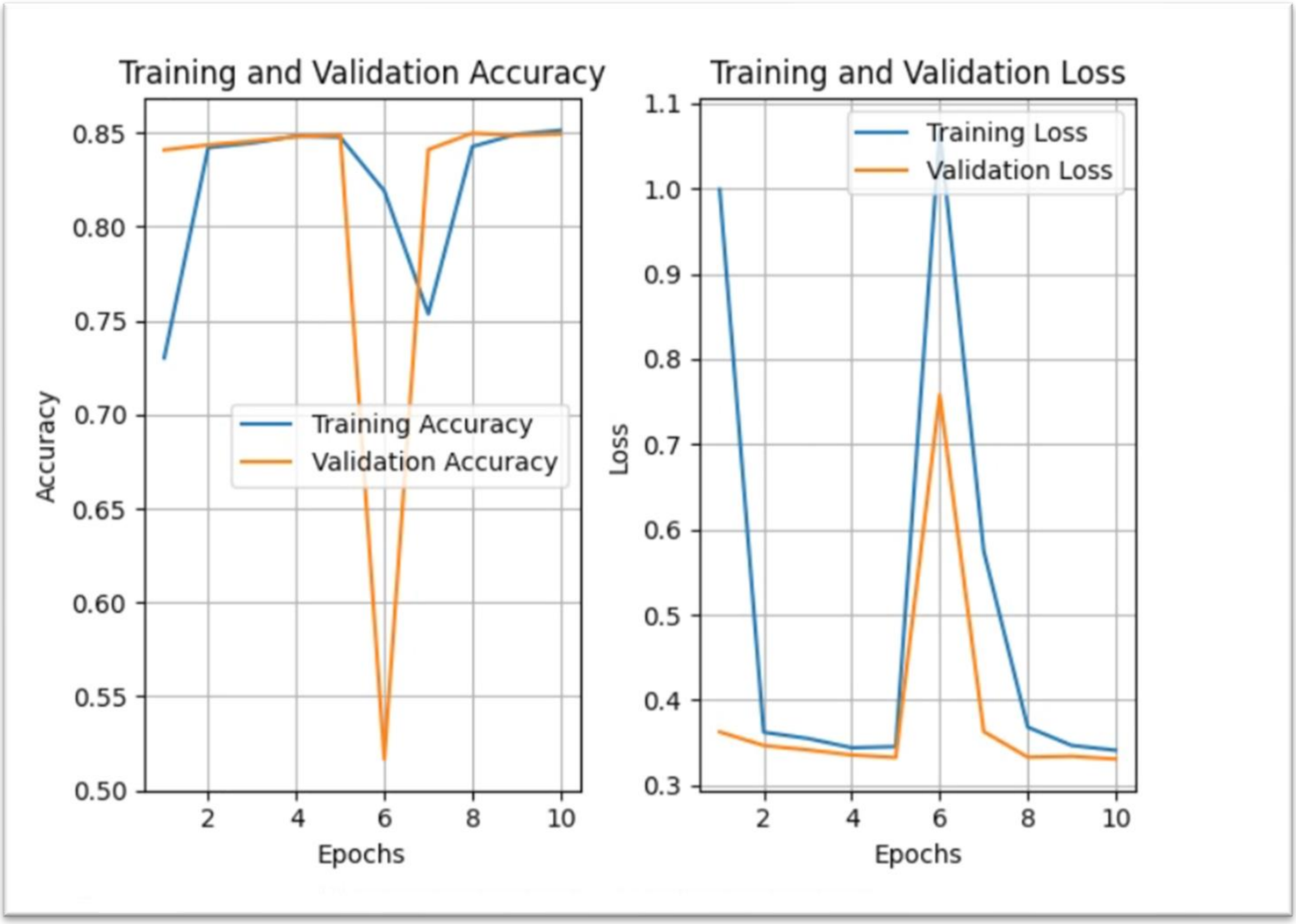

Figure 14 Vgg16 on RSNA dataset

	Training	Testing
Accuracy	85.12%	85.27%
Loss	1	0.75

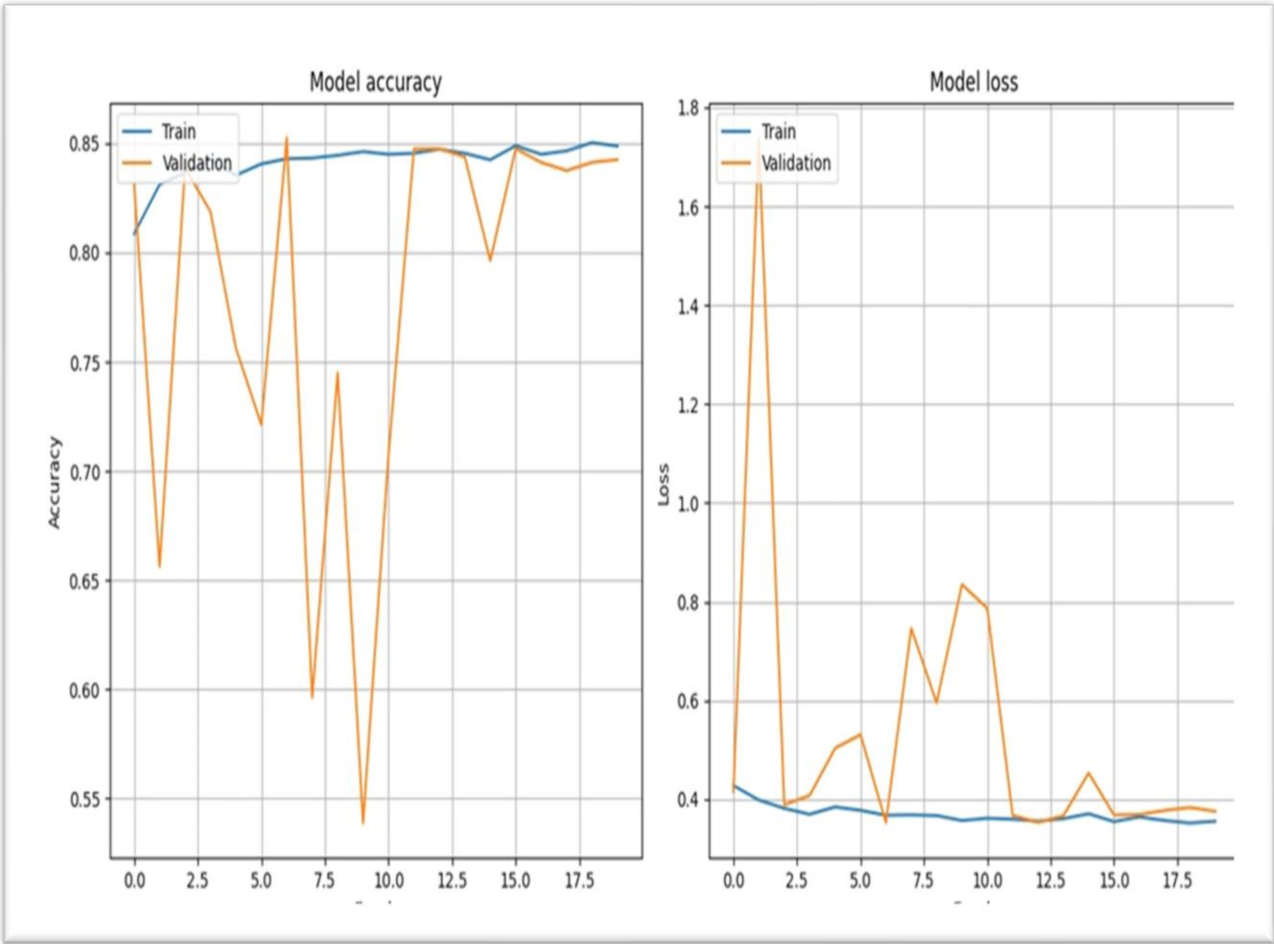

Figure 15 ResNet-50 on RSNA dataset.

