## BREAST CANCER DETECTION USING DEEP LEARNING

**A PROJECT REPORT**

***Submitted by***

|  |  |
| --- | --- |
| **ABINAYA T** | **[613519104002]** |
| **HARNI S** | **[613519104020]** |
| **MAHALAKSHMI R** | **[613519104025]** |
| **SHIVARANJANI N** | **[613519104037]** |

***in partial fulfilment for the award of the degree***

***of***

## BACHELOR OF ENGINEERING

## *in*

COMPUTER SCIENCE AND ENGINEERING



## GOVERNMENT COLLEGE OF ENGINEERING, DHARMAPURI ANNA UNIVERSITY: CHENNAI 600 025

MAY 2023

## BONAFIDE CERTIFICATE

Certified that this project report “**BREAST CANCER DETECTION USING DEEP LEARNING**” is the bonafide work of the following students **ABINAYA T [613519104002]**, **HARNI S [613519104020], MAHALAKSHMI R [613519104025], SHIVARANJANI N [613519104037]** who carried out the project work under my supervision.

## SIGNATURE SIGNATURE

**Dr. J. C. MIRACLIN JOYCE PAMILA Mrs. P. SUGAVANESHWARI**

## HEAD OF THE DEPARTMENT SUPERVISOR

## Assistant Professor

Department of Computer Science Department of Computer Science and Engineering, and Engineering,

Government College of Engineering, Government College of Engineering,

Dharmapuri-636704. Dharmapuri-636704.

Submitted for the university examination held at Government College of Engineering, Dharmapuri on ……………..

## INTERNAL EXAMINER EXTERNAL EXAMINER

**ACKNOWLEDGEMENT**

We are personally indebted to a number of people who gave us their useful insights to aid in our overall progress for this project. A complete acknowledgement would therefore be encyclopedic. First of all, we would like to give our deepest gratitude to our parents for permitting us to take up this course.

We feel happy to convey our kind regards and sincere thanks to our beloved **Principal Dr. V. SUMATHY, M.E, Ph.D.,** who provided her kind concern for carrying out this project work and providing suitable environment to work with.

We wish to express our sense of gratitude and sincere thanks to our **Head of the Department Dr. J. C. MIRACLIN JOYCE PAMILA, Ph.D.,** of Computer Science and Engineering for her valuable guidance in the preparation and presentation of this project.

We specially thank our project **Supervisor Mrs. P. SUGAVANAESHWARI M.E.,** Assistant professor, of Computer Science and Engineering for her valuable guidance in preparation and presentation of the project.

We thank all our Friends, Teaching, Non-Teaching staffs and our well wishers for their constant support all the time.

**ABSTRACT**

Breast cancer is one of the most common cancers, especially among women. If diagnosed early, the treatment process can be easier and more successful. For this reason, it is of great importance to reach the right result quickly in the scans performed. Deep learning algorithms are one of the methods that can be used in the diagnosis of breast cancer because they provide fast and successful analysis opportunities. A series of mammogram images are used to carry out preprocessing to convert a human visual image into a computer visual image and adjust suitable parameter for the CNN classifier. CNN has improve the accuracy toward classification on the mammogram images. Thus, the result shows that the proposed method has higher accuracy. This system is proposed for classifying benign and malignant mass tumors in breast mammography images.

## TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO** | **TITLE** | **PAGE NO** |
|  | **ABSTRACT** | **iv** |
|  | **LIST OF FIGURES** | **vii** |
|  | **LIST OF ABBREVIATIONS** | **viii** |
| **1** | **INTRODUCTION**   * 1. OVERVIEW   2. OBJECTIVE OF PROJECT | **1**  **1**  **2** |
| **2** | **LITERATURE SURVEY** | **3** |
| **3** | **SYSTEM ANALYSIS**  3**.**1 EXISITING SYSTEM  3.1.1 DISADVANTAGES  3.2 PROPOSED SYSTEM  3.2.1 ADVANTAGES | **5**  **5**  **5**  **6**  **6** |
| **4** | **SYSTEM DESIGN**  4.1 ARCHITECTURE DIAGRAM  4.2 REQUIREMENT SPECIFICATION  4.2.1 INTRODUCTION  4.2.2 HARDWARE SPECIFICATION  4.2.3 SOFTWARE SPECIFICATION | **7**  **7**  **8**  **8**  **8**  **8** |

|  |  |  |
| --- | --- | --- |
| **5** | **SYSTEM IMPLEMENTATION** | **11** |
| **6** | **CODING** | **12** |
| **7** | **SYSTEM EVALUATION** | **18** |
| **8** | **SCREENSHOTS** | **20** |
| **9** | **CONCLUSION** | **22** |
|  | **REFERENCES** | **23** |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No** | **Figure Name** | **Page No** |
| 4.1 | Architecture Diagram | 7 |
| 6.1 | ROC Curve (Training) | 16 |
| 6.2 | ROC Curve (Testing) | 16 |
| 7.1 | Model Accuracy and Loss | 19 |
| 7.2 | System Evaluation | 19 |
| 8.1 | Running in command Prompt | 20 |
| 8.2 | Front Page | 20 |
| 8.3 | Web service Home Page | 21 |
| 8.4 | Result of diagnosis | 21 |

**LIST OF ABBREVATIONS**

**CNN**  Convolutional Neural Network

**IDLE** Integrated Development And Learning Environment

**ROC curve** Receiver Operating Characteristics curve

**HTML** Hyper Text Markup Language

**CSS**  Cascading Style Sheets

**JS** Javascript

**CHAPTER 1**

**INTRODUCTION**

**1.1. Overview**

Breast cancer detection using deep learning is an approach that uses deep learning algorithms to analyze medical images, such as mammograms or ultrasound scans, to identify patterns that are indicative of breast cancer. Deep learning techniques are a type of artificial intelligence that can automatically learn to recognize complex patterns and features in data, without being explicitly programmed to do so. The most commonly used deep learning techniques for breast cancer detection are convolutional neural networks (CNNs) and recurrent neural networks(RNNs).

