Melanoma Skin Cancer Detection

A Project Report

Submitted for the course: Image Processing (CSE4019)

By

NIKITA NEGI - 16BCE2038

ISHITA GUPTA -17BCE0710

ADHARSH G - 17BCE0014

SUDERSHAN SRIDHAR – 17BCE0724

Fall Semester 2018 - 2019
Slot: TE1

Under the Guidance of:

Prof. RAJKUMAR S (SCOPE)

Curriculum for Applied Learning

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

November 2018

DECLARATION BY THE CANDIDATE

We hereby declare that the project report entitled **"Melanoma Skin Cancer Detection"** submitted by us to Vellore Institute of Technology, Vellore in partial fulfilment of the requirement for the award of the degree of **B. Tech (CSE)** is a record of J- component of project work carried out by us under the guidance of **Prof. RAJKUMAR S**. We further declare that the work reported in this project has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place : Vellore Institute of Technology, Vellore.
Date : 12-11-2018

Signature of the faculty Signature of the Candidate

TABLE OF CONTENTS

1. Introduction

1.1. Abstract:

Skin cancers are a type of cancers that arise from the skin. They are due to the development of abnormal cells that have the ability to invade or spread to other parts of the body. It occurs whenever unrepaired DNA damages to skin cells triggers mutations, or any other genetic defects, that lead the skin cells to multiply readily and form malignant tumors. Image processing is a commonly used method for skin cancer detection from the appearance of affected area on the skin. The input to the system is the skin lesion image and then by applying novel image processing techniques, it analyses it to conclude about the presence of skin cancer.

1.2. Background and problem description

Melanoma, a type of deadly skin cancer affects the region of skin which is exposed directly to UV radiation which shows a rapid death chance. In order to lower the death rate early detection methods are adapted. According to the statistical information available, it has been proved that the melanoma incidence rates showed an increase of 2% to 7% per year in between 2006-2010 and also the death rate showed an increase of 1.1% in males and 0.2% in females per year. According to the recent survey of 2014, it has been stated that the total effects of melanoma are around 76,100 and the deaths are around 9,710. In order to decrease the death rate, image processing is used for the detection of skin cancer. By using this methodology early detection of skin cancer can be achieved. It lessens the burden of Dermatologists. A solution to this problem is using image processing to detect skin cancer.

In recent days, skin cancer is commonly seen as one of the most dangerous forms of the cancers identified in Humans. Exposure to ultraviolet radiation from the Sun causes more than 90% of the cases. This exposure increases the risk of all three major skin cancer types. Partly due to a thinner ozone layer, exposure has increased. Tanning beds become another common source of ultraviolet radiation. For basal cell and melanoma cancers ex. Decreasing exposure to ultraviolet radiation and the use of sunscreen appear to be effective methods for the prevention of melanoma and squamous cell skin cancer. It is not clear whether sunscreen affects the risk of basal cell cancer. Skin cancer is the most common form of cancer, globally accounting for at least 40% of cases. The most common type is nonmelanoma skin cancer, which occurs in at least 2-3 million people per year. This is a rough estimate, however, as good statistics are not kept. Of nonmelanoma skin cancers, about 80% are basal-cell cancers and 20% squamous-cell skin cancers. Basal-cell and squamous-cell skin cancers rarely result in death. In the United States they were the cause of less than 0.1% of all cancer deaths. Globally in 2012 melanoma occurred in 232,000 people, and resulted in 55,000 deaths. White people in Australia, New Zealand and South Africa have the highest rates of melanoma in the world.The three main types of skin cancer have become more common in the last 20 to 40 years, especially in those areas which are mostly Caucasian.Skin cancer is classified into various types such as Melanoma, Basal and Squamous Cell Carcinoma out of which Melanoma

is the most unpredictable and the most common form of cancer. Melanoma could be a notably deadly variety of skin cancer, and though it justifies solely 4% of all types of skin cancers, it is responsible for 75% of all skin cancer deaths. Image processing is one of the most widely used methods for skin cancer detection. 'Dermoscopy' could be a non-invasive examination technique that supports the cause of incident light beam and oil immersion technique for the visual investigation of surface structures of the skin. The detection of melanoma using dermoscopy is higher than individual observation based detection, but its diagnostic accuracy depends on the factor of training the dermatologist. The diagnosis of melanoma is not very clear and easy to identify, especially in the early stage. Thus, automatic diagnosis tool is more effective and essential. Other than 'dermoscopy', a computerized melanoma detection system using Artificial Neural Network has been adapted which is more efficient than the conventional one for identification and classification. The image is then modified such that unnecessary parts of the image can be removed from the image which is then processed to check if it is cancerous .The preprocessing part includes conversion of image to grey scale and tracing the boundary of the cancer .The Lesion Image analysis tools checks for the various Melanoma parameters Like Asymmetry, Border, Color, Diameter, (ABCD rule) etc. by texture, size and shape analysis for image segmentation and feature stages. The extracted feature parameters are used to classify the image as Normal skin and Melanoma cancer lesion. Artificial Neural Network (ANN) is one of the important branches of Artificial Intelligence, which has been accepted as a brand-new technology in computer science for image processing. It has been used to analyze Melanoma parameters Like Asymmetry, Border, Color, Diameter, etc. which are calculated using MATLAB from skin cancer images intending to developing diagnostic algorithms that might improve triage practices in the emergency department. The network is well trained with upwards of 80% accuracy.