	Training	Testing
Accuracy	85%	84.24%
Loss	0.3758	1.7

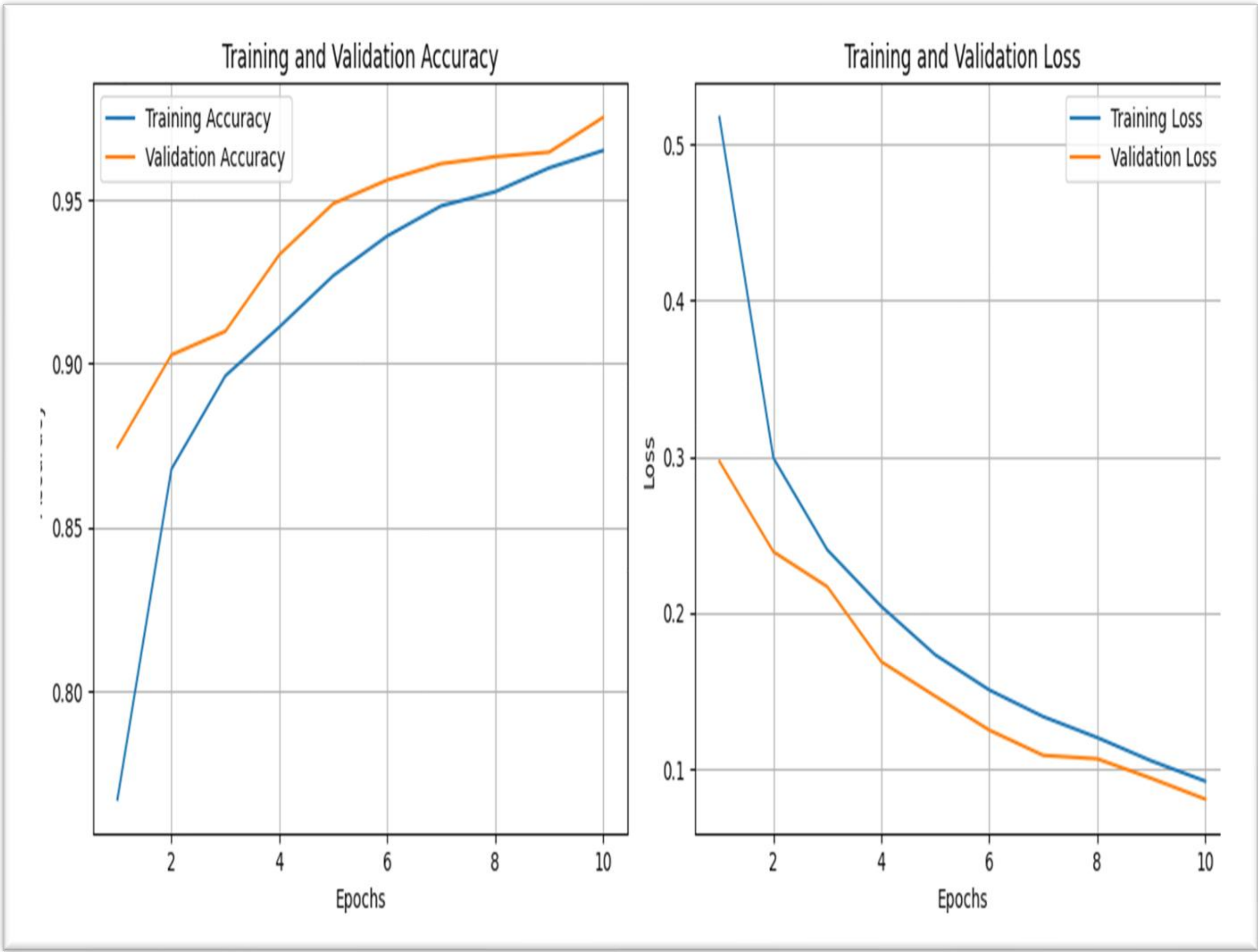

Figure 16 Vgg-16 on DBM

	Training	Validation
Accuracy	96.51%	97.52%
Loss	0.0924	0.0809

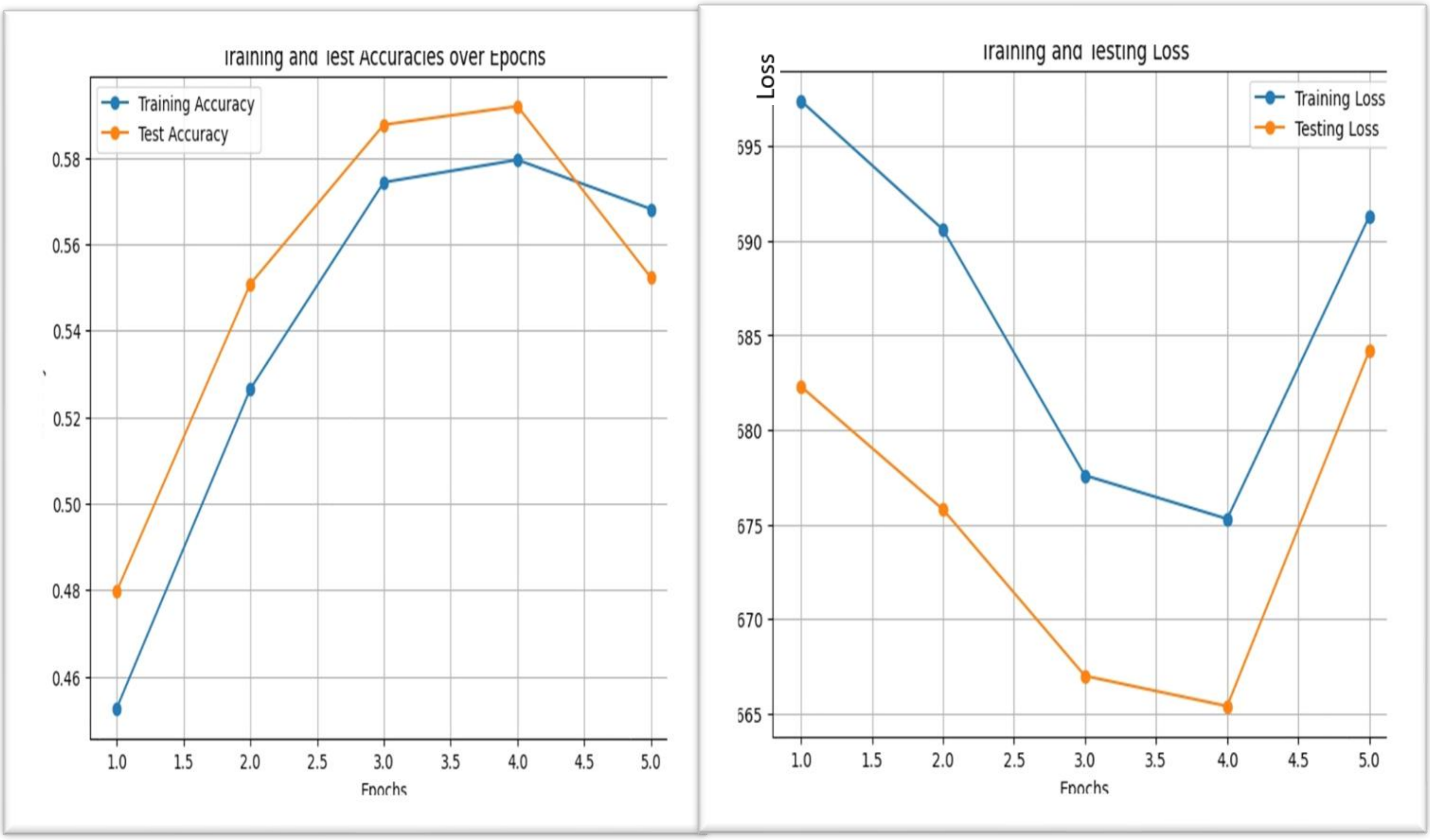

Figure 17 ResNet-50 on DBM dataset

	Training	Testing
Accuracy	58.08%	59.19%
Loss	0.7049	0.6842

Hence from above results it can easily be concluded that the VGG-16 model trained on DBM dataset worked much more efficiently than all three. It has much larger accuracy i.e., 97.52 % and also had the minimum loss. Also, the DBM dataset contained very clear mammograms which were the much larger number of images i.e., 24,576.

3.6. VGG-16 Architecture:

We have leveraged the VGG-16 model pre-trained on the ImageNet dataset. We have excluded the fully connected layers at the top and added custom layers on the top of the base model. Our model's architecture includes flattening layer to transform the base model's output into 1-D layer which is followed by the dense layer consisting of 256 neurons and ReLu activation function for future learning.

A dropout layer with the dropout value of 0.5 has been added to prevent overfitting. And in the end for binary classification dense output layer has been added with the sigmoid activation function which would predict probabilities for the presence of Breast Cancer. This approach has resulted in the model which is 97.52% accurate and highlights the efficiency of transfer learning in designing efficient and accurate architectures for medical image analysis.

- **Activation Function**

  The activation function used in dense layer is ReLu function with 256 units. ReLu is the activation used to provide non-linearity in the model by providing the output directly if the input given is positive and otherwise zero. This could help the model learn complex relationships in dataset and prevent the problem of vanishing gradient.

  The activation function which is used in the final dense layer is sigmoid function with 1 unit. The main task of this activation is to perform binary classification and here it is performing the same task. The output it provides lies in the range of 0 and 1 which is being referred as probability. In our project the output is being referred as single value probability which is 1 and being labelled as positive class.

- **Regularization**

  In order to prevent overfitting in our model 'dropout' regularization technique has also been added. Dropout with the value of 0.5 has been added after the dense layer having 256 units. The main objective of this technique is to randomly assign zero input units during training. This allows the remaining neurons to learn without relying on the dropped neurons or input units. Thus, dropout is used to break interdependent learning among neurons.

  By using above techniques, we design the VGG-16 architecture which can efficiently perform binary classification takes which mitigating the risk of overfitting.

- **Loss Function**

  The loss function being used in our model is binary cross-entropy. It is the most popularly used loss function which performs binary classification tasks. This loss function is more appropriate for our model to classify benign and malignant cases as it has optimized the model to improve its efficiency and make correct predictions for each class.

- **Optimizer**

Adam optimizer is being used in our model. Adam is referred to as Adaptive Moment Estimation which is efficiently used in Deep learning due its ability of adapting different learning rates and provide efficient convergence on wide range of problems. This property of adaptive learning rate is used to remove noise from data and handling sparse gradients. Here we have used the learning rate of ‘0.001’. This learning rate can help in controlling step size during optimization. The learning rate 0.001 is much smaller which is usually common when training Deep Neural Networks as it can ensure stable training process. Hence in this way we can prevent the chances of misclassification in our model.