Recent studies have shown that deep learning methods, such as convolutional neural networks (CNNs), can improve the accuracy of breast cancer detection from mammogram images. CNNs are a type of neural network that can automatically learn and extract relevant features from images, allowing for accurate classification.

The results of the proposed approach demonstrate that deep learning CNNs can achieve high accuracy in detecting breast cancer and outperform existing methods for breast cancer detection. The use of CNNs for breast cancer detection can improve early detection rates, leading to better treatment outcomes and reduced mortality rates.

# 1.2. Objective of the Project

* + To detect breast cancer at an early stage when it is most treatable and increase the chances of survival.
  + To improve the accuracy of breast cancer detection by reducing the number of false positives and false negatives in mammogram analysis.
  + To automate the breast cancer detection process using deep learning CNN models, reducing the time and effort required for manual diagnosis.
  + To provide personalized treatment planning by predicting the response of breast cancer patients to different treatment options.
  + To improve the accuracy of breast cancer screening programs by identifying false positives and false negatives.

## CHAPTER 2 LITERATURE SURVEY

**Title**: Breast Cancer Classification From Histopathological Images Using Patch-Based Deep Learning Modeling [1]

**Author**: Irum Hirra , Mubashir Ahmad , Ayaz Hussaim , M.Usman Ashraf

**Year**: 2021

**Description**: This paper proposes a deep learning approach for breast cancer detection using neural networks. In this study, a novel patch-based deep learning method called Pa-DBN-BC is proposed to detect and classify breast cancer on histopathology images using the Deep Belief Network (DBN). Features are extracted through an unsupervised pre-training and supervised fine-tuning phase. The features extracted from the patches are fed to the model as input and the model presents the result as a probability matrix as either a positive sample (cancer) or a negative sample (background)

**Title**: Preprocessing of Breast Cancer Images to Create Datasets for Deep-CNN [2] **Author**: Abhijith Reddy Beeravolu , Sami Azam , Mirjam Jonkman

**Year**: 2021

**Description**: This research aims to propose effective image pre-processing methods that are computationally simple to implement and this methods are used on mammographic images to remove as much unwanted area as possible and enhance the details so that ROIs and regions within the ROIs can be detected easily. This will improve the performance of the neural network model when used in the real world. After processing the images with the proposed methods, the input images for the D-CNN are to be used for training, validation, and testing. Results are collected separately for each step of the implementation of the proposed methods.

**Title**: Feature-Sensitive Deep Convolutional Neural Network for Multi-Instance Breast Cancer Detection [3]

**Author**: Yan Wang , Lei Zhang , IEEE , Xin Shu , Yangqin Feng , Zhang Yi

**Year**: 2021

**Description**: The proposed method firstly uses a pre-train model with some custom layers to extract image features. Then, it adopts a feature fusion module to learn to compute the weight of each feature vector. It makes the different instances of each case have different sensibility on the classifier. Lastly, a classifier module is used to classify the fused features. The experimental results on both our constructed clinic dataset and two public datasets have demonstrated the effectiveness of the proposed method. It achieves better performance than the baseline methods on two datasets. These results prove that it can perform well on both non-standard and standard data.

**Title:** The Diagnostic Performance of Machine Learning in Breast Microwave Sensing on an Experimental Dataset [4]

**Author:** Tyson Reimer , Stephen Pistorius

**Year:** 2021

**Description:** This paper assesses the diagnostic performance of deep learning methods for tumour detection in breast microwave sensing (BMS). Methods: A convolutional neural network (CNN) was used to predict the presence of a cancerous lesion in data from experimental scans of MRI-derived phantoms. An experimental dataset containing data from 1257 scans was used. The CNN was compared to a similarly sized dense neural network (DNN) and logistic regression classifier. Only the CNN was found to obtain better-than random classification performance, indicating that the inherent locality of the sinogram data representation can be exploited for tumour-detection purposes.

## EXISTING SYSTEM

## MAMMOGRAPHY

**CHAPTER 3 SYSTEM ANALYSIS**

Mammography is the most widely used method for breast cancer detection. It involves taking X-ray images of the breast and analyzing them for any abnormalities. Computer-aided detection (CAD) systems can be used to assist radiologists in detecting abnormalities in mammograms.

## ULTRASOUND

Breast ultrasound is often used as a complementary method to mammography, particularly for women with dense breast tissue. It involves using high-frequency sound waves to create images of the breast tissue.

## MRI

Breast MRI can be used to detect breast cancer in women at high risk for the disease, as well as for evaluating the extent of cancer in the breast.

## DISADVANTAGES

* + - * Increase in False Positives and False Negatives
      * Low Accuracy
      * Limited Sensitivity
      * Inefficiency in Dense Breast Tissue

## PROPOSED SYSTEM

Breast cancer detection using CNNs involves training a deep learning model to analyze mammogram images and identify signs of breast cancer. CNNs are particularly well-suited for this task, as they can learn complex patterns and features from the images, making them highly effective at image recognition and classification tasks.

The process typically involves acquiring a large and diverse dataset of mammogram images, preprocessing the images to ensure their quality and consistency, and training a CNN model to analyze the images and identify patterns associated with breast cancer. The trained model is then validated on independent datasets to ensure its accuracy and robustness.

The use of CNNs for breast cancer detection has the potential to significantly improve the accuracy and efficiency of breast cancer diagnosis. By providing healthcare professionals with a powerful tool for early detection and treatment planning, the use of CNNs can potentially lead to better patient outcomes and improved healthcare

## ADVANTAGES

* + - * High Accuracy
      * Early Detection
      * Efficiency
      * Quality Assurance
      * Improved Health Care
      * Better Patient Outcomes

## CHAPTER-4

## SYSTEM DESIGN

* 1. **ARCHITECTURE DIAGRAM**

**Dataset**

**Data Preprocessing**

Cleaning

Transformation

**Feature Selection**

Test Data

Training Data

Implementing CNN algorithm

Prediction of cancer

**Model Generation**

Patient New Data

**Result with accuracy**

**Fig 4.1 Architecture Diagram**

* 1. **REQUIREMENT SPECIFICATION**

**4.2.1 INTRODUCTION**

The requirements specification is a technical specification of requirements for the hardware products. It is the first step in the requirements analysis process it lists the requirements of a particular hardware system including functional and performance requirements. The purpose of hardware requirements specification is to provide a detailed overview of the hardware project, its parameter and goals.

