**MACHINE LEARNING (17CS3166)**

**PROJECT BASED CLASS REPORT**

**On**

**IMPLEMENTATIONOF BREAST CANCER USING ALEXNETS AND GOOGLENETS**

**Submitted in partial fulfilment of the**

**Requirements for the award of the Degree of**

**Bachelor of Technology**

**In**

**Computer Science & Engineering**

**By**

|  |  |
| --- | --- |
| **NAME** | **ID-NO** |
| P. Meghana | 170030949 |
| CH. Sai Shankar | 170030239 |
| Baby Mounica Andey | 170030092 |



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**(DST-FIST Sponsored Department)**

**K L E F**

Green Fields, Vaddeswaram, Guntur District-522 502

**K L E F**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**(DST-FIST Sponsored Department)**



***CERTIFICATE***

This is to certify that this project based Class report entitled **“ IMPLEMENTATION OF BREAST CANCER USING ALEXNETS AND GOOGLENETS”** is a bonafide work done by P.Meghana(170030949), CH. Sai Shankar(170030239),Baby Mounica Andey(170030092) partial fulfilment of the requirement for the award of degree in **BACHELOR OF TECHNOLOGY** in **Computer Science and Engineering** during the academic year 2018-2019.

**Dr. K. Swarna Mr. V. Hari Kiran**

**Faculty in Charge Head of the Department**

**K L E F**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**(DST-FIST Sponsored Department)**



***DECLARATION***

We hereby declare that this project-based lab report entitled **“IMPLEMENTATION OF BREAST CANCER USING ALEXNETS AND GOOGLENETS”** has been prepared by us in partial fulfilment of the requirement for the award of degree “**BACHELOR OF TECHNOLOGY in COMPUTERSCIENCE AND ENGINEERING**” during the academic year 2019-2020.

We also declare that this project-based class report is of our own effort and it has not been submitted to any other university for the award of any degree.

**Date: 1-11-2019**

**Place: Vaddeswaram**

**ACKNOWLEDGEMENTS**

Our sincere thanks to **Dr.K.Swarna** for rendering outstanding support throughout the project for the successful completion of the work

We express our gratitude to **Mr.V.Harikiran,** Head of the Department for Computer Science and Engineering for providing us with adequate facilities, ways and means by which we are able to complete this term paper work.

We would like to place on record the deep sense of gratitude to the honourable Vice Chancellor, K L University for providing the necessary facilities to carry the concluded term paper work.

Last but not the least, we thank all Teaching and Non-Teaching Staff of our department and especially my classmates and my friends for their support in the completion of our term paper work.

**Submitted By:**

|  |  |
| --- | --- |
| **NAME** | **ID-NO** |
| P.Meghana | 170030949 |
| CH.Sai Shankar | 170030239 |
| Baby Mounica Andey | 170030092 |

TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **S.NO** | **NAME OF THE CONTENT** | **PAGE NO** |
| 1. | ABSTRACT | 6 |
| 2. | INTRODUCTION | 7-8 |
| 3. | LITERATURE SURVEY | 9 |
| 4. | METHODOLOGY | 10 |
| 5. | SOURCE CODE | 11-17 |
| 6. | OUTPUTS | 18-22 |
| 7. | CONCLUSION | 23 |
| 8. | FUTURE SCOPE | 24 |
| 9. | REFERENCES | 25 |

**ABSTRACT**

To classify microscopic images, we used convolutional neural networks (CNN) of two types: GoogLeNet and AlexNet. Due to the very large size of images of cytological specimen (on average 200000 × 100000 pixels), they were divided into smaller patches of size 256 × 256 pixels. Breast cancer classification usually is based on morphometric features of nuclei. Therefore, training and validation patches were selected using Support Vector Machine (SVM) so that suitable amount of cell material was depicted. Neural classifiers were tuned using GPU accelerated implementation of gradient descent algorithm. Training error was defined as a cross-entropy classification loss. Classification accuracy was defined as the percentage ratio of successfully classified validation patches to the total number of validation patches. The best accuracy rate of 83% was obtained by GoogLeNet model. We observed that more misclassified patches belong to malignant cases.