3.7. Software Implementation of selected Model:

- The images have been extracted from the imported dataset zip file and categorized into benign and malignant.
- The dataset is split into training and validation sets using tensor flow’s image dataset utilities while making sure that all the images are properly pre-processed which includes resize all the images and adjusting batch sizes.
- The pre-trained VGG-16 model has been loaded by excluding its top fully connected layers in order to enhance its feature extraction capabilities.
- The custom layers haven added on the top of VGG-16 model to make it as binary classification model.
- Flatten and dense layers have been added with ReLu activation function and dropout is added for regularization and final dense layer is added with sigmoid activation function.
- The compilation of model is done using Adam optimizer and binary cross entropy loss function.
- Training and validation are performed using the splited datasets.
- The progress of training is being monitored along with accuracy and loss metrics to access its performance.
- The trained model is being saved for further use in future.
- In the end the trained model is being used for making predictions on Real-time data.
- And finally models performance is evaluated by plotting training and validation accuracy plots and confusion matrix

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 vgg16 (Functional)          (None, 7, 7, 512)         14714688

 global_average_pooling2d (  (None, 512)               0
 GlobalAveragePooling2D)

 dense (Dense)               (None, 256)               131328

 dropout (Dropout)           (None, 256)               0

 dense_1 (Dense)             (None, 1)                 257

=================================================================
Total params: 14846273 (56.63 MB)
Trainable params: 131585 (514.00 KB)
Non-trainable params: 14714688 (56.13 MB)
```

Figure 18 Model summary.

3.8 Classifier Performance index:

The ability for diagnosis of classifiers is determined by using the confusion matrix. Confusion matrix is also known as error or contingency matrix in machine learning domain. The basic design obtained for confusion matrix is shown in figure 19. In this frameworks True positives (TP) are the samples which are classified correctly and true negatives are the sample which are already negative and have been predicted correctly. False negative are the positive samples which are being misclassified as positive and False positives are the negative cases which are misclassified as positive.

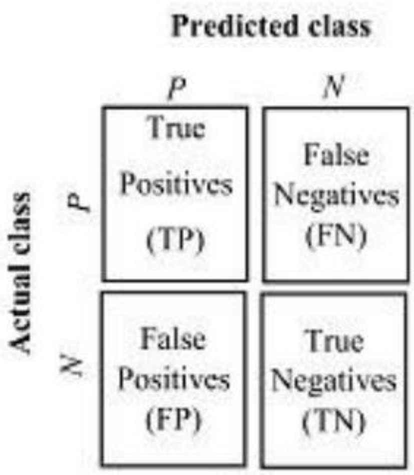

Figure 19 Basic framework of Confusion matrix

3.9. Raspberry Pi:

After model finalization, we deploy it onto raspberry pi board for making a portable device that will be used in hospital for breast cancer detection from mammograms. We have used Raspberry pi 4 which is from the fourth generation series of raspberry pi single board computers.

3.9.1. Specifications:

- **Processor:**

  The processor includes a 64-bit quad core Cortex-A72 processor that can run at speed of up to 1.5Hz. This processor provides a performance boost as compared to other models.

- **Memory (RAM):**

  We are using an 8GB LPDDR4-3200 SDRAM that can handle larger datasets and execute multiple tasks without memory constraints.

- **Storage:**

  We have used a 32GB micro-SD card for the operating system and data storage that can allow installation of software, storage of multiple files and setting up various projects.

- **Connectivity:**

  For connectivity it has 2.4 GHz and 5 GHz IEEE wireless LAN, Bluetooth 5.0, BLE Gigabit Ethernet and 2 × USB 3.0 ports, 2 × USB 2.0 ports.

- **Operating System Support:**

  Raspberry Pi OS is used to setup pi along with installing and importing files.

- **Power:**

  5V DC via USB-C connector is given to turn on the board.

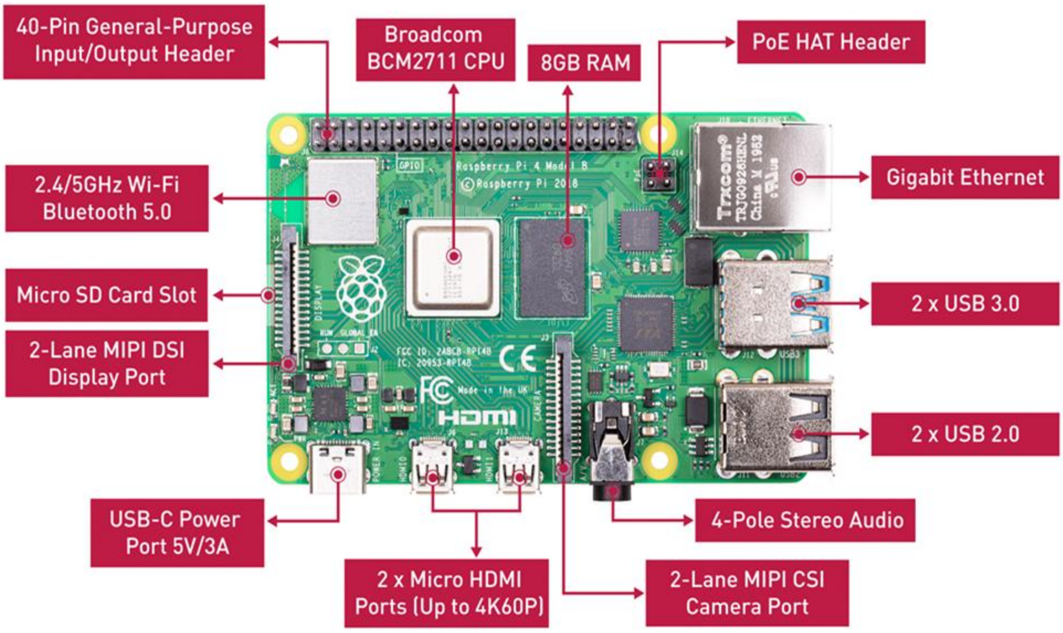

Figure 20 Specifications of Raspberry Pi

3.9.2. Raspberry Pi Setup:

We performed the following steps to setup our Raspberry Pi:

- **Hardware setup:**

First, we insert the 32GB micro-SD card onto Pi board. Next, we connect a cable to Pi and with the LCD display to set up the board.

- **Software setup:**

For software, first we install the Raspberry Pi imager utility that will write on the operating system images to the micro-SD card. Next, we install Raspberry Pi OS of version bookworm. After imager is downloaded you select your OS on it that you installed earlier. Memory option is selected on the imager, after which the OS is installed on our micro-SD card. After placing the SD card back onto Pi, we now connect it with display screen and also connect mouse and keyboard with it. We get IP address displayed on screen after which our Raspberry pi is booted.

3.9.3. Hardware Implementation:

- **LCD integration:**

For our project we had to interface our Raspberry pi with and Lcd display that will show the results produced by our model whether the image is benign or malignant. For that we use the i2c module that is interfaced with 16x2 lcd display which makes all its 16-connections multiplexed together to give a 4-connection port. So, we connect 4 pins of Lcd display to our Pi.

- **Real VNC viewer:**

We use a Virtual Networking Computer (VNC) software that helps us control Raspberry Pi desktop environment on our pc. After installing real vnc viewer we connect it with Pi by entering the raspberry pi address of our device. After connection we are able to see the desktop environment of pi on our screens.

- **Testing images:**

We insert a testing code onto the pi environment that imports our tested model.h5 file onto pi and then using testing code script it is making decisions of the input image using the prediction value it gets from the model. We insert an image onto Raspberry pi using USB port connection of Pi, and then model makes prediction and displays on lcd screen the results using the prediction value score of the image. Following is the test code integrated with lcd that we run on Raspberry pi.

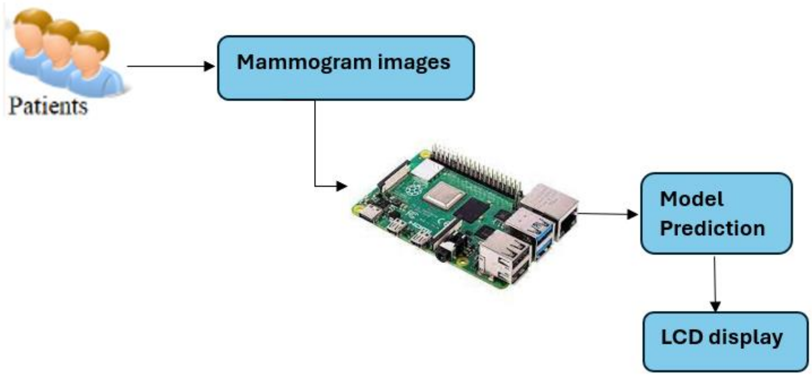

Figure 21 Hardware Methodology.

3.10. Mobile App development

Mobile app development is the next step for our project, so that our model can be deployed on user friendly application and healthcare facility can be accessed by patients anywhere in the world.

3.10.1. Mobile App development stages:

There are some generic steps and stages that we need to go through in order to make a mobile application. Following are the generic steps of app development:

1) **Strategy development**

The first step is to identify the problem for which we have to provide a solution of. After brainstorming ideas for app features and its functionalities we make a plan or strategy for the application

2) **Analysis & Planning**

In the next phase we decide the target audience, purpose and selling point for our app. Proper market research are conducted to understand the trends and user needs. Unique ideas that fills the market gap and provide solution to problems are suggested.

3) **UI/UX Design**

To make the application look appealing, this stage is important such that the user can enjoy a great experience. The architecture of the is defined and designed in this stage. Low-fidelity frameworks with basic outlines are created and after getting feedback from people/users high-fidelity prototypes are made with the final look reveal.

4) **App development**

In this stage of development, the platform on which you will be making your app along with the programming language is selected. For android apps JAVA or Kotlin are used mostly due to their easy understanding, modern features and better syntax.

5) **Front-End development**

As the name suggests front end is the front face of the app that deal with the client side functionality. This part focuses on creating visually appealing applications to provide better user experience. Using different languages and frameworks creative and interactive designs are made by designing protypes.

6) **Back-End development**

In this side, the behind-the-scenes processes of an app are done. Functionalities such as data storage, user authentication, and server-side processing are handled in this stage**.** Using server-side programming language and framework we create back-end infrastructure.

7) Mobile App Testing

After designing the app goes through different testing stages to ensure all the requirements of the app are fulfilled. Functionality testing, usability testing, compatibility testing and performance testing are done such that all the target goals are checked in the app. Testing environments like emulators, simulators and testing frameworks are set-up to test the app and debug any issues effectively.

8) Deployment

In this stage the application is prepared and released for distribution among users. After final testing and quality assurance the app is prepared for deployment. The app is sent to respective app stores for approval along with documentation that describes the complete usage and design of the app.

9) Support and performance monitoring