* + 1. **HARDWARE SPECIFICATION**
       1. Processor: Intel, Ryzen, AMD
       2. Installed memory (RAM): 4 GB or above
       3. Hard Disk: 100 GB or above
       4. Operating System: Windows 7,8,10,11 - 64 bit

## 4.2.3 SOFTWARE SPECIFICATION

* + - 1. Software:
* Google Colab
* Visual studio code
  + - 1. Programming language:
         1. Frontend:
* HTML
* CSS
* Javascript
  + - * 1. Backend:
* Python
* PHP

**GOOGLE COLAB**

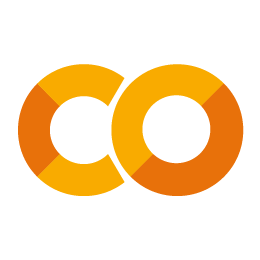
Google Colab is used in breast cancer detection as a platform for developing and running machine learning models using python.

Its purpose is to provide a cloud-based environment that facilitates the training and evaluation of models using breast cancer data.

In Google Colab, the following processes are involved:

* + - Data preprocessing
    - Model development and training
    - Visualization
    - Model evaluation

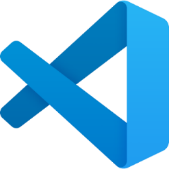
**GOOGLE COLAB**



**VISUAL STUDIO CODE**

Visual Studio code is popular choice for editing HTML files due to its extensive feature and customizability. The Front end pages for Breast cancer detection are Sign-in page, Signup page, Main page, Front page are used for visualizing the output .Model deployment is closely related to ML systems architecture, which refers to the arrangement and interactions of software components .

**VISUAL STUDIO CODE**

****

**HTML**

HTMLstands for Hypertext Markup Language. It is used for building the user interface or front end of web applications that may interact with machine learning models.

**CSS**

**C**ascading **S**tyle **S**heets, fondly referred to as **CSS**. It is primarily used in our project to enhance the visual presentation and styling of web-based user interfaces that interact with the machine learning model.

## JavaScript

JavaScript is a lightweight, cross-platform, single-threaded, and interpreted compiled programming language which is also known as the scripting language for webpages. It is used in our project to enhance the interactivity and responsiveness of user interfaces.

**Python**

Throughout this process, Python and its associated libraries, such as TensorFlow, Keras, Sklearn, Matplotlib, Pandas, Numpy are used for tasks such as data manipulation, model selection and training, evaluation and model deployment. Python's rich ecosystem of libraries and its simplicity make it a popular choice for implementing the various stages of cancer detection.

**PHP**

PHP is a recursive acronym for "PHP: Hypertext Preprocessor".

PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases.

## CHAPTER-5

## SYSTEM IMPLEMENTATION

1. **Data Collection and Preprocessing:**

Collected a large dataset of mammogram images with annotations indicating whether a patient had breast cancer or not. Preprocessed the data by resizing, normalizing, and augmenting the images to improve the accuracy of the model.

1. **Model Architecture:**

CNN is the most popular neural network architecture used for image classification tasks, including breast cancer detection. The model consist of multiple convolutional layers followed by pooling layers, and fully connected layers to learn features and classify the images.

1. **Training and Evaluation:**

Split the dataset into training, validation, and testing sets, and train the CNN model using an appropriate loss function and optimizer. Evaluated the model's performance on the validation set and fine-tune the model's hyperparameters if necessary. Finally, we evaluated the model's performance on the testing set to determine its accuracy, precision, recall, and F1-score.

1. **Deployment:**

Once the model was trained and evaluated, it was deployed in a production environment for breast cancer detection. This involve integrating the model with an existing healthcare system or building a new system from scratch.

**CHAPTER 6**

**CODING**

**6.1 CODING**

**Main.py**

import cv2

import os

import tensorflow as tf

import numpy as np

from flask import Flask, request

from flask\_cors import CORS

from flask import render\_template

from tensorflow.keras.models import load\_model

import mysql.connector

#Labeling function required for load\_learner to work

def GetLabel(fileName):

return fileName.split('-')[0]

os.environ['CUDA\_VISIBLE\_DEVICES'] = '-1'

model = load\_model('C:/Users/N SHIVARANJANI/Downloads/Breast-Cancer- Detection-main/Breast-Cancer-Detection-main/server/mammogramsclassifier.h5')

app = Flask(\_\_name\_\_)

cors = CORS(app) #Request will get blocked otherwise on Localhost

@app.route('/')

def main():

return render\_template("main.html")

@app.route('/register')

def register():

return render\_template("register.html")

@app.route('/login')

def login():

return render\_template("login.html")

@app.route('/learn')

def learn():

return render\_template("learnmore.html")

@app.route('/login\_validation', methods=['POST'])

def login\_validation():

return render\_template('index.html')

@app.route('/add\_user',methods =['POST'])

def add\_user():

return render\_template("login.html")