2. Planning

2.1. Software/Data Requirements

1.MATLAB - Implementation part
2. Sample dataset of benign and malign lesion image (https://iisic-archive.com)

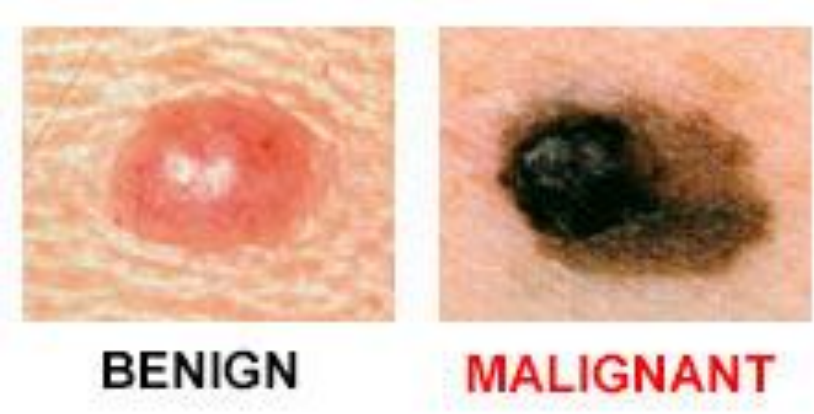


Fig 1. Sample of the two datasets

2.2. HARDWARE REQUIREMENTS:

1. Processor (CPU) with 2 gigahertz (GHz) frequency or above
2. A minimum of 2 GB of RAM
3. Monitor Resolution 1024 X 768 or higher
4. A minimum of 2 GB of available space on the hard disk.

3. Literature Summary

After carrying a survey on around twenty-five research papers, we can draw the inference that There is still a lot of scope for research in the field of image processing for skin cancer detection and it can be furthermore used to reduce the number of deaths caused by melanoma and other kinds of cancer. Image-based computer aided diagnosis systems have a much significant potential for screening and early detection of malignant melanoma. We reviewed the state of the art in these systems and then examine the current practices, problems, and prospects of image acquisition, pre-processing, segmentation, feature extraction and selection, and classification of dermoscopic images.

The incidence of skin cancer incidents has been drastically elevating day-to-day. Skin cancer in early stage could be cured easily by simple procedures or techniques but advanced skin cancer cannot be treated effectively by any medications. So, there is a need to detect and treat disease at early stage. Overall, 4% of the cancer cases are melanoma. UV-A and B are mainly responsible for skin cancer. Outdoor workers are generally more prone to skin cancer because they get easily exposed to skin cancers. So, precautionary measures like application of sunscreen lotions need to be done. It can be treated at initial stages, as the duration is extended, the chances for treating skin cancer gets hastened. New molecular therapeutic approaches for skin cancer include several medications like cryosurgery, immunomodulation with imiquimod, 5-FU, photodynamic therapy etc.

Tomographic imaging of any soft tissue like skin, has a potential role in cancer detection. The penetration of infrared wavelengths makes a confocal approach based on laser feedback interferometry feasible. Experimental results were in agreement with numerical simulations and structural changes were evident which would permit discrimination of healthy tissue and tumor. Furthermore, cancer type discrimination was also able to be visualized using this imaging technique.

4. Methodology

The following chart illustrates the flow and various methods used in the implementation of the project. The step by step execution of the main module and steps are illustrated to provide a visual representation of the execution.