This stage is important for maintaining the reliability and stability of the app. Once the app is made available user-support is set up such that all sorts of feedback and issues are taken. Performance metrics and monitoring tools are made to track key performance indicators (KPIs) in the app such as crash time, loading rate etc.

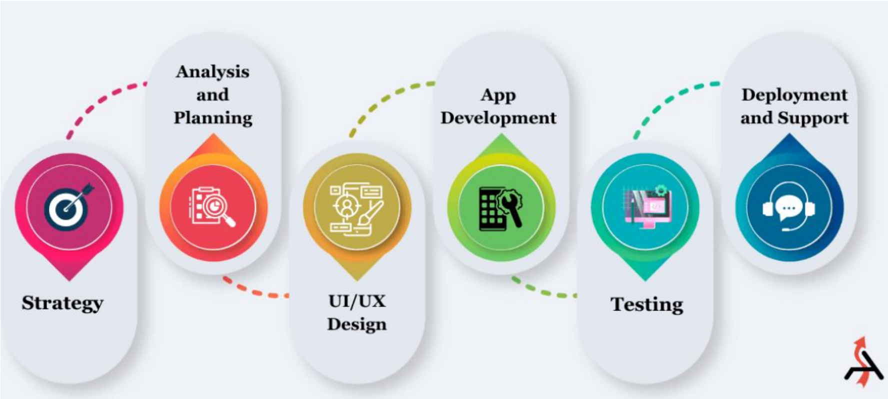

Figure 22 Mobile App Development stages

3.10.2. Our Mobile App (MammoDetect)

We followed the same mobile app development stages as discussed before. Our main aim was to design an app in which whenever a user enters their mammogram test results, the app predicts for them if the images are benign or malignant.

- **UI design**

We decided to go with Android studio as the software on which we will be making our app. Using XML layout files on Android studio we made a design for our app. Buttons, text views and image views for mammogram test results are developed.

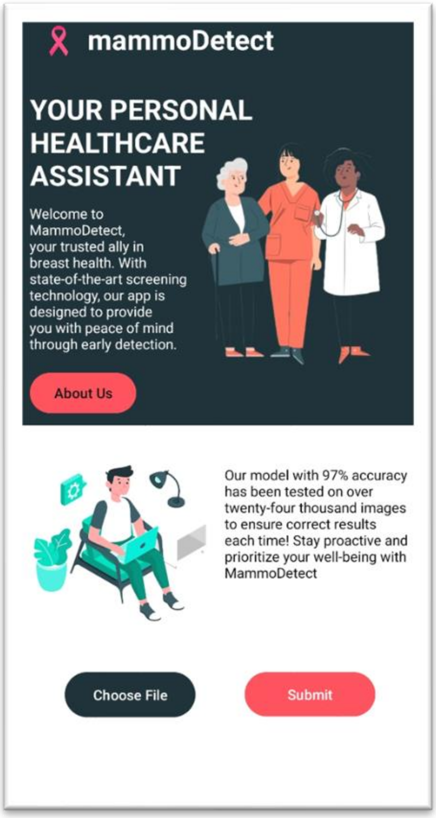

Figure 23 MammoDetect design

- **Integration with Vgg-16 API**

Our obtained Vgg-16 model that we trained on the DBM dataset with 97% accuracy is used in the app. Using code, we take the API from our model and connect it with the android app for taking mammogram images as input and sending out results.

- **Front-End development**

Using drag and drop feature of android studio, we develop the front-end part. We start by providing a button or a drop where the user will be able to upload their mammogram images. Next a display is made to show the results and prediction value scores on screen. While waiting for results a Wait message is also displayed. Action buttons for tasks like uploading another image are also displayed.

- **Back- End development**

For back-end development Kotlin language with Django as the framework was selected. An endpoint is made at the back end where we receive and process the upcoming mammogram images. Next we integrate backup with the Vgg-16 model API. After the images as analyzed and detection results are sent using API, authentication and authorization mechanism are done to make sure backend API endpoints are secure.

- **Emulator**

Once the prototype is made, we test and debug are app by using an emulator on android studio. Before sending the app to physical devices we run the emulator on android studio so that the app can be tested on wide range of virtual devices and any error can be detected at an early stage.

- **Deployment**

Using Wi-Fi feature of android studio, we can simply launch are apps on any nearby android phones with an internet connection. In this way the app is easily deployed onto android phones

3.11. Software Components:

Our project includes deep learning and computer vision approaches. These methods require different software applications for development and testing. Our models run in these programs and also have the capability to show their output in them. In this section, we will discuss the software applications, models, and software libraries we have used for our project.

3.11.1. Software Applications used:

We have used three software applications for our project namely, PyCharm, Google Colab, and Thonny IDE.

- **Google Colab**

  Google Colab (short for Collaboratory) is an online platform that provides a free Jupyter notebook environment for running and executing Python code. It is a cloud-based service offered by Google and is part of the Google Cloud ecosystem. Google Colab allows users to create and share interactive notebooks that combine code, text, and visualizations. It provides access to powerful hardware resources, including GPUs and

TPUs, enabling users to execute computationally intensive tasks. Colab integrates with other Google services like Google Drive, allowing users to easily import and export data. It also supports collaborative editing, making it ideal for teamwork and educational purposes.

Figure 24 Google Collab logo

- **Thonny IDE:**

  Thonny IDE is commonly used with Raspberry Pi as it provides a beginner-friendly environment for programming and learning Python on the Raspberry Pi platform. It provides a simplified interface, making it easy to write, debug, and run Python code. Thonny IDE offers features such as syntax highlighting, code completion, and a built-in debugger to assist learners in understanding and troubleshooting their code. It also includes a simple step-by-step execution mode to help beginners grasp the flow of program execution. Thonny IDE is cross-platform and supports various Python versions.

Figure 25 Thonny IDE logo

- **Tensor Flow:**

  Tensor flow is an open-source library developed by the Google Brain Team that is primarily used for Machine Learning and Artificial Intelligence. It has been used for variety of tasks but it mainly focused on training deep Neural Networks. It is a powerful and versatile library that has become milestone in the field of Machine Learning and Artificial Intelligence. It supports flexible architecture in order to deploy computations on different platforms such as GPUs, CPUs, TPUs and even on mobile phones. This library supports numerical computations that involves dataflow graphs. It can also support computations involving multi-dimensional arrays that are known as Tensors.

Figure 26 TensorFlow logo

- **NumPy:**

  NumPy is the important python library that is used for performing numerical computations which can support multi-dimensional arrays and matrices. It is also used for performing mathematical functions such as arithmetic, trigonometry, statistics, linear algebra and many more for operating on these arrays efficiently. These mathematical functions have the ability to perform efficiently on large number of datasets. It can provide a powerful array-based computing environment which is essential for Machine learning, scientific computations and data analysis. It is the base for many other libraries which can support scientific computing and data analysis.

Figure 27 NumPy logo

- **Keras:**

  Keras is a powerful and user friendly open-source framework that allows developers to speedily prototype and shape neural networks. With its advanced interface and flexible design, Keras removes complications of applying deep learning models, empowering operators to concentrate on the core elements of their research or application. The development procedure is sped up by the wide range of pre-built neural network layers and tools delivered by Keras. Utilizing these parts enables operators to shape complex models with little to no coding, saving both time and resources. Furthermore, Keras links effortlessly with other renowned deep learning libraries like TensorFlow, permitting effective utilization of hardware resources, which includes GPUs. Researchers control massive datasets more competently which may improve performance and accelerate convergence.

Figure 28 Keras logo

- **Android Studio**

  Android studio is a compiler, or an IDE (Integrated Development Environment) used for building mobile applications typically for Android devices like android smartphones, tablets and IoT devices etc. It is a user-friendly interface where coders can write, test and debug their front-end and back-end design. It uses languages like Java, Kotlin and python. It also designs app interfaces using drag-drop interface.

Figure 29 Android studio logo

Chapter 4- Results

4.1 Introduction

This chapter demonstrates the result of applying Convolutional Neural Network model (CNN), providing all the information regarding ROC curves, precision, accuracy, confusion matrix and the results of application deployment.

4.2 Summary of Findings

In this dissertation, we have proposed a simple and efficient method for the classification of mammography Breast Cancer images in case of large training data. After the Convolutional Neural Network has been trained on large dataset, we observe the following results:

- **Classification Accuracy** (ratio of samples which have been correctly classified): 97.5%
- **Error Rate** (ratio of samples which have been misclassified): 2.48%
- **Specificity** (ratio of real negative which are predicted negative): 97.45%
- **Sensitivity** (ratio of real positive which are predicted positive): 97.57%
- **Precision** (ratio of real positive which are predicted positive): 97.51%

From the above values it can be seen that the model has shown significantly improved accuracy for the classification of mammograms images. The model has been validated using 20% of the training data which are approximately 4916 images. That is why the Kera's feature extraction and classification of images have been performed more accurately.

Figure 31 showcases the confusion matrix obtained after training the dataset. While figure shows the training and validation accuracy curves which we have plotted after conducted the 10 epochs.

The plot of training and validation accuracies and curves have been obtained from tensor flow prediction, and along with the plot of loss over epochs is also shown. The training and validation accuracies have been increased with the increase in number of epochs which shows that the model works more efficiently when the training time has been increased.