@app.route('/predict', methods=['GET', 'POST'])

def predict():

f = request.files['file']

img\_path = "static" + f.filename

f.save(img\_path)

img = cv2.imread(img\_path)

resize = tf.image.resize(img, (256,256)) # size accepted by the model

yhat = model.predict(np.expand\_dims(resize/255, 0)) # make the matrix items vary between 0 and 1

type= 'Normal'

score = yhat[0][2] # [[0.2 0.7 0.1]]

if yhat[0][0]>yhat[0][1] and yhat[0][0]>yhat[0][2]:

type = 'Benign'

score = yhat[0][0]

elif yhat[0][1]>yhat[0][2]:

type = 'Malignant'

score = yhat[0][1]

return f'{type} with a precision of {score\*100:.1f}%'

if \_\_name\_\_=='\_\_main\_\_':

app.run(host="0.0.0.0", port=5

**Preprocessing:**

import tensorflow as tf

from tensorflow.keras import Sequential

from tensorflow.keras.layers import Conv1D MaxPool1D,Flatten,Dense,Dropout,BatchNormalization

from tensorflow.keras.optimizers import Adam

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn import datasets,metrics

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler

from google.colab import files

data = pd.read\_csv('/content/drive/MyDrive/BMC Cancer /data.csv',index\_col=0)

data = data.drop('Unnamed: 32',axis=1)

data.diagnosis = 1 \* (data.diagnosis=='M')

def plot\_var(i,X):

name = X.columns[i]

plt.hist(X.iloc[:,i],bins=20)

plt.xlabel(name)

plt.ylabel('Number of Entries')

plt.show()

for i in range(data.shape[1]):

plot\_var(i,data)

def analyze(mod,X1,y1,X2,y2):

mod.fit(X1,y1)

y1\_pred = mod.predict(X1)

y2\_pred = mod.predict(X2)

rep1 = classification\_report(y1,y1\_pred)

rep2 = classification\_report(y2,y2\_pred)

print('Training classification report:')

print(rep1)

print('Validation classification report:')

print(rep2)

from sklearn.preprocessing import StandardScaler

from sklearn.model\_selection import train\_test\_split

y = data['diagnosis']

X = data.drop('diagnosis',axis=1)

print(X.head())

X\_train,X\_test,y\_train,y\_test = train\_test\_split(X,y,random\_state=124,

test\_size=0.3,stratify=y)

lr = LogisticRegression(C=1e10)

analyze(lr,X\_train,y\_train,X\_test,y\_test)

def analyze\_roc(mod,X1,y1,X2,y2):

ytrain\_pred = mod.predict\_proba(X1)

ytest\_pred = mod.predict\_proba(X2)

# Results have probabilities for results [0,1]

roc\_train = metrics.roc\_curve(y1,ytrain\_pred[:,1])

roc\_test = metrics.roc\_curve(y2,ytest\_pred[:,1])

roc\_auc\_train = metrics.roc\_auc\_score(y1,ytrain\_pred[:,1])

roc\_auc\_test = metrics.roc\_auc\_score(y2,ytest\_pred[:,1])

print('ROC Area-under-curve for training set: {:0.3}'.format(roc\_auc\_train))

print('ROC Area-under-curve for validation set: {:0.3}'.format(roc\_auc\_test))

fig = plt.figure(1,figsize=[6,6])

plt.plot(roc\_train[0],roc\_train[1],label='Test set',c='b')

plt.plot(roc\_test[0],roc\_test[1],label='Validation set',c='r')

plt.xlabel('False Positive Rate')

plt.ylabel('True Positive Rate')

plt.legend(loc='lower right')

plt.show()

def analyze\_prec\_recall(mod,X1,y1,X2,y2):

ytrain\_pred = mod.predict\_proba(X1)

ytest\_pred = mod.predict\_proba(X2)

# Results have probabilities for results [0,1]

roc\_train = metrics.precision\_recall\_curve(y1,ytrain\_pred[:,1])

roc\_test = metrics.precision\_recall\_curve(y2,ytest\_pred[:,1])

roc\_auc\_train = metrics.average\_precision\_score(y1,ytrain\_pred[:,1])

roc\_auc\_test = metrics.average\_precision\_score(y2,ytest\_pred[:,1])

print('Ave. precision score for training set: {:0.3}'.format(roc\_auc\_train))

print('Ave. precision score for validation set: {:0.3}'.format(roc\_auc\_test)

fig = plt.figure(1,figsize=[6,6])

plt.step(roc\_train[1],roc\_train[0],label='Test set',c='b')

plt.step(roc\_test[1],roc\_test[0],label='Validation set',c='r')

plt.xlabel('Recall')

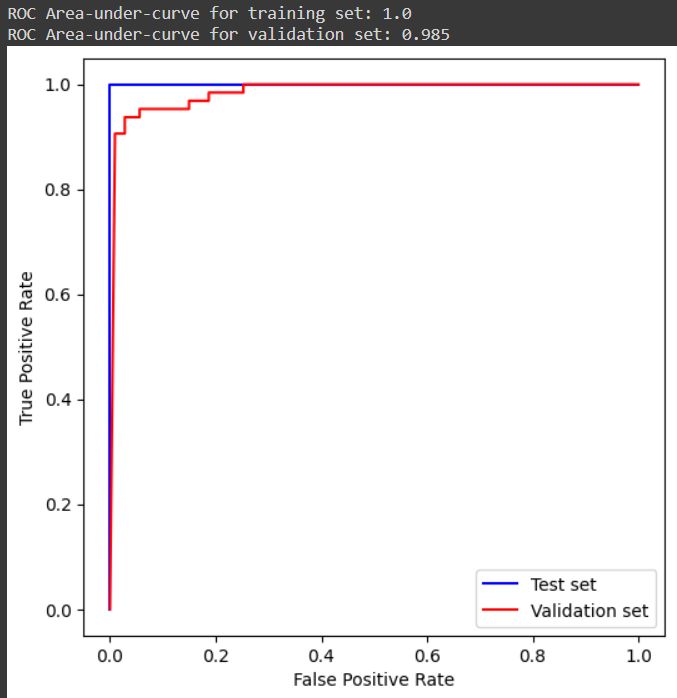
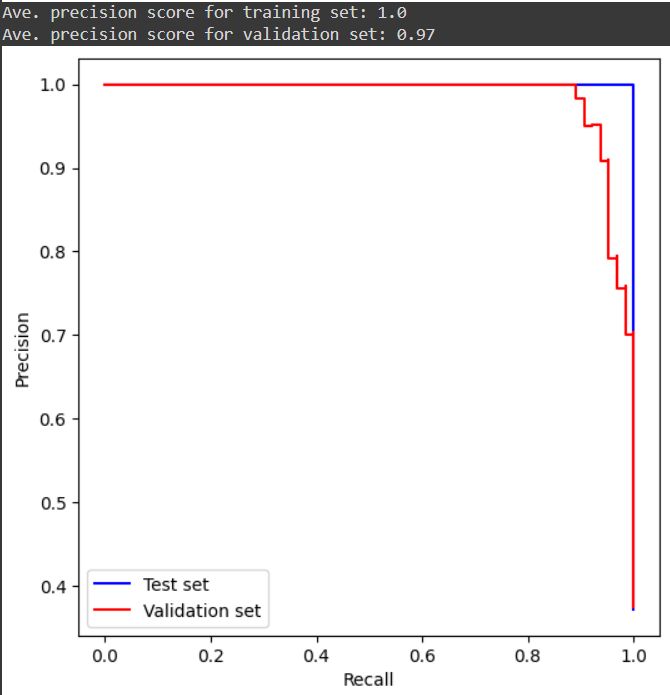
plt.ylabel('Precision')