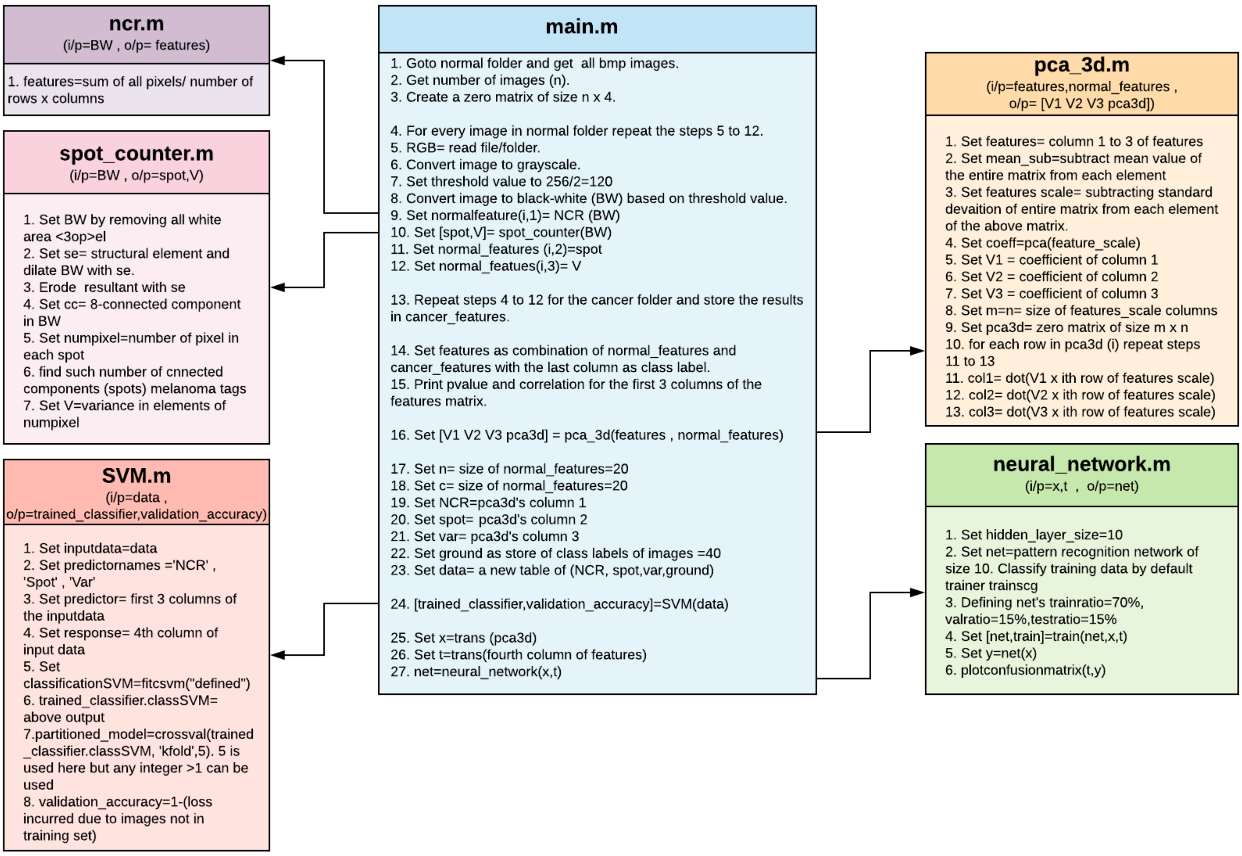
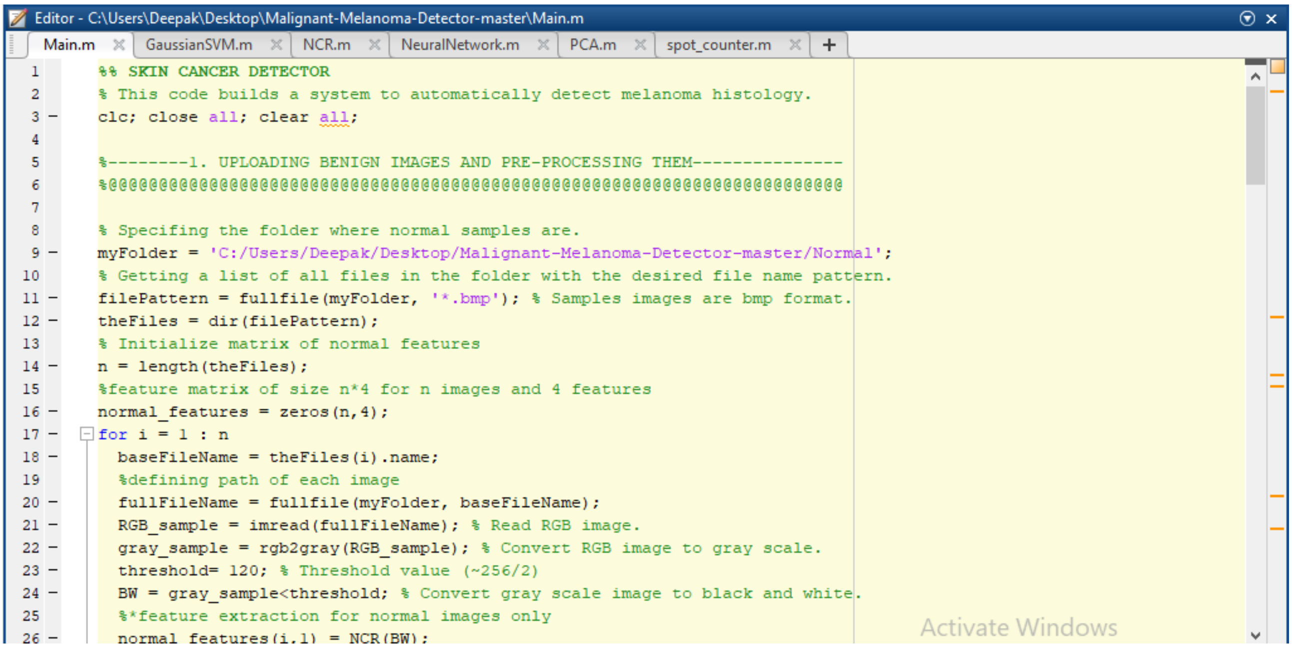