We evaluate three network architectures: a shallow CNN (the baseline model), an AlexNet [16] and a GoogLeNet [22]. For both the AlexNet and GoogLeNet, we use the same base architecture as the original works but replace the last fully-connected (FC) layer to output 2 classes. We also remove the two auxiliary classifiers from the GoogLeNet which we found impaired our training in practice. The baseline model’s architecture is inspired by the early layers of AlexNet [16]. We additionally use batch normalization [13]. The network takes a 224 × 224 × 3 image as input. It consists of 3 convolutional blocks composed of 3 × 3 Convolutions - Batch Norm - ReLU - Max Pooling, with respectively 32, 32 and 64 filters each, followed by 3 FC layers of size 128, 64 and 2. The final layer is a soft-max layer for binary classification. We use ReLU activation functions, Xavier [12] weight initialization, and the Adam [15] update rule with a base learning rate of 10−3 and batch size 64.

**INTRODUCTION**

Breast cancer is one of the frequent diagnosis diseases among women. It can be detected by clinical breast examination, yet the detection rate endures to be very low. Additionally, the abnormal areas that cannot be felt can be quite challenging to check using traditional techniques but can be easily seen on a conventional mammogram or with ultrasound. Mammography is currently the best method for detecting breast cancer at its early stage. The problem with mammography images are they are complex. Thus, image processing and features extraction techniques are used to assist radiologist for detecting tumour. Features extracted from suspicious regions in mammography images can help doctors to discover the existence of the tumour at real time thus speeding up treatment process. Detecting breast cancer can be quite a challenging job. Specially, as cancer is not a single disease but is a collection of multiple diseases. Thus, every cancer is different from every other cancer that exist. Also, the same drug may have different reaction on similar type of cancer. Thus, cancer vary from person to person. Depending on only one technique or one algorithm to detect breast cancer may not provide us with the best possible result. As one cancer differ from another, similarly every breast appears differently from another. The mammography image can also be compromised if the patient has undergone some breast surgery.

It can be implemented by using alexnets and as well as google nets and Googlenet consists of 4 components:

**stem**

stem layer is the sequential chain of convolution, pooling, and local response normalization operations, similar to AlexNet. Stem layer is referred in later papers. The authors of later papers cite technical issues with it when compared with pure network of inception modules, so it may be disappear.

**inception modules(which its inspiration comes from the meme 'we need to go deeper')**

inception module is the basic building block of GoogLeNet. It is a set of convolutions and poolings at different scales, each done in parallel, then concatenated together with depth. Along the way, 1x1 convolutions(3x3 reduce, 5x5 reduce) are used to reduce the dimensionality of inputs to convolutions with larger filter sizes(3x3, 5x5). This approach results in a high performing model with drastically fewer parameters. As a result, GoogLeNet has a 12 times lower training parameters than AlexNet.

**auxiliary classifiers**

Given the relatively large depth of the network, the ability to back-propagate gradient through all layer is a concern. On the other hand, the researchers thought that the features produced by layers in the middle of the network should be very discriminative. By adding auxiliary classifiers connected to these intermediate layers(inception modules), encouraging discrimination between classes is expected, increasing backpropagation signal and eventually solving "vanishing gradient" problem. During training, their loss gets added to the total loss of the network with a discount weight(the losses of the auxiliary classifiers were weighted by 0.3)

**output classifier for final classification**

The output classifier is the component which classifies the image. It performs an average pooling operation followed by a softmax activation on a fully connected layer

Breast Cancer has been a big topic in research field for the last two decades. It has been well funded medical research topic across the globe. Many people have been cured of it, due to early detection. Still the progress in diagnosis and treatment for it remains expensive and time consuming. Automated detection of mass still remains a difficult task, this might be due to the fact that every cancer is different like it’s host and each requires customized medication to be cured. So, a lot of work is still left to be done. Some of the reasons for the challenges in automated detection as follows.

First to begin with, the object of interest can be to an extraordinary degree small, inciting to potential miss-identification. Second, unique sizes, different shapes, and variable appropriations of microcalcifications show up in mammograms, therefore, sample matching seems to be impossible. Third, the Region of Interest (ROI) might be of low contrast. The refinement between suspicious reaches and their enveloping tissues can be thin. Fourth, the thick tissues as well as skin thickening, particularly in young women, cause suspicious territories to be practically undetectable. Finally, dense tissues may easily be confused as calcifications, resulting in high false-positive cases.

**LITERATURE SURVEY**

Detecting macrocalcification in dense breast tissue can be a difficult task as both tends to depict white pixel on the mammogram. The number of false positive cases on dense breast tissue are higher. Indicators of cancer symptomare generally, masses and microcalcifications. Detecting masses are more challenging task than detection of micro-calcifications. As their size and shape varies in large variation and they often exhibit poor image contrast.  The utilization of grouping frameworks in classification and pattern recognition system, in medical diagnosis, specially cancer diagnosis are growing rapidly. Evaluation and decision making based on machine learning for medical diagnosis is a key factor. Intelligent classification algorithm may help doctor in identifying symptoms that may not be possible through traditional approaches .

