VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnanasangama, Macche, Santibastwada Road
Belagavi-590018, Karnataka

A
PROJECT REPORT
of

"AMD and Glaucoma Detection Using Retinal Funduscopic Image Processing "

Submitted in partial fulfillment of the requirement for the degree of

Bachelor of Engineering
in
Electronics & Communications Engineering
by

Prerana Varsha	**1DS14EC094**
Priyesh Raj	**1DS14EC196**
Rukmini	**1DS14EC110**
Megha Kumari	**1DS14EC074**

Under the
guidance
of

Prof. T C Manjunath
Internal Project Guide
Professor, ECE Dept., DSCE, Bengaluru

Department of Electronics & Communication Engineering
(An Autonomous College affiliated to VTU Belgaum, accredited by NBA & NAAC)
Shavige Malleshwara Hills, Kumaraswamy Layout,
Bengaluru-560078, Karnataka, India

May 2018

1.Introduction

1.1 Overview

Image processing is the processing of image by converting an image into digital conformation by applying mathematical operations in the form of signal processing for which the input is an image, a video or a series of image, like video or photograph frame, the outcome of image processing may be either an image or a set of parameters or characteristics related to the image.

Glaucoma is the eye disease in which the nerve connecting the eye to the brain is damaged, usually due to high eye pressure. Age-related macular degeneration (AMD) is a decay or breakdown of the eye macula.

Glaucoma and age-related macular degeneration (AMD) are these days two of the most incessant reasons for visual impairment and vision misfortune. In addition, high growth of the diseases will be experienced due to diabetes disease incidence increase and the lifestyle lead by populous that is ageing in the present society. Their diagnosis at earliest stage through appropriate good treatment will reduce medical treatment costs generated when they are in early stage else condition may become critical. Figure below shows the age group at risk of getting diseased either of two ailments.

Fig 1.1.1 : Glaucoma affected age group

Fig 1.1.2: AMD affected age group

Nonetheless, a screening effort need a substantial workload for good trained specialists to determine and analyze the divergent and peculiar feature patterns of each ailment that added to the in danger populace increase that makes these campaign financially infeasible. In this way, the requirement for automatic programmed screening systems is highlighted. Based on these certainty, computer-aided diagnosis software trained for discriminating through image processing between a normal healthy fundus (without pathology), Glaucomatous and age related macular degeneration patients was developed.

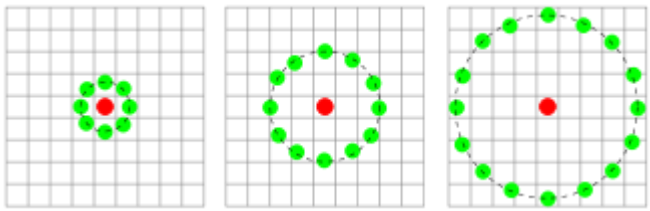


Fig 1.1.3 : Neighbourhood example to define a texture and calculate LBP.

Local Binary Pattern (LBP) is a simple straightforward yet exceptionally efficient texture operator that works by labelling the pixels of an image by thresholding the circular neighbourhood vicinity of every pixel and returns the outcome as the binary number. Due to the discerning power and computing simplicity of the LBP texture operator, it has eventually become a prominent approach in distinct applications. It is identified as the unifying approach to a conventionally divergent statistical and anatomical texture analysis models where the most decisive property of an LBP operator in existing real-world applications is from optimized robustness to the gray-scale monotonic modulation caused, an example is, by illumination variations another important property is its computing simplicity that makes it probable for analyzing images in the challenging real-time settings. [11]

The purpose of proposed work is to distinguish between normal fundus image, age- related macular degeneration (AMD) and Glaucoma images at the same time and avoiding any foregoing previous segmentation stage of retinal lesions. The retinal background textural patterns are obtained by direct means of analyzing local binary patterns, and only this knowledge or information data is used for differentiating between healthy patients and these two anomalies. In specific, the final aim or purpose of the proposed software in this work is to be used in an automatic screening of these diseases making assessment analysis possible for the population that is at-risk.

1.2 Conventional Methods

Several methods have been developed for the segmentation of the OD and cup regions from 2D color fundus images, with more methods on OD, but with fewer on cup due to the cup's interweavement with blood vessels and surrounding tissues. A multi resolution sliding band filter (SBF) is applied for OD segmentation. A low resolution SBF and a high resolution SBF are used to obtain a set of pixels associated with the maxi-mum responses giving a coarse estimation of the OD boundary. This estimation is regularized using a smoothing algorithm. Principal component analysis (PCA) and mathematical morphology based method is used to extract OD contour. This method employed different techniques such as generalized distance function, stochastic watershed, and geodesic transformation for segmenting OD. The contour of OD can be estimated as a circle or an ellipse as the shape of OD is round or vertically slightly oval. A template-based methodology is used for detecting OD contour. Morphological and edge detection methods followed by Circular Hough Transform can be applied to get a circular OD boundary estimation. A location methodology based on a voting-type algorithm is used to find the location of OD. OD contour was estimated by the Hausdorff based template matching between the detected edges and the template of circle with different sizes.Very few methods have been proposed for cup segmentation from color fundus image. 3D depth information is used to detect cup boundary. Thresholding is used to get the set of potential pixels which corresponds to the cup boundary.