Fig 2. Implementation strategy chart

5. System Implementation

5.1. Code




```
%% SKIN CANCER DETECTOR
% This code builds a system to automatically detect melanoma histology.
clc; close all; clear all;

%--------1. UPLOADING BENIGN IMAGES AND PRE-PROCESSING THEM---------------
%@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

% Specifing the folder where normal samples are.
myFolder = 'C:/Users/Deepak/Desktop/Malignant-Melanoma-Detector-master/Normal';
% Getting a list of all files in the folder with the desired file name pattern.
filePattern = fullfile(myFolder, '*.bmp'); % Samples images are bmp format.
theFiles = dir(filePattern);
% Initialize matrix of normal features
n = length(theFiles);
%feature matrix of size n*4 for n images and 4 features
normal_features = zeros(n,4);
for i = 1 : n
  baseFileName = theFiles(i).name;
  %defining path of each image
  fullFileName = fullfile(myFolder, baseFileName);
  RGB_sample = imread(fullFileName); % Read RGB image.
  gray_sample = rgb2gray(RGB_sample); % Convert RGB image to gray scale.
  threshold= 120; % Threshold value (~256/2)
  BW = gray_sample<threshold; % Convert gray scale image to black and white.
  %*feature extraction for normal images only
  normal_features(i,1) = NCR(BW);
```

```
    [spot, V] = spot_counter(BW);
    normal_features(i,2) = spot;
    normal_features(i,3) = V;
end

%--------2. UPLOADING MALIGNANT IMAGES AND PRE-PROCESSING THEM-------------
%@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

% Specify the folder where cancer samples are.
myFolder = 'C:/Users/Deepak/Desktop/Malignant-Melanoma-Detector-master/Cancer';
% Get a list of all files in the folder with the desired file name pattern.
filePattern = fullfile(myFolder, '*.bmp'); % Samples images are bmp format.
theFiles = dir(filePattern);
% Initialize matrix of normal features
n = length(theFiles);
%feature matrix of size n*4 for n images and 4 features
cancer_features = zeros(n,4);
cancer_features(:,4) = ones(n,end);
for i = 1 : n
    baseFileName = theFiles(i).name;
    %defining path of each image
    fullFileName = fullfile(myFolder, baseFileName);
    RGB_sample = imread(fullFileName); % Read RGB image.
    gray_sample = rgb2gray(RGB_sample); % Convert RGB image to gray scale.
    BW = gray_sample<threshold; % Convert gray scale image to black and white.
    %*feature extraction for malignant images only
    cancer_features(i,1) = NCR(BW);
    [spot, V] = spot_counter(BW);
    cancer_features(i,2) = spot;
    cancer_features(i,3) = V;
end

%--------3. VISUALIZE THE EXTRACTED FEATURES-----------------------------
%@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

% Asses correlation between normal and cancer features
features = [normal_features; cancer_features];
[linear_corr,pval] = corr(features(:,1:3))

%--------4. PERFORMING PRINCIPLE COMPONENT ANALYSIS----------------------
%@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

[V1,V2,V3,pca_3d] = PCA(features,normal_features);

%--------5. PERFORMING TRAININNG OF THE CLASSIFIER-----------------------
%@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

n = size(normal_features,1);
c = size(cancer_features,1);
NCR = pca_3d(:,1);
spot = pca_3d(:,2);
Variance = pca_3d(:,3);
%actual class of the images
Ground_truth = vertcat(repmat(['Normal'],n,1),repmat(['Cancer'],c,1));
data = table(NCR,spot,Variance,Ground_truth);
% Classification Learner
[trainedClassifier, validationAccuracy]= GaussianSVM(data)
% Neural Net Pattern Recognition
% Transpose column vector data
% The output will be raw output of neural network which is a number from 0 to 1.
%Numbers close to 0 indicate normal while numbers close to 1 indicate cancer.
x = pca_3d';
t = features(:,4)';
net = NeuralNetwork(x,t);

%--------6. PERFORMING PREPROCESSING VISUALIZATION-----------------------
%@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

I = imread('C:/Users/Deepak/Desktop/Malignant-Melanoma-Detector-master/Normal/benign1.bmp');
ei=25;
st=35;
k=ei*st;
%defining median filter
h = ones(ei,st) / k;
```

```
102     %applying filter
103 -   I1 = imfilter(I,h,'symmetric');
104 -   figure
105 -   subplot(2,2,1),imshow(I), title('Original image');
106 -   subplot(2,2,2), imshow(I1), title('Filtered Image');
107     %converting to Grayscale
108 -   IG=rgb2gray(I1);
109     %Converting to BW
110     %stretching intensity value
111 -   I11 = imadjust(IG,stretchlim(IG),[]);
112     %finding threshold grey value
113 -   level = graythresh(I11);
114 -   BWJ = im2bw(I11,level);
115 -   dim = size(BWJ);
116 -   IN=ones(dim(1),dim(2));
117 -   BW=xor(BWJ,IN);  %inverting colors
118 -   subplot(2,2,3), imshow(BW), title('Black and White');
119     %Finding of initial point
120     %average centre row for boundary detection
121 -   row = round(dim(1)/2);
122     %finding the column where lesion starts
123 -   col = min(find(BW(row,:)));
124     %Tracing
125 -   boundary = bwtraceboundary(BW,[row, col],'W');
126 -   subplot(2,2,4),imshow(I), title('Traced');
127 -   hold on;
128     %Display traced boundary in green
129 -   plot(boundary(:,2),boundary(:,1),'g','LineWidth',2);
130 -   hold off;
131     %-------------------------------------------------------------------------
132
```