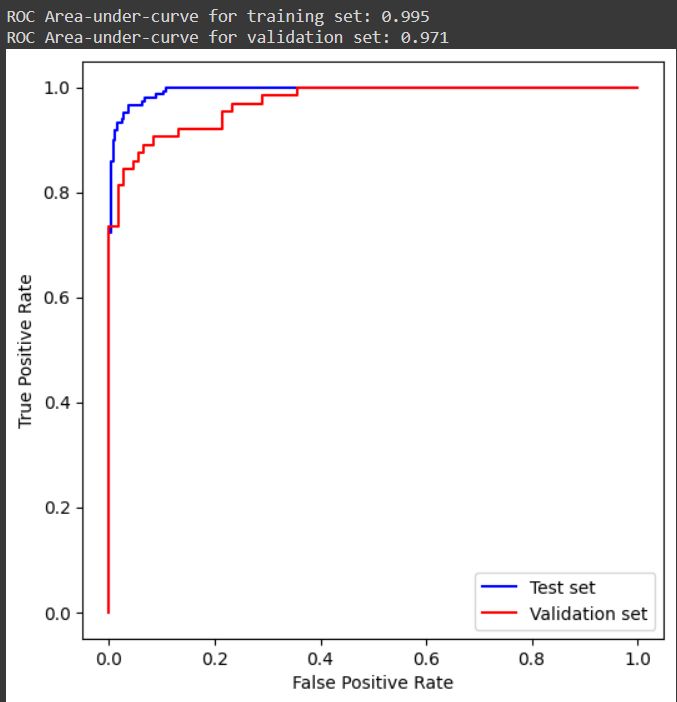
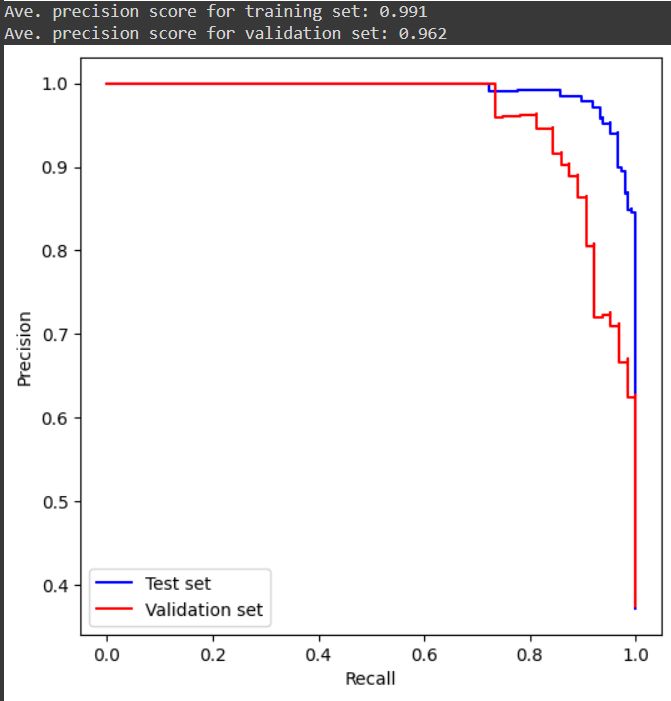
plt.legend(loc='lower left')

plt.show()

analyze\_roc(lr,X\_train,y\_train,X\_test,y\_test)

analyze\_prec\_recall(lr,X\_train,y\_train,X\_test,y\_test)





**Fig 6.1 ROC Curve(training)**

analyze\_roc(lr,X\_train,y\_train,X\_test,y\_test)

analyze\_prec\_recall(lr,X\_train,y\_train,X\_test,y\_test)

**Fig 6.2 ROC Curve (Testing)**

fig = plt.figure(1,figsize=[12,6])

ax = fig.add\_subplot(121)

ax.plot(features,train\_score,label='Training set',c='b')

ax.plot(features,val\_score,label='Validation set',c='r')

ax.set\_xlabel('Number of Features')

ax.set\_ylabel('ROC AUC Score')

ax.set\_title('Validation Plot for # of PCA Variables')

ax.legend(loc='lower right')

ax.set\_ylim([0.9,1])

ax.grid(alpha=0.2,linestyle='-')

ax = fig.add\_subplot(122)

ax.plot(features,train\_score\_2,label='Training set',c='b')

ax.plot(features,val\_score\_2,label='Validation set',c='r')

ax.set\_xlabel('Number of Features')

ax.set\_ylabel('Average Precision Score')

ax.set\_title('Validation Plot for # of PCA Variables')

ax.legend(loc='lower right')

ax.set\_ylim([0.9,1])

ax.grid(alpha=0.2,linestyle='-')

plt.show()

X\_train = X\_train.reshape(512,30,1)

X\_test = X\_test.reshape(57,30,1)

model = Sequential()

model.add(Conv1D(filters=16,kernel\_size=2,activation='relu',input\_shape=(30,1)))

model.add(BatchNormalization())

model.add(Dropout(0.2))

model.add(Conv1D(32,2,activation='relu'))

model.add(BatchNormalization())

model.add(Dropout(0.2))

model.add(Flatten())

model.add(Dense(32,activation='relu'))

model.add(Dropout(0.2))

model.add(Dense(1,activation='sigmoid'))

model.summary()

history = model.fit(X\_train,y\_train,epochs=35,verbose=1,validationdata=(X\_test,y\_test))

## CHAPTER 7

## SYSTEM EVALUATION

System evaluation for breast cancer detection using CNN typically involves measuring the accuracy, recall, f1 score, precision and area under the receiver operating characteristic (ROC) curve of the model.