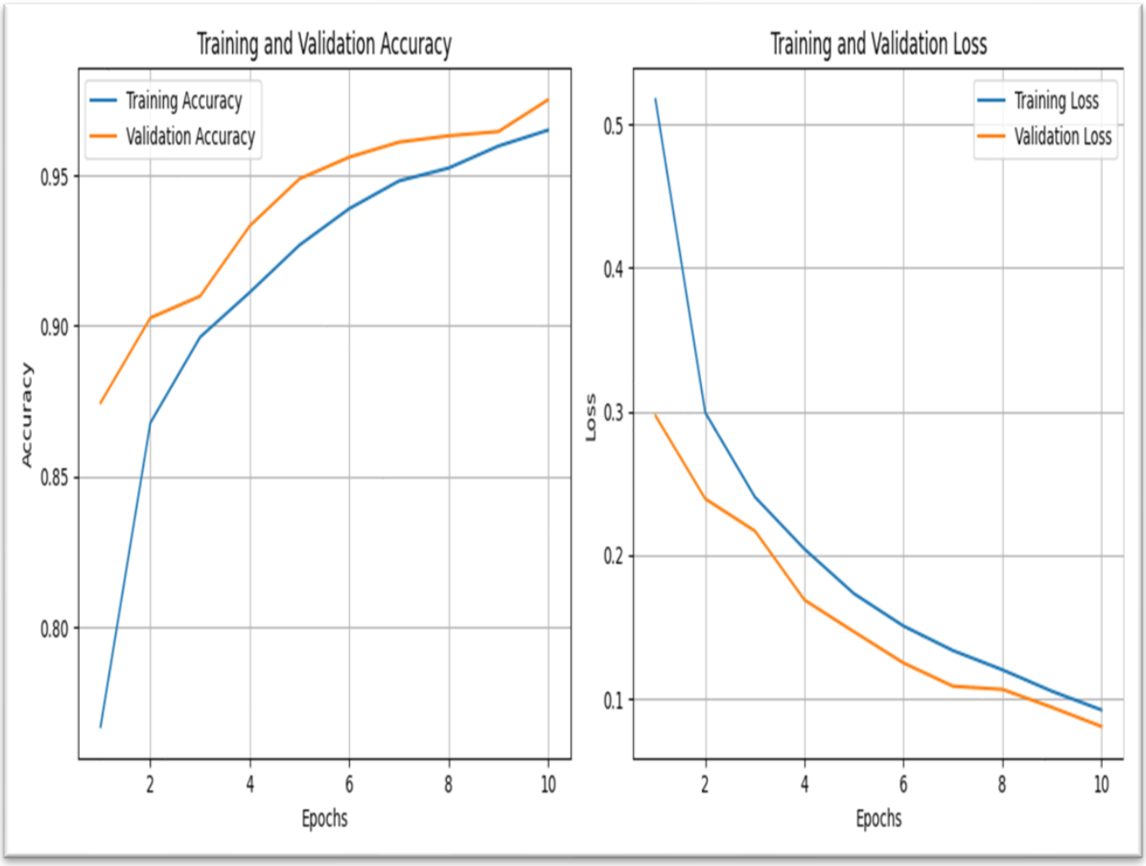

Figure 30 Training & validation Accuracy vs Training & validation loss

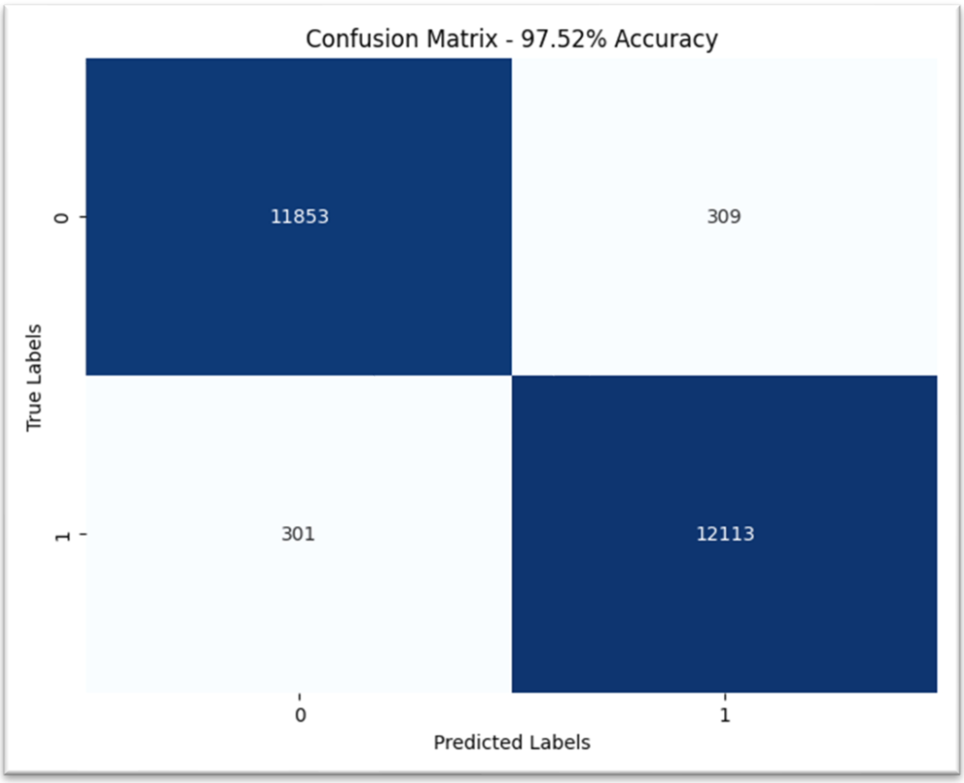

Figure 31 Final Confusion Matrix performance

4.3. Hardware Results on Raspberry Pi

Results on hardware were displayed on lcd screen which stated either the uploaded image is benign or malignant. We integrated USB with our Raspberry pi board that included real life mammogram images and testing images in a folder. These were then predicted by Pi and displayed on screen. Prediction score of our model is also displayed on screen for the accuracy of our model. Following is the running hardware of our project:

Figure 32 Raspberry Pi integrated with LCD display

Also, exact results are shown on lcd display on hardware as follows:

Figure 33 Lcd display showing results

Along with Malignant prediction, prediction value of our model is also shown. If the value is greater than 0.7 the image is malignant and otherwise the result is benign

4.4. Mobile Application results

On our mobile application the user was given an option to import their mammograms on the app. Next after some processing time, results were displayed on screen that stated whether the image is benign or malignant. Following are the results displayed:

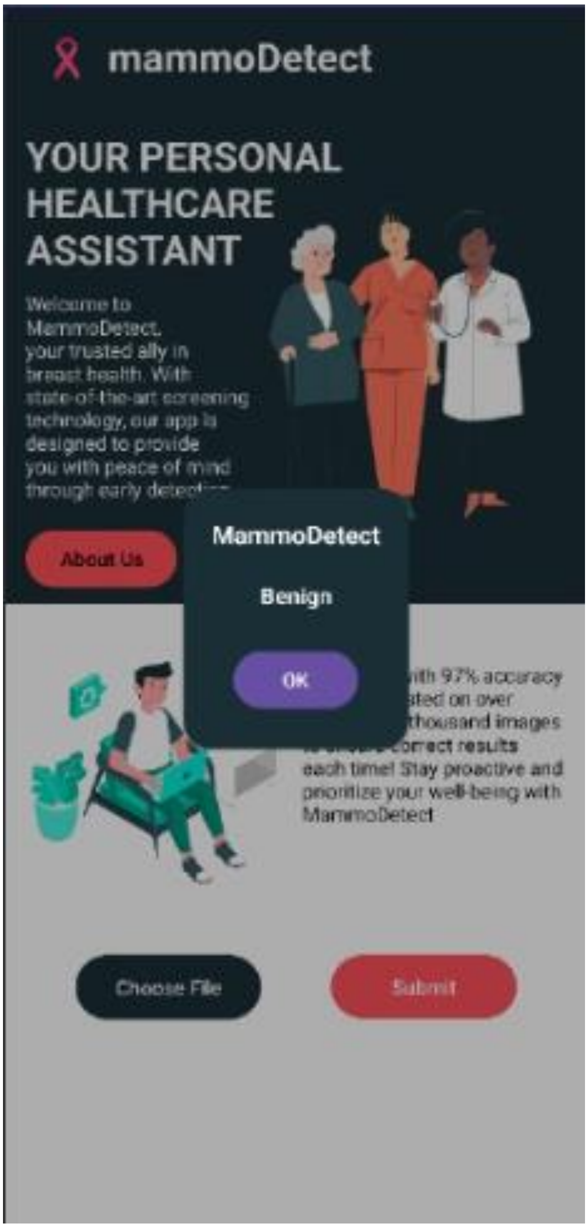

Figure 34 MammoDetect Results

When a benign image is input through gallery, after some processing time, the results of prediction by model are displayed on screen stating Benign.

Chapter 5 – CONCLUSION AND FUTURE WORK

5.1. Conclusion

In this project, we explored the detection of breast cancer from mammogram images. This detection was done using a pre-trained model of Convolutional Neural Network (CNN). After model optimization and overall evaluation, we deployed it onto Raspberry Pi for making a portable device. Lastly, we designed a mobile application that will be used in remote areas for early diagnostics and detection.

The competition of this project shows the importance of the revolution of disease detection, particularly in the field of breast cancer, by using deep learning. By utilizing deep learning technology, this project showcases efficient and accurate detection of cancer using medical images. We must acknowledge the evolution of deep learning technology and increasing availability of radiologists approved datasets of medical images.

The results of this project have so far been successful by testing out real life images obtained from actual patients. Our project ensures reliability and practical ability of detection of cancer using deep learning.

5.2. Future work suggestions

Future work can be done to make this project more successful and for possible advancements in this field. Following are the suggested things that can be done:

5.2.1. Integrating cloud computing

We can incorporate cloud computing in our project, this can be done such that the results of malignancy or benignity that are predicted by our device, can further be verified using cloud computing. A radiologist can see the results on cloud and can send a verification message to cloud that will be received by our device. This will help in improving and cross verifying the results predicted by our device.

5.2.2. Collaboration with Research labs

By collaborating with research labs, we can further get more data of different patients that will help our device get diverse imaging data and it will further improve its accuracy. Different deep learning algorithms can also be incorporated such that they can be tested on different datasets that will help in deep learning model usage on imaging data research.

5.2.3. Partnerships with diagnostic centers

We can establish partnerships with diagnostic centers to get diverse range of datasets. This will allow for further optimization and refinement of deep learning models on real world actual patient datasets. This will ensure our device's applicability in clinical research.

5.2.4. Industry collaboration

We can collaborate with companies that specialize in medical imaging technology or deep learning driven healthcare solutions. Collaboration with industry can provide an opportunity to commercialize our project for research innovation in breast cancer detection.

5.2.5. Telemedicine and remote diagnostics

With further improvement in our built mobile application, it can be used by telemedicine industries for remote breast cancer screening and diagnosis. Using cloud computing remote healthcare providers can access and verify medical images in real time, improving access to early and efficient diagnostics.

All these provided future work suggestions can lead to the development of more effective, accessible and scalable solutions for early detection and diagnosis of breast cancer, which will ultimately provide improved patient outcomes and reduced burden of disease.

REFERENCES

[1] G. S. A. A. a. A. S. Katti, "Magnetic resonance imaging (MRI)–A review," *International journal of dental clinics,* pp. 65-70, 2011.

[2] R. R. A. F. S. D. A. E. P. K. L. P. E. .. &. H. S. C. Sood, "Ultrasound for breast cancer detection globally: a systematic review and meta-analysis," *ournal of global oncology* , 2019.

[3] G. C. M. C. M. B. D. C. V. S. S. I. A. H. a. P. D. B. Wishart, "The accuracy of digital infrared imaging for breast cancer detection in women undergoing breast biopsy.," *European Journal of Surgical Oncology (EJSO) 36, no. 6,* pp. 535-540, 2010.

[4] D. a. A. K. S. Singh, "Role of image thermography in early breast cancer detection-Past, present and future.," *Computer methods and programs in biomedicine 183,* p. 105074, 2020.

[5] S. P. a. D. H. Na, "The role of various modalities in breast imaging," *Biomed Pap Med Fac Univ Palacky Olomouc Czech Repub 151.2,* pp. 209-2018, 2007.

[6] A. a. H. G. W. Bleyer, "Effect of three decades of screening mammography on breast-cancer incidence.," *ew England Journal of Medicine 367.21* , pp. 1998-2005, 2012.

[7] P. C. a. O. O. Gøtzsche, "Is screening for breast cancer with mammography justifiable?," *The Lancet 355.9198,* pp. 129-134, 2000.

[8] C. B. B. F. R. C. S. D. S. &. B. C. Dromain, "Computed-aided diagnosis (CAD) in the detection of breast cancer," *European journal of radiology 82.3,* pp. 417-423, 2013.

[9] A. T. a. S. A. E.-S. Azar, "Performance analysis of support vector machines classifiers in breast cancer mammography recognition," *Neural Computing and Applications 24,* pp. 1163-1177, 2014.

[10] B. Mahesh, "Machine learning algorithms-a review.," *nternational Journal of Science and Research (IJSR).[Internet] 9.1,* pp. 381-386, 2020.

[11] S. T. K. a. S. N. Khalid, "A survey of feature selection and feature extraction techniques in machine learning," *science and information conference. IEEE,* pp. 372-378, 2014.

[12] I. L. R. M. C. G. D. L. C. S. M. I. F. G. A. C. N. C. D. a. F. S. Castiglioni, " AI applications to medical images: From machine learning to deep learning," *Physica medica 83* , pp. 9-24, 2021.

[13] B. G. J. a. P. v. d. S. Kayalibay, "CNN-based segmentation of medical imaging data.," *arXiv preprint arXiv:1701.03056,* 2017.

[14] C. B. Q. J. W. X. L. X. Z. Y. L. a. Q. Z. Yin, "Automatic generation of medical imaging diagnostic report with hierarchical recurrent neural network.," *IEEE international conference on data mining (ICDM,* pp. 728-737, 2019.

[15] L. J. Z. A. J. H. A. A.-D. Y. D. O. A.-S. J. S. M. A. F. M. A.-A. a. L. F. Alzubaidi, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of big Data 8,* pp. 1-74, 2021.

[16] S. Tammina, "Transfer learning using vgg-16 with deep convolutional neural network for classifying images," *International Journal of Scientific and Research Publications (IJSRP) 9.10* , pp. 143-150, 2019.

[17] Y. Q. Z. Y. W. B. L. M. W. a. Y. L. Chen, "Fine-tuning ResNet for breast cancer classification from mammography.," *In Proceedings of the 2nd International Conference on Healthcare Science and Engineering 2nd,* pp. 83-96, 2019.

[18] E. L. e. a. Omonigho, "Breast cancer: tumor detection in mammogram images using modified alexnet deep convolution neural network.," *2020 international conference in mathematics, computer engineering and computer science (ICMCECS). IEEE,* pp. 1-6, 2020.

[19] Y. S. N. B. Z. O. B. a. D. L. Mednikov, "Transfer representation learning using Inception-V3 for the detection of masses in mammography," *40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC),* pp. 2587-2590, 2018.

[20] A. I. S. a. G. E. H. Krizhevsky, "mageNet classification with deep convolutional neural networks.," *Communications of the ACM 60.6,* pp. 84-90, 2017.

[21] M. A. a. M. S. A. S. Hossain, "Classification of image using convolutional neural network (CNN)," *Global Journal of Computer Science and Technology 19, no. 2* , pp. 13-14, 2019.

[22] J. W. Jolles, "Broad-scale applications of the Raspberry Pi: A review and guide for biologists.," *Methods in Ecology and Evolution 12.9* , pp. 1562-1579, 2021.