Editor - C:\Users\Deepak\Desktop\Malignant-Melanoma-Detector-master\spot_counter.m

Main.m | GaussianSVM.m | NCR.m | NeuralNetwork.m | PCA.m | spot_counter.m

```
1   function [spot, V] = spot_counter(BW)
2   % This function takes a grayscale image and counts cell spot
3
4   % Image preprocessing
5 - BW = bwareaopen(BW,30); % get rid of white regions smaller than 30 pixels
6 - se = strel('disk', 2); % setting threshold
7 - BW = imdilate(BW, se); %dilating and eroding using structural element
8 - se = strel('disk', 2);
9 - BW = imerode(BW, se);
10 - cc = bwconncomp(BW); % default connectivity is 8
11 - numPixels = cellfun(@numel,cc.PixelIdxList);
12
13  % Count spot
14 - spot  = cc.NumObjects;
15  % spot Size Variance
16 - V = var(numPixels,0,2); % Returns a scalar containing the variance of the elements in each row.
17 - end
```

Editor - C:\Users\Deepak\Desktop\Malignant-Melanoma-Detector-master\PCA.m

Main.m | GaussianSVM.m | NCR.m | NeuralNetwork.m | PCA.m | spot_counter.m

```
1   function [V1,V2,V3,pca_3d] = PCA(features,normal_features)
2   % Perform Principal Component Analysis on the features
3   % Mean normalization and feature scaling
4 - features = features(:,1:3); % Discard the class column
5   %subtracting mean value of entire matrix from each element
6 - mean_substracted = (features - repmat(mean(features,1),size(features,1),1));
7 - features_scaled =mean_substracted./repmat(std(features,1),size(features,1),1);
8   % Perform PCA
9 - coeff = pca(features_scaled);
10 - V1 = coeff(:,1); V2 = coeff(:,2); V3 = coeff(:,3);
11  % Project the features data into the new coordinate space
12 - m = size(features_scaled,1);
13 - n = size(features_scaled,2);
14 - pca_3d = zeros(m,n);
15 - for i = 1:m
16 -     pca_3d(i,:) = [dot(V1,features_scaled(i,:)) dot(V2,features_scaled(i,:)) dot(V3,features_scaled(i,:))];
17 - end
18 - end
```

Editor - C:\Users\Deepak\Desktop\Malignant-Melanoma-Detector-master\NeuralNetwork.m