**Accuracy:**

It can be determined as the number of correct predictions to the total number of predictions.

**Accuracy** = Number of correct predictions

Total number of predictions

**Precision:**

Precision measures the proportion of true positives out of all predicted positives.

**Precision** = TP

(TP + FP)

**Recall:**

Recall measures the proportion of true positives out of all actual positives.

**Recall** = TP

(TP + FN)

**F1-score:**

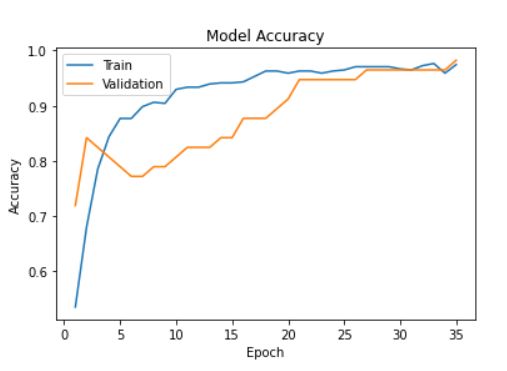
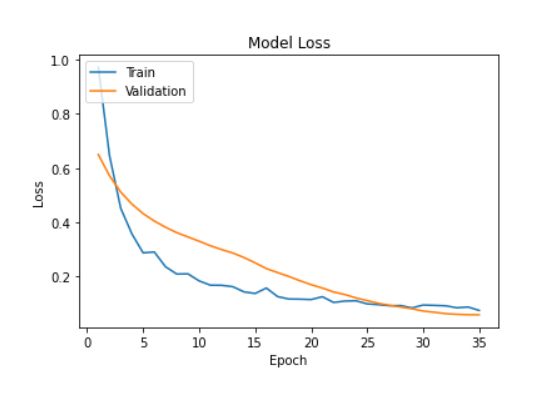
The F1-score is a weighted average of precision and recall.

Precision x Recall

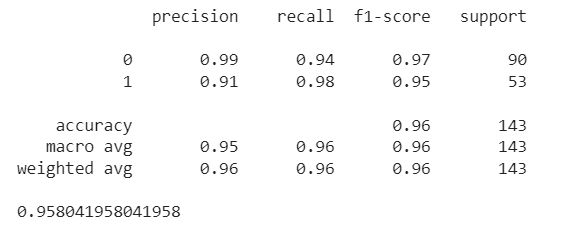
**F1-score** = 2 x

Precision + Recall

* The confusion matrix provides a more detailed breakdown of true positives, true negatives, false positives, and false negatives