For the AlexNet model, we initialize the convolutional layers with pre-trained weights and a smaller learning rate multiplier of 0.1, and randomly initialize the 3 FC layers. For the GoogLeNet, we use the same weight initialization scheme. We use a learning rate multiplier of 0.1 for the layers before the Inception\_5a module, 1 for the Inception\_5a and Inception\_5b modules, and 10 for the last FC layer for more aggressive learning. We train the AlexNet with Adam, base learning rate 10−3 , and dropout rate 0.5. We train the GoogLeNet with Vanilla SGD, base learning rate 10−2 , and dropout rate 0.2

Any Image processing and analysis applications would require a unique function for alignment of feature for classification and segmentation. Mainly texture features and statistical features are of more suitable in pattern recognition area to find this alignment.

Screening Mammography is the easiest and affordable way to diagnosis for breast cancer. The mammography image is checked through several techniques like finding edges, smoothing border, finding structures & shapes among matrixes. Finally finding the size distribution of tissues in an Image without explicitly segmenting each object .

Digital mammography is the standard procedure for breast cancer diagnosis, various classification problem is applied on the digital mammography image. Various features are extracted as per standard procedure for breast cancer diagnosis. These features are calculated from the sensitive part of the breast to avoid any unwanted features to affect the classification problem. Area of tumour is calculated by the Maximum Likelihood Estimation (MLE). All the features extraction techniques are applied on the stored database image .

The paper mainly studies the multiple  image processing algorithms which can be extensively used for finding cancerous cells. The techniques in computer aided mammography includes image pre-processing, and build a sequential model. Further developments are required to extract more features to find pattern in tumour to have a better understanding on them. Texture analysis method can be used to classify between benign and malignant masses by means to identify the micro-calcification in the mammography.

**METHODOLOGY**

In deep learning a **convolutional neural network** (**CNN**, or **ConvNet**) is a class of deep neural networks, most commonly applied to analyzing visual imagery. They are also known as **shift invariant** or **space invariant artificial neural networks** (**SIANN**), based on their shared-weights architecture and translation invariancecharacteristics.[[1]](https://en.wikipedia.org/wiki/Convolutional_neural_network#cite_note-:0-1)[[2]](https://en.wikipedia.org/wiki/Convolutional_neural_network#cite_note-:1-2) They have applications in image and video recognition, recommender systems, image classification, medical image analysis, and natural language processing

Early breast cancer is difficult to detect because it appears similar to normal breast tissue. The difference between cancer and normal tissue needs to be amplified, and for this the use of lower energy X-rays is required. This can be achieved only with specially designed radiographic equipment dedicated to mammography.

**SOURCE CODE**

import numpy as np

import pandas as pd

import tensorflow as tf

import matplotlib.pylab as plt

import seaborn as sns

from glob import glob

import fnmatch

import cv2

import sklearn

from sklearn import model\_selection

from sklearn.model\_selection import train\_test\_split, kfold, cross\_val\_score, stratifiedkfold, learning\_curve, gridsearchcv

from sklearn.metrics import confusion\_matrix, make\_scorer, accuracy\_score

import keras

from keras import backend as k

from keras.callbacks import callback, earlystopping, reducelronplateau, modelcheckpoint

from keras.preprocessing.image import imagedatagenerator

from keras.utils import plot\_model

from keras.utils.np\_utils import to\_categorical

from keras.utils.vis\_utils import model\_to\_dot

from keras.models import sequential, model\_from\_json

from keras.optimizers import sgd, rmsprop, adam, adagrad, adadelta

from keras.layers import dense, dropout, embedding, spatialdropout1d, activation, flatten, batchnormalization, conv2d, maxpool2d, maxpooling2d