Main.m | GaussianSVM.m | NCR.m | NeuralNetwork.m | PCA.m | spot_counter.m

```
function [net] = NeuralNetwork(x,t)
% Solve a Pattern Recognition Problem with a Neural Network
% This script assumes these variables are defined:
%   pca_3d - input data.
%   output - target data.
% Create a Pattern Recognition Network
hiddenLayerSize = 10;
net = patternnet(hiddenLayerSize);

% Setup Division of Data for Training, Validation, Testing
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;

% Train the Network x=pca3d and t=class labels i.e. 4th col of features
[net,tr] = train(net,x,t);

% Test the Network
y = net(x);
% Plots
figure, plotconfusion(t,y)

end
```

Editor - C:\Users\Deepak\Desktop\Malignant-Melanoma-Detector-master\GaussianSVM.m

Main.m | GaussianSVM.m | NCR.m | NeuralNetwork.m | PCA.m | spot_counter.m

```
function [trainedClassifier, validationAccuracy] = GaussianSVM(trainingData)
% trainClassifier(trainingData)
%  returns a trained classifier and its accuracy.
%
%   Input:
%       trainingData: the training data of same data type as imported
%        in the app (table or matrix).
%
%   Output:
%       trainedClassifier: a struct containing the trained classifier.
%        The struct contains various fields with information about the
%        trained classifier.
%
%       trainedClassifier.predictFcn: a function to make predictions
%        on new data. It takes an input of the same form as this training
%        code (table or matrix) and returns predictions for the response.
%
%       validationAccuracy: a double containing the accuracy in
%        percent.
%
% Extract predictors and response
% This code processes the data into the right shape for training the
% classifier.
inputTable = trainingData;
predictorNames = {'NCR', 'spot', 'Variance'};
predictors = inputTable(:, predictorNames);
response = inputTable.Ground_truth;

% Train a classifier
% This code specifies all the classifier options and trains the classifier.
classificationSVM = fitcsvm(...
    predictors, ...
    response, ...
    'KernelFunction', 'gaussian', ...
    'PolynomialOrder', [], ...
    'KernelScale', 6.9, ...
    'BoxConstraint', 1, ...
    'Standardize', true, ...
    'ClassNames', ['C' 'a' 'n' 'c' 'e' 'r'; 'N' 'o' 'r' 'm' 'a' 'l']);

% Add additional fields to the result struct
trainedClassifier.ClassificationSVM = classificationSVM;

% Perform cross-validation
partitionedModel = crossval(trainedClassifier.ClassificationSVM, 'KFold', 5);

% Compute validation accuracy
validationAccuracy = 1 - kfoldLoss(partitionedModel, 'LossFun', 'ClassifError');
end
```

```
Editor - C:\Users\Deepak\Desktop\Malignant-Melanoma-Detector-master\NCR.m
Main.m | GaussianSVM.m | NCR.m | NeuralNetwork.m | PCA.m | spot_counter.m

1  function [ feature ] = NCR ( BW )
2  %feature=sum of all elments/no of row* no of col
3  feature = sum(sum(BW))/(size(BW,1)*size(BW,2));
4  end
5
6
```

5.2. Results and discussion

After image preprocessing and using the two classification models on the dataset we have got the precision and accuracy level of about 80% as demonstrated in the following test results for SVM and neural network classification in the confusion matrix. We can also see that neural networks being non-linear classifiers, give better results.

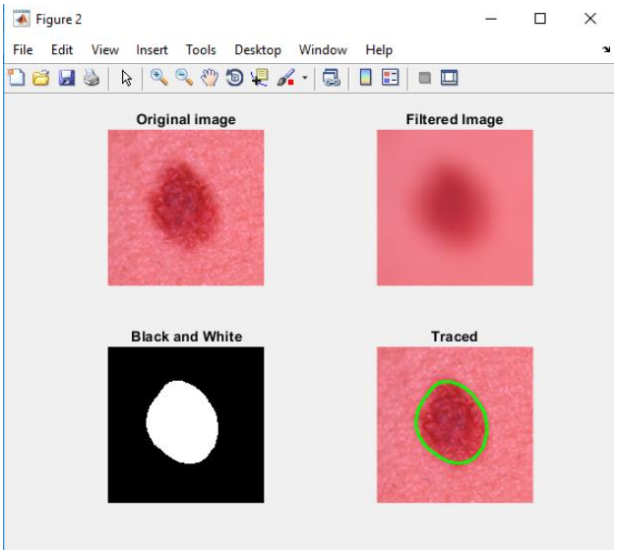


Fig 3. Preprocessing results

```
Editor - C:\Users\Deepak\Desktop\Malignant-Melanoma-Detector-master\Main.m
Command Window

linear_corr =

    1.0000   -0.2406    0.3798
   -0.2406    1.0000   -0.0644
    0.3798   -0.0644    1.0000


pval =

    1.0000    0.1347    0.0157
    0.1347    1.0000    0.6928
    0.0157    0.6928    1.0000


trainedClassifier =

  struct with fields:

    ClassificationSVM: [1×1 ClassificationSVM]


validationAccuracy =

    0.7000

fx >>
```