****

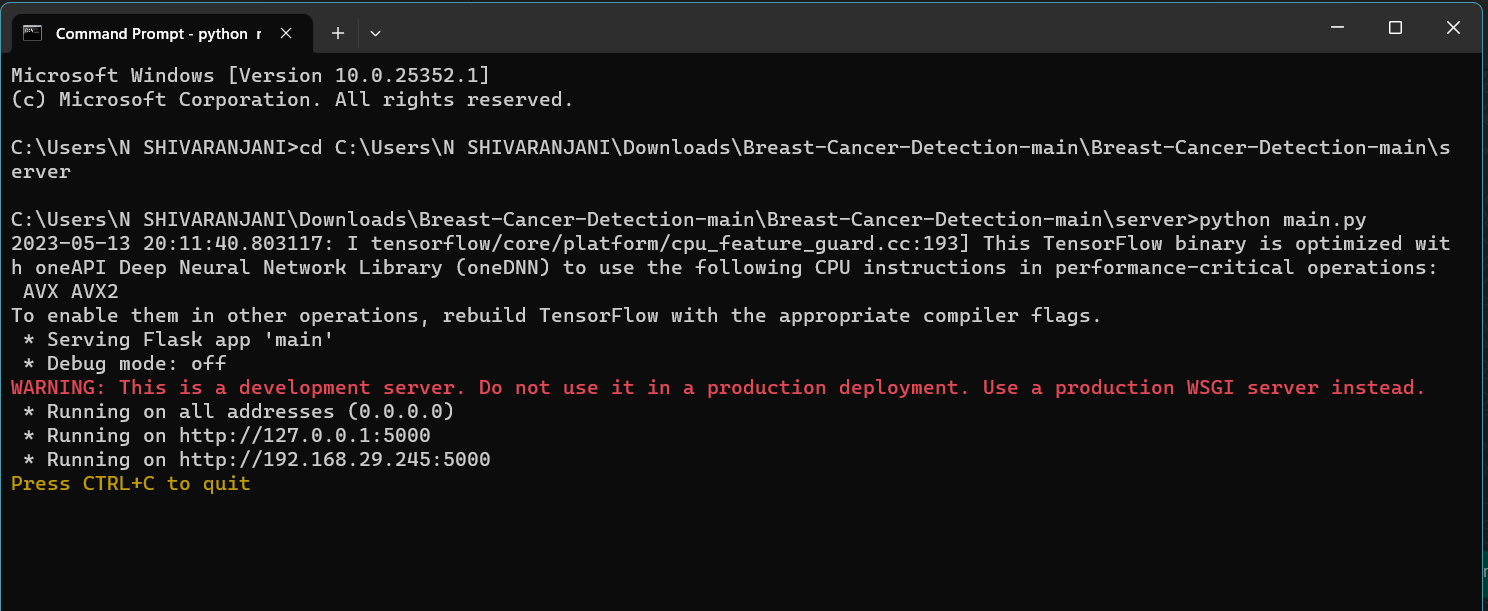
**Fig 7.1 Model accuracy and loss**

****

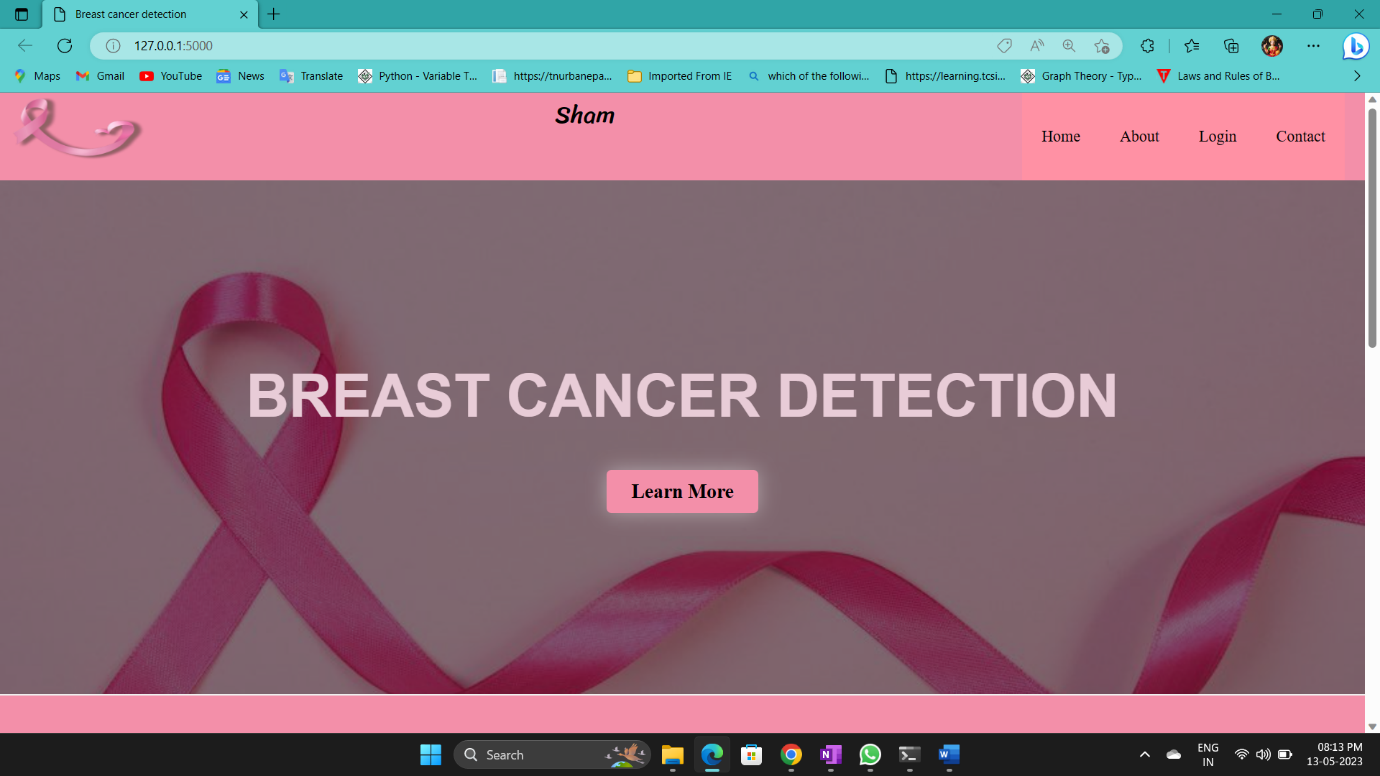
**Fig 7.2 System Evaluation**

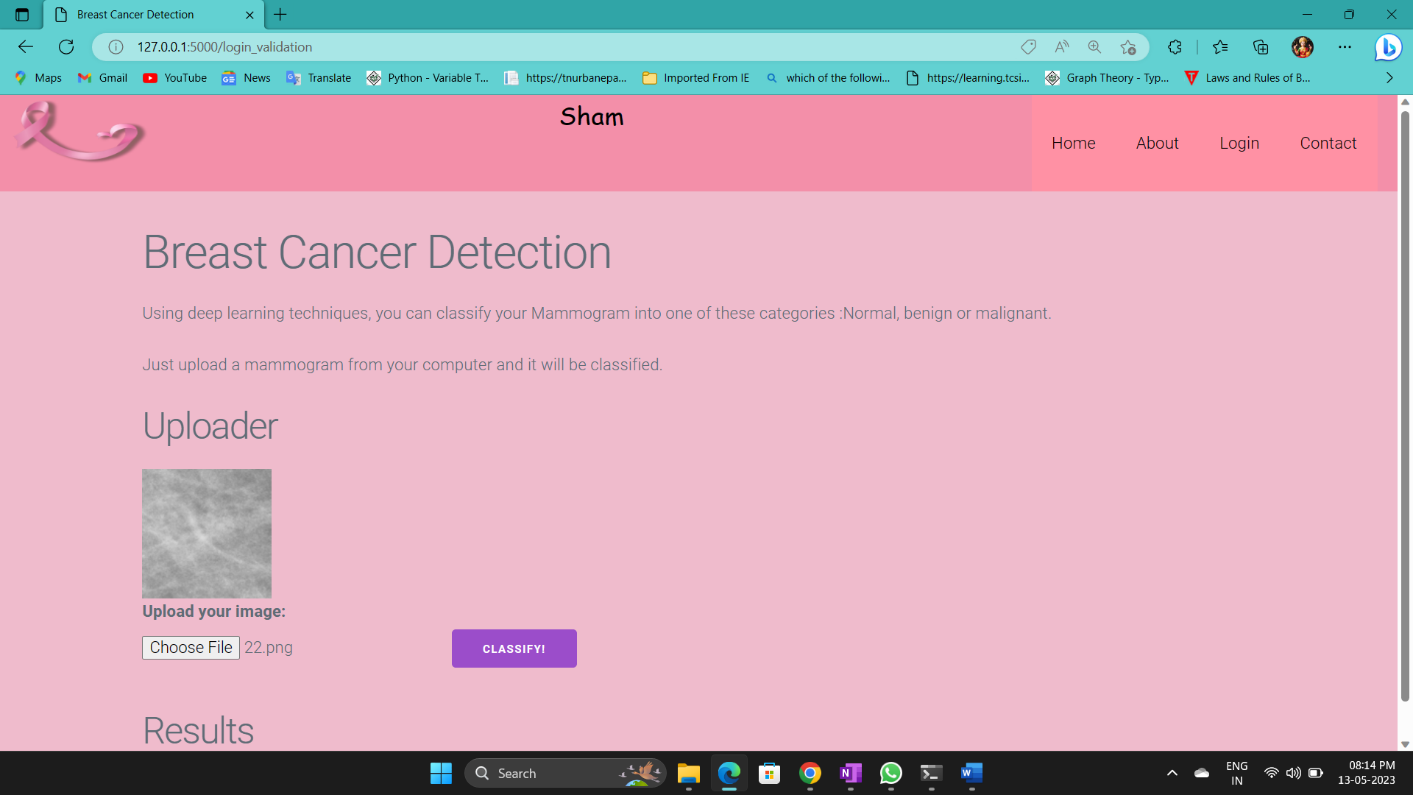
**CHAPTER 8**

## SCREENSHOTS



**Fig 8.1 Running in Command Prompt**

 **Fig 8.2 Front page**



## Fig 8.3 Web Service Homepage

## 

**Fig 8.4 Result of diagnosis**

**CHAPTER 9**

**CONCLUSION AND FUTURE ENHANCEMENT**

**CONCLUSION:**

This paper demonstrates the potential diagnostic utility of machine learning methods for automated tumor-detection in breast microwave sensing using an experimental dataset. The dataset consisted of scans of Mammogram images. To make better use of the different contributions of each instance in each case, a feature-sensitive deep CNN method was also proposed. According to our results, it is observed that the model is better than the models that work using whole images for feature extraction. This reduced the computational cost and also provides high accuracy and classify the benign and malignant mass-tumors in Breast mammography images .

**FUTURE ENHANCEMENT:**

This work can further be enhanced by detecting the stages of cancer for a patient in order to provide earlier treatment and improved health care.

Bottom of Form**REFERENCES**

1. I. Hirra et al., "Breast Cancer Classification From Histopathological Images Using Patch-Based Deep Learning Modeling," in IEEE Access, vol. 9, pp. 24273-24287, 2021, doi: 10.1109/ACCESS.2021.3056516.
2. R. K. Bhogal, P. D. Suchit and C. Naresh, "Review: Breast Cancer Detection Using Deep Learning," 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2021, pp. 847-854, doi: 10.1109/ICOEI51242.2021.9452835.
3. N. Khuriwal and N. Mishra, "Breast Cancer Detection From Histopathological Images Using Deep Learning," 2018 3rd International Conference and Workshops on Recent Advances and Innovations in Engineering (ICRAIE), Jaipur, India, 2018, pp. 1-4, doi: 10.1109/ICRAIE.2018.8710426.