[23] M. E. A. M. a. P. K. Joorabchi, "Real challenges in mobile app development," *ACM/IEEE International Symposium on Empirical Software Engineering and Measurement. IEEE,* pp. 15-24, 2013.

[24] A. S. W. S. W. A. S. A. H. R. a. M. E. Alsolami, "King Abdulaziz University Breast Cancer Mammogram Dataset (KAU-BCMD)," *Data 6, no. 11: 111.*

[25] R. G. F. H. A. &. R. D. Sawyer-Lee, "Curated Breast Imaging Subset of Digital Database for Screening Mammography (CBIS-DDSM) [Data set]," 2016.

[26] M. Mahesh, "AAPM/RSNA physics tutorial for residents: digital mammography: an overview," *Radiographics 24.6,* pp. 1747-1760, 2004.

[27] A. Alsolami, W. Shalash, W. Alsaggaf, S. Ashoor, H. Refaat and M. Elmogy, "King Abdulaziz University Breast Cancer Mammogram Dataset (KAU-BCMD)," *Data 6, no. 11: 111,* 2021.

[28] D. a. V. W. West, "Model selection for a medical diagnostic decision support system: a breast cancer detection case," *Artificial Intelligence in medicine 20.3,* pp. 183-204.

[29] H. H. M. H. A. M. a. T. N. Asri, "Using machine learning algorithms for breast cancer risk prediction and diagnosis," *Procedia Computer Science 83,* pp. 1064-1069, 2016.

[30] S. Tammina, "Transfer learning using vgg-16 with deep convolutional neural network for classifying images," *nternational Journal of Scientific and Research Publications (IJSRP) 9, no. 10,* pp. 143-150, 2019.

[31] M. B. a. M. S. Hossain, "Low-Complexity Low-Memory VGG Models for Accurate Diagnosis of Breast Cancer.," *SoutheastCon 2024. IEEE,* 2024..

[32] P. R. D. a. M. D. Jaglan, "Breast cancer detection techniques: issues and challenges," *Journal of The Institution of Engineers (India): Series B 100.4,* pp. 379-386., 2019.

[33] D. S. a. T. A. Gowri, "A review on mammogram image enhancement techniques for breast cancer detection.," *International Conference on Intelligent Computing Applications. IEEE,* 2014.

[34] M. M. R. e. a. Khan, "Study and observation of the variation of accuracies of KNN, SVM, LMNN, ENN algorithms on eleven different datasets from UCI machine learning repository.," *4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT). IEEE,* 2018.

[35] M. a. A. Š. Muštra, "Segmentation masks for the mini-mammographic image analysis society (mini-MIAS) database.," *IEEE consumer electronics magazine 9.5,* pp. 28-33, 2020.

[36] H. e. a. Dhahri, "Automated breast cancer diagnosis based on machine learning algorithms," *ournal of healthcare engineering* , 2019.

[37] S. e. a. Khan, "A novel deep learning based framework for the detection and classification of breast cancer using transfer learning," *Pattern Recognition Letters 125 (2019),* pp. 1-6.

[38] A. e. a. Yala, "Multi-institutional validation of a mammography-based breast cancer risk model.," *Journal of Clinical Oncology 40.16,* p. 1732, 2022.

[39] A. C. L. T. S. T. P. a. R. B. Yala, "A deep learning mammography-based model for improved breast cancer risk prediction," *Radiology 292, no. 1,* pp. 60-66, 2019.

[40] W. K. Y. Y. B. T. X. H. Y. a. Y. R. Yu, "Visualizing and comparing AlexNet and VGG using deconvolutional layers," *In Proceedings of the 33 rd International Conference on Machine Learning,* 2016.

[41] T. e. a. Rai, "Can ImageNet feature maps be applied to small histopathological datasets for the

classification of breast cancer metastatic tissue in whole slide images?," *Medical Imaging 2019: Digital Pathology. Vol. 10956. SPIE,* 2019.

[42] A. A. E.-G. S. E. a. M. Z. R. Heikal, "Fine tuning deep learning models for breast tumor classification," *Scientific Reports 14, no. 1,* p. 10753, 2024.

[43] Y. B. A. F. H. a. S. T. Benhammou, "BreakHis based breast cancer automatic diagnosis using deep learning: Taxonomy, survey and insights.," *Neurocomputing 375* , pp. 9-24, 2020.

[44] A. N. a. A. S. Bokade, "Classification of Mammography Images Using Deep-CNN based Feature Ensemble Approach and its Implementation on a Low-Cost Raspberry Pi.," *International Journal of Computing and Digital Systems 13.1,* 2023.

[45] P. a. N. D. P. N. Fozin Theophile, " Thesis for: Master Degree in Telecommunications and," 2023.

[46] S. T. K. a. S. N. Khalid, "A survey of feature selection and feature extraction techniques in machine learning.," *science and information conference. IEEE,* 2014.

APPENDIX A: SUSTAINABLE DEVELOPMENT GOALS

SUSTAINABLE DEVELOPMENT GOALS FOR FYP

FYP TITLE:

Breast Cancer Detection from Mammograms using Deep Learning

FYP SUPERVISOR: Asst Prof Sobia Hayee

GROUP MEMBERS:

	REGISTERATION NUMBER	NAME
1	338890	Amna Muneer
2	347638	Fatima Zafar
3	348076	Shumaila Naveed
4	342321	Waseem Arif

SDGs:

	SDG No.	Justification after consulting
1	**1**	Our project is in accordance with the following target: **1.3**
2	**3**	Our project is in accordance with the following targets **3.1,3.7,3.8,3.d**
3	**9**	Our project is in accordance with the following targets: **9.5, 9.7**
4	**10**	Our project is in accordance with the following targets: **10.8**
5	**17**	Our project is in accordance with the following targets: **17.6,17.7,17.8**

FYP Advisor Signature: Sobia Hayee

UN website: https://focus2030.org/Understanding-the-Sustainable-Development-Goals#:~:text=The%20Sustainable%20Development%20Goals%20(SDGs,life%20for%20all%2C%20by%202030.

APPENDIX B: COMPLEX ENGINIEERING PROBLEM

Breast Cancer Detection from Mammograms using Deep Learning Models

Abstract

Breast cancer detection is a critical task in medical imaging, with mammography being the primary screening tool. In recent years, deep learning techniques have shown promise in improving the accuracy and efficiency of breast cancer detection from mammograms. This study proposes a deep learning model for breast cancer detection from mammograms, leveraging convolutional neural networks (CNNs) for feature extraction and classification. The model is trained on a large dataset of mammograms annotated by expert radiologists. Experimental results demonstrate the effectiveness of the proposed deep learning model in accurately identifying suspicious regions indicative of breast cancer, achieving competitive performance compared to traditional methods. The proposed approach holds significant potential for enhancing breast cancer screening programs, aiding radiologists in early detection and diagnosis, and ultimately improving patient outcomes.

This project defines a complex engineering problem focused on improving breast cancer detection using deep learning techniques applied to mammograms.

1. WP1 - Depth of Knowledge Required: (WK3, WK4, WK8) In-depth engineering knowledge is necessary at the level of WK3 (engineering fundamentals), WK4 (engineering specialist knowledge), and possibly WK8 (engagement with selected knowledge in the research literature) to understand and apply deep learning techniques, particularly convolutional neural networks (CNNs), to medical imaging problems like breast cancer detection.
2. WP2 - Range of Conflicting Requirements: While not explicitly mentioned in the text, there are likely conflicting requirements such as balancing the need for high sensitivity and specificity in breast cancer detection, minimizing false positives and false negatives, and optimizing computational efficiency for real-time or near-real-time analysis.
3. WP3 - Depth of Analysis Required: Developing a deep learning model for breast cancer detection requires abstract thinking and originality in analysis to formulate suitable models using CNNs and interpret the results effectively.

4. WP4 - Familiarity of Issues: While deep learning techniques have shown promise in medical imaging tasks, developing and implementing these techniques for breast cancer detection may involve addressing issues that are relatively new or infrequently encountered in traditional medical imaging practice.
5. WP5 - Extent of Applicable Codes: The application of deep learning techniques to medical imaging, while becoming more common, may still be outside the scope of existing standards and codes of practice for medical imaging and diagnostic procedures.
6. WP6 - Extent of Stakeholder Involvement and Level of Conflicting Requirements: The project involves multiple stakeholders, including radiologists, patients, healthcare providers, and regulatory agencies, each with different needs and expectations regarding breast cancer screening and detection accuracy.
7. WP7 - Interdependence: The development and implementation of a deep learning model for breast cancer detection require the integration of various component parts, including data preprocessing, model training, validation, and testing, all of which are interdependent on each other for the successful deployment of the solution.
8. In summary, this project represents a complex engineering problem that requires a deep understanding of engineering fundamentals, expertise in deep learning techniques, and interdisciplinary collaboration to improve breast cancer detection accuracy and efficiency from mammograms.

	WP1						WP2	WP3	WP4	WP5	WP6	WP7
	WK3	WK4	WK5	WK6	WK7	WK8						
PLO1 (WA1)	X											
PLO2 (WA2)		X						X				
PLO3 (WA3)												
PLO4 (WA4)						X			X			
PLO5 (WA5)									X			
PLO6 (WA6)												
PLO7 (WA7)											X	
PLO8 (WA8)												

APPENDIX C: CODE