%matplotlib inline

import random

# source: https://www.kaggle.com/paultimothymooney/predicting-idc-in-breast-cancer-histology-images/notebook

imagepatches = glob('idc\_regular\_ps50\_idx5/\*\*/\*.png', recursive=true) # search pathname/folder to find .png files recursively then parse files

for filename in imagepatches[0:10]:

print(filename)

image\_name = "idc\_regular\_ps50\_idx5/9135/1/9135\_idx5\_x1701\_y1851\_class1.png" #image to be used as query

def plotimage(image\_location):

image = cv2.imread(image\_name)

image = cv2.resize(image, (50,50))

plt.imshow(cv2.cvtcolor(image, cv2.color\_bgr2rgb)); plt.axis('off')

return

plotimage(image\_name)

patternzero = '\*class0.png'

patternone = '\*class1.png'

classzero = fnmatch.filter(imagepatches, patternzero) #filename pattern matching returns subset of list of names

classone = fnmatch.filter(imagepatches, patternone)

print("idc(-)\n\n",classzero[0:5],'\n')

print("idc(+)\n\n",classone[0:5])

def proc\_images(lowerindex,upperindex):

"""

returns two arrays:

x is an array of resized images

y is an array of labels

"""

x = []

y = []

width = 50

height = 50

for img in imagepatches[lowerindex:upperindex]:

full\_size\_image = cv2.imread(img)

x.append(cv2.resize(full\_size\_image, (width,height), interpolation=cv2.inter\_cubic))

if img in classzero:

y.append(0)

elif img in classone:

y.append(1)

else:

return

return x,y

%time

x,y = proc\_images(0,100000)

df = pd.dataframe()

df["images"]=x

df["labels"]=y

x2=df["images"]

y2=df["labels"]

x2=np.array(x2)

imgs0=[]

imgs1=[]

imgs0 = x2[y2==0] # (0 = no idc, 1 = idc)

imgs1 = x2[y2==1]

def describedata(a,b):

print('total number of images: {}'.format(len(a)))

print('number of idc(-) images: {}'.format(np.sum(b==0)))

print('number of idc(+) images: {}'.format(np.sum(b==1)))

print('percentage of positive images: {:.2f}%'.format(100\*np.mean(b)))

print('image shape (width, height, channels): {}'.format(a[0].shape))

describedata(x2,y2)

dict\_characters = {0: 'idc(-)', 1: 'idc(+)'}

print(df.head(10))

print("")

print(dict\_characters)

def plotone(a,b):

"""

plot one numpy array

"""

plt.subplot(1,2,1)

plt.title('idc (-)')

plt.imshow(a[100])

plt.subplot(1,2,2)

plt.title('idc (+)')

plt.imshow(b[100])

plotone(imgs0, imgs1)

x=np.array(x)

x=x/255.0

x\_train, y\_train, = (x, y)

from keras.models import sequential

from keras.layers import dense

from keras.layers import conv2d

from keras.layers import flatten, activation

import keras.backend as k

model=sequential()

model.add(flatten(input\_shape=(50,50,3)))

model.add(dense(128, activation='relu'))

model.add(dense(10, activation='softmax'))

#source: (fashion mnist) https://www.tensorflow.org/tutorials/keras/basic\_classification

model.compile(loss='sparse\_categorical\_crossentropy',

optimizer=('adam'),

metrics=['accuracy'])

history = model.fit(x\_train, y\_train, validation\_split=0.20, epochs=20)

print(history.history.keys())

model.summary()

# source: https://keras.io/visualization/

# summarize history for accuracy

import matplotlib.pyplot as plt

plt.plot(history.history['acc'])

plt.plot(history.history['val\_acc'])

plt.title('model accuracy')

plt.ylabel('accuracy')

plt.xlabel('epoch')

plt.legend(['train', 'test'], loc='upper left')

plt.show()

# summarize history for loss

plt.plot(history.history['loss'])

plt.plot(history.history['val\_loss'])

plt.title('model loss')

plt.ylabel('loss')

plt.xlabel('epoch')

plt.legend(['train', 'test'], loc='upper left')

plt.show()

import matplotlib.pyplot as plt

plt.plot(history.history['acc'])

plt.plot(history.history['val\_acc'])

plt.title('model accuracy')

plt.ylabel('accuracy')

plt.xlabel('epoch')

plt.legend(['train', 'test'], loc='upper left')

plt.show()

# Summarize history for loss

plt.plot(history.history['loss'])

plt.plot(history.history['val\_loss'])

plt.title('model loss')

plt.ylabel('loss')

plt.xlabel('epoch')

plt.legend(['train', 'test'], loc='upper left')

plt.show()

#Source: https://www.pyimagesearch.com/2018/09/10/keras-tutorial-how-to-get-started-with-keras-deep-learning-and-python/?fbclid=lwAR3WMQfZB513JosNLudYcv6DaaVwEv5pD6rQEMB4l\_fDRxlwWfNG8gObmJU

# evaluate the network

print("[INFO] evaluating network...")