1.3 Drawbacks of the Conventional Methods

The conventional methods show assurance in capturing a range of shape and image variations. However the accuracy in the segmentation is sensitive to the contour initialization and also in cases where the object to be segmented cannot be easily distinguished in terms of global statistics and may lead to erroneous segmentations. In cup segmentation, approaches are highly dependent on the preliminary cup boundary obtained. Furthermore, the statistical rules used for selecting vessel pixels are very sensitive to the inter-image variations. Also, in regions where the vessel kinks are absent, these methods failed to have reasonable accuracy. To meet these issues, a new optic cup segmentation method is proposed that use both structural and gray level properties of the OD region.

In case of AMD , the conventional methods are time consuming and not precise as required. Various features such as the Retinal blood vessel extraction requires a facsimile like results, hence we use machine learning tool (SVM) that extract features precisely and arrives to a decision that is unfailing.

1.4 Modern technology methods going to be employed

The ratio of the size of the optic cup to the optic disc, also known as the cup-to-disc ratio (CDR), is an important indicator for glaucoma assessment. In clinical practise, CDR is measured manually which can be subjective, limiting its use in screening for early detection.Hence we describe a method for automatic calculation of CDR, optic cup and optic disc are needed to be segmented. Generally estimating the boundary of optic cup is too challenging because of inter weavement of blood vessels around the optic disc. In this paper, they proved that estimation accuracy of the boundary can be improved through convex hull based neuro-retinal optic cup ellipse optimization algorithm. Results proved that the new proposed algorithm outperformed ARGALI(Automatic cup to disc Ratio measurement system for Glaucoma Detection and Analysis) system.

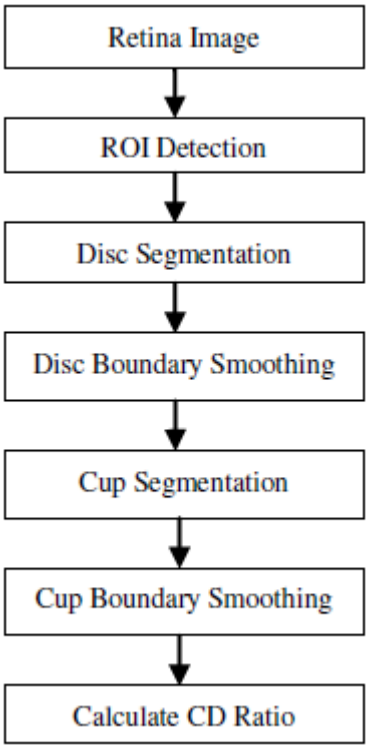


Fig 1.4.1: CDR calculation in Glaucoma Diagnosis

For the detection of AMD, a computer aided machine learning model is used as it gives a probable result after which patient can go for proper treatment.Support vector Machine (SVM) is used for supervised learning of fundus images of both affected and unaffected eyes.SVM models are associated with learning algorithms that analyze data used for classification and regression analysis. Pirbhai and Sheidow (Pirbhai and Sheidow,2004) showed that telemedicine can be used to diagnose AMD patients with high confidence using non-stereo CFP. The automatic system could achieve 89.2% sensitivity and 99.1% specificity for detecting CNV. For the presence of RPE GA the specificities were up to 86.8%. Very few treatable lesions were missed in AMD patients.

Below shown is the working model of SVM which requires training of datasets.Training involves DIP operations like preprocessing to increase contrast,noise removal,etc. feature extraction which are correlated to the extracted feature of test fundus image to give accurate response.

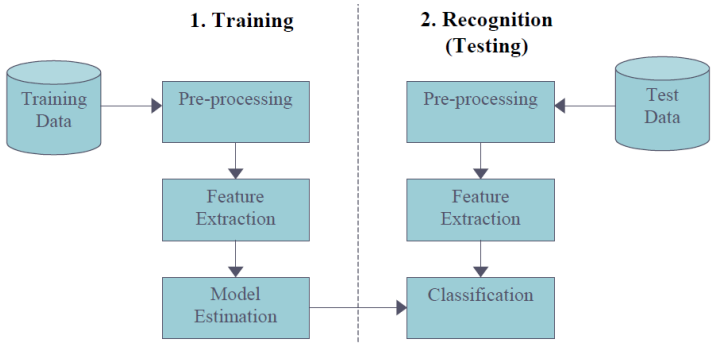


Fig 1.4.2: SVM working model

1.5 Advantages of the modern technology method going to be employed

Computer-aided digital image processing screening technique eliminated the need of a specialist (ophthalmologist) for detection. Thus making it a less extensive task and low chances of error. It gives faster result and the data can be used accordingly,i.e stored to some database, sending results to healthcare institutions or patients or for further studies. It can be employed for a specific individual or mass population both. Telemedicines when conjuncted with DIP detection techniques can be of greater advantage as it will make detection and communication to remote areas easier. Henceforth, the automatic detection technique using the fundus image to differentiate between without pathology and diseased eye is cheaper also with a better efficiency than conventional methods adopted.