```
from google.colab import drive
drive.mount('/content/drive')
```

```
import zipfile

# Path to the uploaded ZIP file
zip_file_path = '/content/drive/MyDrive/DDSM.zip'

# Directory where you want to extract the images
output_directory = '/content/drive/MyDrive/DDSM'

# Create the output directory if it doesn't exist
import os
os.makedirs(output_directory, exist_ok=True)

# Extract the contents of the ZIP file
with zipfile.ZipFile(zip_file_path, 'r') as zip_ref:
    zip_ref.extractall(output_directory)

# List the files in the extracted directory (optional)
extracted_files = os.listdir(output_directory)
print("Extracted files:", extracted_files)
```

```
import os

# Specify the path to the directory containing the images
directory_path ='/content/drive/MyDrive/DDSM/DDSM/benign'

# List all the files in the directory
file_list = os.listdir(directory_path)

# Initialize a counter for the image files
image_count = 0

# List of common image file extensions
image_extensions = ['.jpg', '.jpeg', '.png', '.gif', '.bmp', '.tif', '.tiff']

# Loop through the files and count the image files
for filename in file_list:
    # Check if the file has a known image file extension
    if any(filename.lower().endswith(ext) for ext in image_extensions):
        image_count += 1

print(f'Total number of images in the directory: {image_count}')
```

```
] CATEGORIES = ['benign', 'malignant']
  import PIL
  benign = file_list.count('benign/*')
  malignant = file_list.count('malignant/*')
```

```python
import os
# Specify the paths to the 'malignant' and 'benign' category directories
malignant_category_directory = '/content/drive/MyDrive/DDSM/DDSM/malignant'
benign_category_directory =  '/content/drive/MyDrive/DDSM/DDSM/benign'
# Count the number of images in the 'malignant' category
malignant_image_count = len(os.listdir(malignant_category_directory))

# Count the number of images in the 'benign' category
benign_image_count = len(os.listdir(benign_category_directory))
```

```python
import os

# Specify the path to the directory containing the labeled images
source_directory = '/content/drive/MyDrive/DDSM/DDSM'

# Create directories for 'malignant' and 'benign' categories
malignant_category_directory = '/content/drive/MyDrive/DDSM/DDSM/benign'
benign_category_directory = '/content/drive/MyDrive/DDSM/DDSM/malignant'

os.makedirs(malignant_category_directory, exist_ok=True)
os.makedirs(benign_category_directory, exist_ok=True)

# Iterate through the images in the source directory
for filename in os.listdir(source_directory):
    source_path = os.path.join(source_directory, filename)