4. F. Shahidi, S. Mohd Daud, H. Abas, N. A. Ahmad and N. Maarop, "Breast Cancer Classification Using Deep Learning Approaches and Histopathology Image: A Comparison Study," in IEEE Access, vol. 8, pp. 187531-187552, 2020, doi: 10.1109/ACCESS.2020.3029881.
5. N. S. Ismail and C. Sovuthy, "Breast Cancer Detection Based on Deep Learning Technique," 2019 International UNIMAS STEM 12th Engineering Conference (EnCon), Kuching, Malaysia, 2019, pp. 89-92, doi: 10.1109/EnCon.2019.8861256.
6. D. M. Vo, N. Q. Nguyen, and S. W. Lee, “Classification of breast cancer histology images using incremental boosting convolution networks,” Inf. Sci. (Ny)., vol. 482, pp. 123–138, 2019.
7. H. Chougrad, H. Zouaki, and O. Alheyane, “Deep Convolutional Neural Networks for breast cancer screening,” Comput. Methods Programs Biomed., vol.157,pp.19–30,201.
8. R. K and M. S. K, "Breast Cancer Prediction by Leveraging Machine Learning and Deep learning Techniques with Different Imaging Modalities," 2022 IEEE 7th International conference for Convergence in Technology (I2CT), Mumbai, India, 2022, pp. 1-6, doi: 10.1109/I2CT54291.2022.9824749.
9. Z. Wang et al., "Breast Cancer Detection Using Extreme Learning Machine Based on Feature Fusion With CNN Deep Features," in IEEE Access, vol. 7, pp. 105146-105158, 2019, doi: 10.1109/ACCESS.2019.2892795.
10. Y. Wang, L. Zhang, X. Shu, Y. Feng, Z. Yi and Q. Lv, "Feature-Sensitive Deep Convolutional Neural Network for Multi-Instance Breast Cancer Detection," in IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 19, no. 4, pp. 2241-2251, 1 July-Aug. 2022, doi: 10.1109/TCBB.2021.3060183.
11. A. R. Beeravolu, S. Azam, M. Jonkman, B. Shanmugam, K. Kannoorpatti and A. Anwar, "Preprocessing of Breast Cancer Images to Create Datasets for Deep-CNN," in IEEE Access, vol. 9, pp. 33438-33463, 2021, doi: 10.1109/ACCESS.2021.3058773.
12. T. Reimer and S. Pistorius, "The Diagnostic Performance of Machine Learning in Breast Microwave Sensing on an Experimental Dataset," in *IEEE Journal of Electromagnetics, RF and Microwaves in Medicine and Biology*, vol. 6, no. 1, pp. 139-145, March 2022, doi: 10.1109/JERM.2021.3076100.