Fig 4. SVM accuracy results

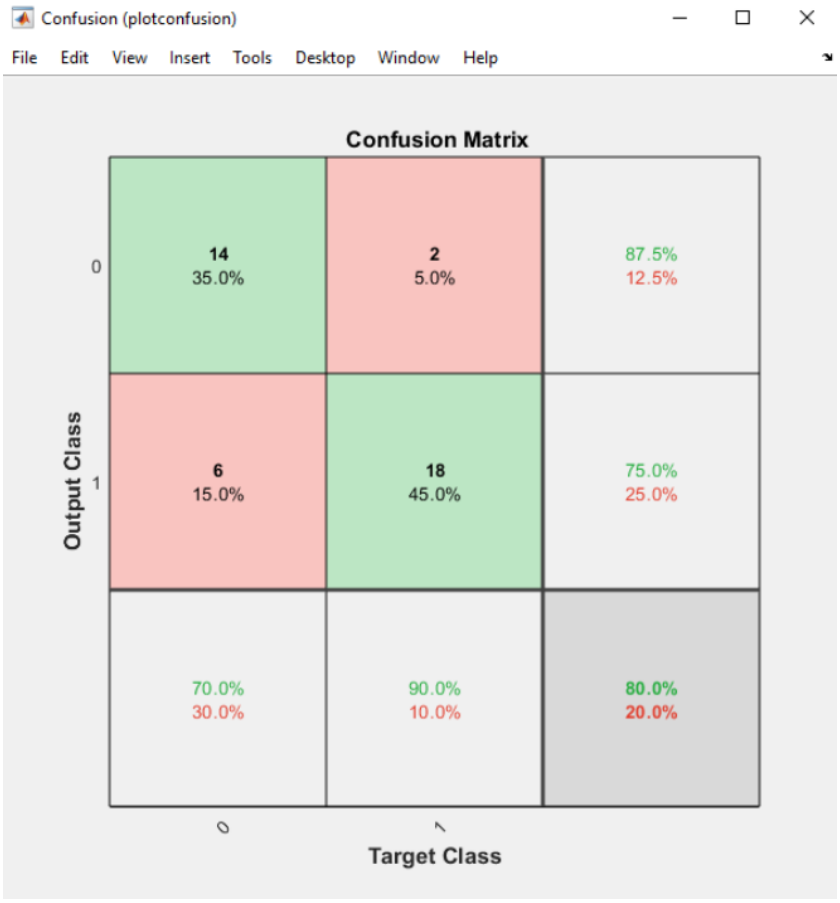


Fig 5. Confusion matrix from the neural network results

6. Conclusion and Future Work

The project done here involves training the model using sample set images that were obtained. SVM algorithm and Artificial Neural Network are used to classify the images of cancer cells. The network is well trained more than 80% accuracy, and then the unknown values are tested for the cancer classification. This classification method proves to be very efficient for the skin cancer classification compared to the conventional methods used for classification and detection.

The scope of the project gets limited with the number of samples used for the model, and even though the algorithm used here is the most optimized one that was compatible with the modules and component that was available, there is always the possibility of rise of another algorithm that could be more compatible with even a slightest configuration of the remaining modules of the project like Adaboost etc.. This detection and classification system that is used here can be expanded to many medical centers that agree to use it, by the doctors, who can instantly upload the medical cases that are presented to them, and help in the improvement of the detection process, by training the model with the images, as and when they are presented to it. This would increase the performance of the model over a period of time, with the

dynamically changing model. This needs a significant amount of research for enhancing performance, and time for sufficient training and testing of the model.

By expanding the current project as mentioned above, there could also be the establishment of a standard system, that could be constructed with the linking of separate collections of datasets at different places, to a network, so that the samples can be collected effectively and the results of the tests happening at different places can be returned quicker, even predicting the stage of the cancer.

7. References

[1] Nachbar, F. , Stolz, W. , Merkle, T. , Cognetta, A. B. , Vogt, T. , Landthaler, M. , & Plewig, G. (1994). The ABCD rule of dermatoscopy: High prospective value in the diagnosis of doubtful melanocytic skin lesions. *Journal of the American Academy of Dermatology, 30* (4), 551–559 .