    # Check if the filename contains "malignant" and categorize as "malignant"
    if "malignant" in filename:
        destination_path = os.path.join(malignant_category_directory, filename)


    # Check if the filename contains "benign" and categorize as "benign"
    elif "benign" in filename:
        destination_path = os.path.join(benign_category_directory, filename)

print("Images have been categorized as 'malignant' and 'benign'.")
```

```python
print(f'Number of "malignant" images: {malignant_image_count}')
print(f'Number of "benign" images: {benign_image_count}')
```

```
import tensorflow as tf
import os
from PIL import Image
from tensorflow.keras import layers, models
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import VGG16
from tensorflow.keras import optimizers
# Define your data directory
directory_path = '/content/drive/MyDrive/DDSM/DDSM'

# Define parameters
img_height = 224
img_width = 224
batch_size = 128
validation_split = 0.2  # 20% of data for validation

# Create the training dataset
train_ds = tf.keras.utils.image_dataset_from_directory(
  directory_path,
  validation_split=validation_split,
  subset="training",
  seed=123,
  image_size=(img_height, img_width),
  batch_size=batch_size)

# Create the validation dataset
validation_ds = tf.keras.utils.image_dataset_from_directory(
  directory_path,
  validation_split=validation_split,
  subset="validation",
  seed=123,
  image_size=(img_height, img_width),
  batch_size=batch_size)
# Load VGG16 model
base_model = VGG16(weights='imagenet', include_top=False, input_shape=(224, 224, 3))

# Fine-tuning the pre-trained model
base_model.trainable = True
```

```python
fine_tune_at = 100  # Fine-tune from this layer onwards
for layer in base_model.layers[:fine_tune_at]:
    layer.trainable = False

# Model creation with data augmentation
model = models.Sequential([
    base_model,
    layers.GlobalAveragePooling2D(),
    layers.Dense(256, activation='relu'),
    layers.Dropout(0.5),
    layers.Dense(1, activation='sigmoid')
])
# Compile the model
model.compile(optimizer=optimizers.Adam(learning_rate=0.001), loss='binary_crossentropy', metrics=['accuracy'])
history = model.fit(
  train_ds,
  epochs=10,
   validation_data=validation_ds,
  )
from keras.models import load_model
model.save('/content/drive/MyDrive/model.h5')
```

```python
# Test the model and evaluate accuracy
loss, accuracy = model.evaluate(validation_ds)
print("Validation Accuracy:", accuracy)
```

```python
from google.colab import drive
from tensorflow.keras.preprocessing import image
import numpy as np


# Path to the image in your Google Drive
image_path = '/content/drive/MyDrive/Benign (13).png'

# Load the image
img = image.load_img(image_path, target_size=(224, 224))
img_array = image.img_to_array(img)
img_array = np.expand_dims(img_array, axis=0)

# Preprocess the image (rescale between 0 and 1)
img_array /= 255.

# Make predictions
prediction = model.predict(img_array)

# Convert prediction to human-readable format
if prediction[0][0] >= 0.7:
    print("The image is predicted to be malignant.")
else:
    print("The image is predicted to be benign.")

print (prediction)
```

```
import matplotlib.pyplot as plt
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']

loss = history.history['loss']
val_loss = history.history['val_loss']
epochs=10
epochs_range = range(epochs)

plt.figure(figsize=(8, 8))
plt.subplot(1, 2, 1)
plt.plot(epochs_range, acc, label='Training Accuracy')
plt.plot(epochs_range, val_acc, label='Validation Accuracy')
plt.legend(loc='lower right')
plt.title('Training and Validation Accuracy')

plt.subplot(1, 2, 2)
plt.plot(epochs_range, loss, label='Training Loss')
plt.plot(epochs_range, val_loss, label='Validation Loss')
plt.legend(loc='upper right')
plt.title('Training and Validation Loss')
plt.show()
```

APPENDIX D: SDG REPORT

BREAST CANCER DETECTION FROM MAMMOGRAMS USING DEEP LEARNING

by Sobia Hayee

Submission date: 16-May-2024 01:27AM (UTC-0700)
Submission ID: 2380882478
File name: BREAST_CANCER_DETECTION_FROM_MAMMOGRAMS_USING_DEEP_LEARNING.txt (55.77K)
Word count: 8512
Character count: 46914

BREAST CANCER DETECTION FROM MAMMOGRAMS USING DEEP LEARNING

ORIGINALITY REPORT

9%	%	9%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Sanket Agrawal, Rucha Rangnekar, Divye Gala, Sheryl Paul, Dhananjay Kalbande. "Detection of Breast Cancer from Mammograms using a Hybrid Approach of Deep Learning and Linear Classification", 2018 International Conference on Smart City and Emerging Technology (ICSCET), 2018 Publication	2%
2	Vivek Lahoura, Harpreet Singh, Ashutosh Aggarwal, Bhisham Sharma et al. "Cloud Computing-Based Framework for Breast Cancer Diagnosis Using Extreme Learning Machine", Diagnostics, 2021 Publication	1%
3	"Communication and Intelligent Systems", Springer Science and Business Media LLC, 2024 Publication	1%
4	Mehvish Firdous, Sanjay Sharma, Anuj Mahajan. "Chapter 31 Breast Cancer	1%

Detection from Mammograms Using Deep Learning and Bio-inspired Optimization Algorithm", Springer Science and Business Media LLC, 2024
Publication

5	Abeer Heikal, Amir El-Ghamry, Samir Elmougy, M. Z. Rashad. "Fine tuning deep learning models for breast tumor classification", Scientific Reports, 2024 Publication	1%
6	"Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2020", Springer Science and Business Media LLC, 2021 Publication	<1%
7	Ashwin Pajankar. "Practical Linux with Raspberry Pi OS", Springer Science and Business Media LLC, 2021 Publication	<1%
8	Dina A. Ragab, Maha Sharkas, Stephen Marshall, Jinchang Ren. "Breast cancer detection using deep convolutional neural networks and support vector machines", PeerJ, 2019 Publication	<1%
9	"Model and Data Engineering", Springer Science and Business Media LLC, 2024 Publication	<1%

10	Yassine, Barhoumi. "Efficient Scopeformer: Towards Scalable and Rich Feature Extraction for Intracranial Hemorrhage Detection Using Hybrid Convolution and Vision Transformer Networks", Rowan University, 2023 Publication	<1 %
11	Mamta Patel, Mehul Shah. "Deep Ensemble Learning Model for Diagnosis of Lung Diseases from Chest X -Ray Images", Indian Journal Of Science And Technology, 2024 Publication	<1 %
12	Springer Series in Bio-/Neuroinformatics, 2015. Publication	<1 %
13	Khaoula Belhaj Soulami, Naima Kaabouch, Mohamed Nabil Saidi. "Breast cancer: Classification of suspicious regions in digital mammograms based on capsule network", Biomedical Signal Processing and Control, 2022 Publication	<1 %
14	"Artificial Intelligence in Medical Imaging", Springer Science and Business Media LLC, 2019 Publication	<1 %
15	Dhurgham Al-Karawi, Shakir Al-Zaidi, Khaled Ahmad Helael, Naser Obeidat et al. "A Review	<1 %

of Artificial Intelligence in Breast Imaging", Tomography, 2024
Publication

16	Zhenbing Zhao, Guozhi Xu, Yincheng Qi. "Chapter 41 Multi-Scale Hierarchy Deep Feature Aggregation for Compact Image Representations", Springer Science and Business Media LLC, 2017 Publication	<1 %
17	Bikram Banerjee, German Spangenberg, Surya Kant. "CBM: An IoT enabled LiDAR sensor for in-field crop height and biomass measurements", Institute of Electrical and Electronics Engineers (IEEE), 2021 Publication	<1 %
18	Chen, Danlei. "3D Food Printing: An Integrated Approach to Achieve Personalized Nutrition and Innovative Texture in Food Products", Rutgers The State University of New Jersey, School of Graduate Studies, 2023 Publication	<1 %
19	"Proceedings of the International Conference on Paradigms of Computing, Communication and Data Sciences", Springer Science and Business Media LLC, 2021 Publication	<1 %
20	Hongxia Zhang, Zhonghao Li, Jinlu Sun, Yushi Fu, Dagong Jia, Tiegen Liu. "Characterization	<1 %

of particle size and shape by an IPI system through deep learning", Journal of Quantitative Spectroscopy and Radiative Transfer, 2021
Publication

21	Ioana Udrea, Silviu Petrache, Georgeta Ionascu, Cristian Gabriel Alionte et al. "Optimizing the Placement of Indoor Thermal Climate Sensors of an Iot Solution", 2020 12th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), 2020 Publication	<1 %
2	Lecture Notes in Computer Science, 2008. Publication	<1 %
23	Wael E. Fathy, Amr S.. "A Deep Learning Approach for Breast Cancer Mass Detection", International Journal of Advanced Computer Science and Applications, 2019 Publication	<1 %
24	Yaxin Ma, Changchun Yang, Juze Zhang, Yiyun Wang, Feng Gao, Fei Gao. "Human Breast Numerical Model Generation Based on Deep Learning for Photoacoustic Imaging", 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), 2020 Publication	<1 %

25	"30th Annual San Antonio Breast Cancer Symposium – December 13–16, 2007", Breast Cancer Research and Treatment, 2007 Publication	<1 %
26	Mehta, Kunal Bhavesh. "Faster Object Detection Using Distributed Systems", University of Colorado at Boulder, 2023 Publication	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		