score = model.evaluate(X\_test, Y\_test, batch\_size=100)

# plot the training loss and accuracy

N = np.arange(0, EPOCHS)

plt.style.use("ggplot")

plt.figure()

plt.plot(N, H.history["loss"], label="train\_loss")

plt.plot(N, H.history["val\_loss"], label="val\_loss")

plt.plot(N, H.history["acc"], label="train\_acc")

plt.plot(N, H.history["val\_acc"], label="val\_acc")

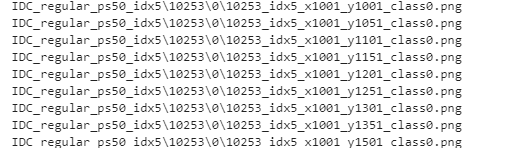
plt.title("Training Loss and Accuracy")

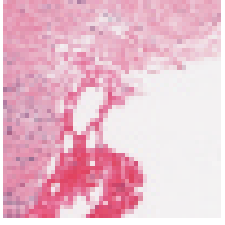
plt.xlabel("Epoch #")

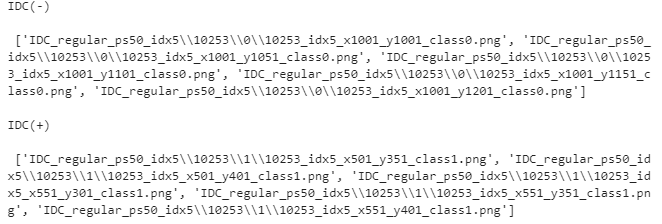
plt.ylabel("Loss/Accuracy")

plt.legend()