[2] Nock, R. , & Nielsen, F. (2004). Statistical region merging. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 26* (11), 1452–1458 .

[3] Oliveira, R. B. , Marranghello, N. , Pereira, A. S. , & Tavares, J. M. (2016). A compu- tational approach for detecting pigmented skin lesions in macroscopic images. *Expert Systems with Applications, 61* , 53–63 .

[4] Riaz, F. , Hassan, A. , Javed, M. Y. , & Coimbra, M. T. (2014). Detecting melanoma in dermoscopy images using scale adaptive local binary patterns. In *Proc. IEEE international conference on engineering in medicine and biology society (EMBS)* (pp. 6758–6761) . Chicago, USA.

[5] Dalal, N. , & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proc. IEEE computer society conference on computer vision and pattern recogni- tion (CVPR): 1* (pp. 886–893) . San Diego, USA.

[6] Do, M. N. , & Vetterli, M. (2003).The finiteridgelet transformfor imagerepresenta- tion. *IEEE Transactions on Image Processing, 12* (1), 16–28 .

[7] Dony, R. , & Wesolkowski, S. (1999). Edge detection on color images using rgb vector angles. In *Proc. IEEECanadian conference on electrical and computer engineering (CCECE): 2* (pp. 687–692) . Edmonton, Canada.

[8] Eltayef, K. , Li, Y. , & Liu, X. (2017). Detection of pigment networks in dermoscopy images. In *Proc. IOP international conference on communication, image and signal processing (CCISP): 787* (p. 012034) . Dubai.

[9] Giotis, I. , Land, N. M. S. , Biehl, M. , Jonkman, M. F. , & Petkov, N. (2015). MED-NODE: A computer-assisted melanoma diagnosis system using non-dermoscopic images. *Expert Systems with Applications, 42* , 6578–6585 .

[10] Huang, D. , Jia, W. , & Zhang, D. (2008). Palmprint verification based on principal lines. *Pattern Recognition, 41* (4), 1316–1328 .

[11] Kiani, K. , & Sharafat, A. R. (2011). E-shaver: An improved DullRazor ®for digitally removing dark and light-colored hairs in dermoscopic images. *Computers in Bi- ology and Medicine, 41* (3), 139–145 .

[12] Korotkov, K. , & Garcia, R. (2012). Computerized analysis of pigmented skin lesions: A review. *Artificial Intelligence in Medicine, 56* (2), 69–90 .

[13] Kruk, M. , Swiderski, B. , Osowski, S. , Kurek, J. , Slowinska, M. , & Walecka, I. (2015). Melanoma recognition using extended set of descriptors and classifiers. *EURASIP Journal on Image and Video Processing, 2015* (1), 1–10 .

[14] Lee, H. , & Chen, Y. P. P. (2015). Image based computer aided diagnosis system for cancer detection. *Expert Systems with Applications, 42* , 5356–5365 .

[15] Lee, T. , Ng, V. , Gallagher, R. , Coldman, A. , & McLean, D. (1997). Dullrazor®: A soft- ware approach to hair removal from images. *Computers in Biology and Medicine, 27* (6), 533–543 .

[16] Alfed, N. , Khelifi, F. , & Bouridane, A. (2016). Improving a bag of words approach for skin cancer detection in dermoscopic images. In *Proc. IEEE international confer- ence on decision and information technologies (CoDIT)* (pp. 228–232) . Saint Ju- lian's, Malta.

[17] Alfed, N. , Khelifi, F. , Bouridane, A. , & Seker, H. (2015). Pigment network-based skin cancer detection. In *Proc. IEEE international conference on engineering in medicine and biology society (EMBS)* (pp. 7214–7217) . Milan,Italy.

[18] Argenziano, G. , Fabbrocini, G. , Carli, P. , Giorgi, V. D. , Sammarco, E. , & Delfino, M. (1998). Epiluminescence microscopy for the diagnosis of doubtful melanocytic skin lesions: Comparison of the ABCD rule of dermatoscopy and a new 7-point checklist based on pattern analysis. *Archives of Dermatology, 134* (12), 1563–1570 .

[19] Ballerini, L. , Fisher, R. B. , Aldridge, B. , & Rees, J. (2013). A color and texture based hierarchical k-nn approach to the classification of non-melanoma skin lesions. In *Color medical image analysis* (pp. 63–86). Springer.

[20] Zhao, Y. , Wang, S. , Zhang, X. , & Yao, H. (2013). Robust hashing for image authentica- tion using zernike moments and local features. *IEEE Transactions on Information Forensics and Security, 8* (1), 55–63 .