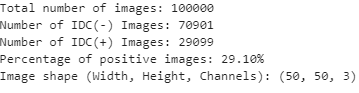
OUTPUT

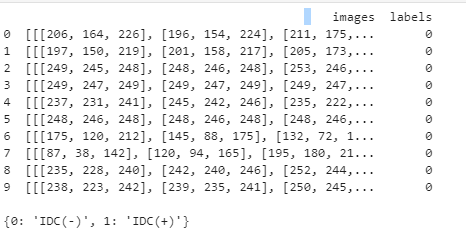
****

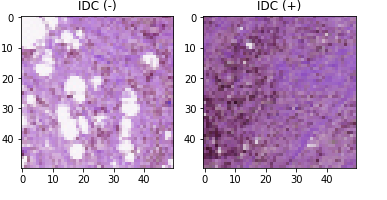
****

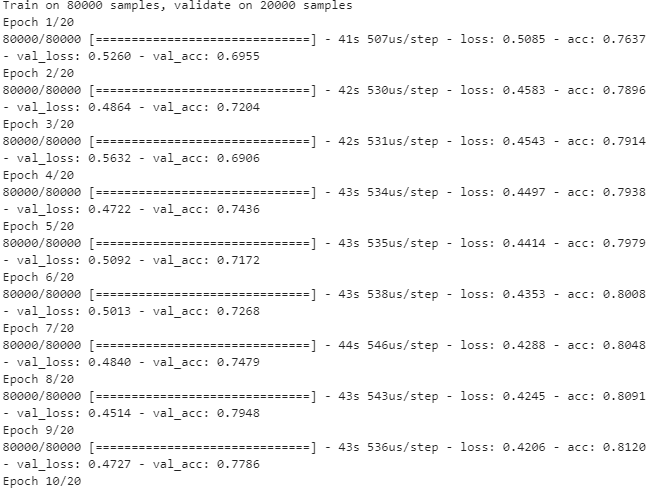
****

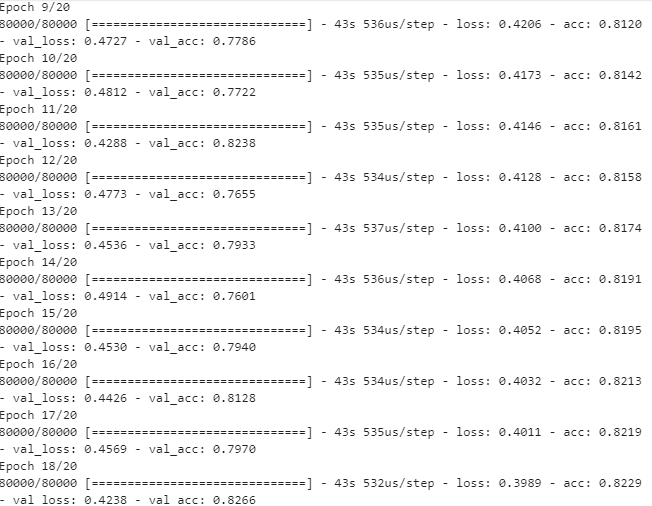
****

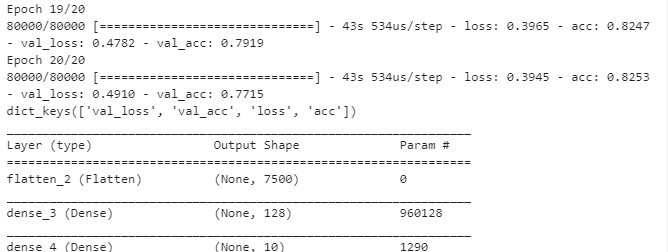
****

****

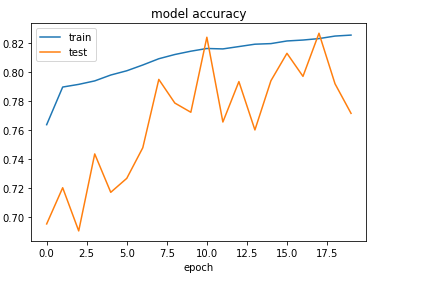
****

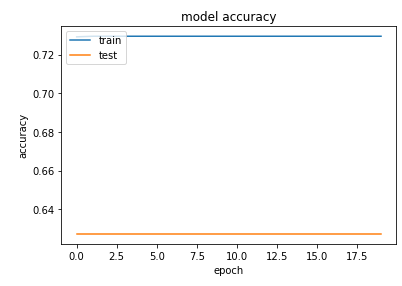
****

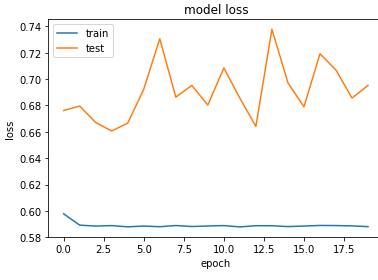
****

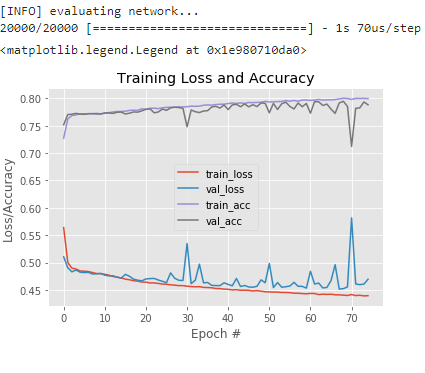
****

****

****

****

****

****

**CONCLUSION**

Breast cancer is one of the major causes of death among women with 1 woman affected by breast cancer out of  8 women.  In the diagnosis process, due to the wide range of features associated to breast abnormalities some abnormalities may be missed or misinterpreted. There is also a number of false positive findings and therefore a lot of unnecessary biopsies may be requied. Computer-aided detection and diagnosis algorithms have been developed to help radiologists give an accurate diagnosis and to reduce the number of false positives. In this study, typical steps in image processing algorithms have been extensively studied. The techniques in the field of computer aided mammography include image pre-processing, image segmentation techniques, feature extraction, feature selection, classification techniques and features for mammograms. Texture feature are obtained to distinguish between normal cell and cancerous cell. Cancer being one of the oldest disease and lot of research has been carried out in this field.  Cancer is not a single disease rather a collection of multiple diseases thus a single medicine to cure cancer is not possible. The key input to cure a cancer is customization of medication based on the type of cancer and it can be cured if found it early.

**FUTURE SCOPE**

Breast Cancer has become one of the most common diseases among women which leads them to death. One woman among every eight women is suffering from breast cancer. There are also false positive findings of breast cancer reports due to many variations in images. The development of a model using machine learning for the detection of breast cancer using images can solve this problem to a larger extent. There is a lot of scope and advantages of this model in future.

**REFERENCES**

1. [https://www.researchgate.net/publication/328562202\_Flood\_Prediction\_Usin g\_Machine\_Learning\_Models\_Literature\_Review](https://www.researchgate.net/publication/328562202_Flood_Prediction_Using_Machine_Learning_Models_Literature_Review)

2. Lohani, A.K.; Goel, N.; Bhatia, K. Improving real time ﬂood forecasting using fuzzy inference system. J. Hydrol. 2014,509, 25–41.

<https://www.sciencedirect.com/science/article/pii/S0022169413008378?via%3Dihub>