2.Literature survey

2.1. Survey on Glaucoma and Age related Macular Degeneration

Glaucoma is a condition that causes damage to the optic nerves that carry information from the eye to the brain. It is always associated with increase in the intraocular pressure (IOP) of the eye. Usually glaucoma is inherited and has few or no symptom. If left untreated or uncontrolled glaucoma may lead to peripheral vision loss initially and eventually lead to permanent blindness.

Survey by glaucoma society of India have revealed that glaucoma is the third leading cause of blindness in India.12 million people are affected accounting for

12.8% of the countries blindness. Population based studies a prevalence between 2% to 13%.

Age-related macular degeneration (AMD) is a worsening disintegration or disruption of the eye's macula. The macula is a little range in the retina the light-inner tissue covering the back of the eye. The macula is the piece of the retina that is in charge of focal vision, permitting to see fine points of interest plainly.

Most of the aged elderly individuals intensifying develop macular degeneration as instinctive aged declining process. There are various types of macular issues, however the most well-known is age-related macular degeneration, macular degeneration have manifestations, for example, fogginess, dim zones or mutilation in your focal vision, and maybe perpetual loss of focal vision. Reasons for macular degeneration comprise the structural pattern of deposits termed drusen below the retina, and now and again, the development of aberrant unusual blood vessels under the retina. With or without treatment, macular degeneration alone never causes add up to visual deficiency.

2.2. Abstraction of few of the related works

Several studies are reported in literature for detection of optic disk and detection and classification of glaucoma and macular degeneration. The work is as follows:

Enhancement of retinal fundus Image to highlight the features for detection of abnormal eyes. This work specifies the methods used to detect main features of retinal fundus images such as optic disk, fovea, and exudates and blood vessels using different techniques. To determine the optic Disk and its centre Author find the brightest part of the fundus and apply Hough transform.

In Year 2010, Vahabi Z proposed ;The new approach to Automatic detection of Optic Disc from non-dilated retinal images. Author describes an ewfiltering approach like Sobel edge detection, Texture Analysis, Intensity and Template matching to detect Optic Disc. The proposed algorithm is applied In wavelet domain on 150 images of Messidor dataset.

In Year 2012, Nilanjan Dewy performed a work, "Optical Cup to Disc Ratio Measurement for Glaucoma Diagnosis Using Harris Corner. In this paper, CDR Is determined using Harris Corner. Harris corner detector measures the local changes

of the signal with patches shifted in different directions by a small amount. It is based on the local autocorrelation function of a signal.

In Year 2014, Dnyaneshwari D. Patil, Ramesh Manza, Gangadevi C. Bekde performed a work , & quot;Diagnose Glaucoma by proposed Image processing Methods. In this paper,a Glaucomatous fundus image is classified using neural network developed by Donald Speech. In Mookiah et al. [1], an alternate approach for AMD portrayal is carried with local configuration patterns (LCP) in preference to LBP. Pattern occurrence features and linear configuration coefficients were extracted and after feature selection a linear SVM was used.

The drawback of this work is although identification of three classes was done they focused only on DR detection and exudates segmentation was required for feature extraction addition to with main structures segmentation (vessels and optic disks).In Garnier et al. deal. [3], a preliminary study for AMD detection from colour fundus photographs is gathered.To bypass the problem of curse of dimensionality and image classification a linear discriminant characterization analysis is made use for feature dimension reduction. In T. Ojala, M. Pietikinen, and T. Menp [9], this work introduces a theoretically exceptionally direct yet efficient multiresolution strategy to rotation invariant texture and gray scale classification based on nonparametric discrimination of sampling and model distributions and local binary patterns. The approach is based on identifying that some local binary patterns termed ‘identical or uniform’ are inherent fundamental properties of local image texture, and occurrence of histogram determines to be a dominant texture feature, a discerned rotation invariant operator and gray scale presentation is derived it allows to detect the ‘uniform’ patterns for any spatial resolution and for any quantization of the angular space that presents a approach for joining combined multiple operators on multiresolution analysis.

The drawback of this work is the spatial support built in operators are inherently larger, only a bound subset of patterns can endure adjoining to a particular pattern only a bounded subset of patterns can reside and operators may not be relevant to discriminating textures where the preeminent features occur at a large scale. In T. Ojala, M. Pietikinen, and T. Menp [7] on the base of local binary patterns this work presents principle to the rotation invariant texture classification and gray scale method.In L. Nanni, A. Lumini, and S. Brahnam [11], this work focuses on machine learning techniques on the image-based uses in medical image analysis, also in this work local binary patterns (LBP)

variants are presented, that are widely accepted as the state of art in texture descriptors, a detailed description on description review of the literature on present existing LBP variants is provided in the work and has been discussed on the mostly concentrated salient approaches with their pros and cons, making use of many of the several LBP-based descriptors new experiments are reported and for the biomedical images representation a set of standard novel texture descriptors are been proposed. The drawback of this work is the histograms concatenations are obtained differing some of the parameters that leads to a features high correlations.

2.3. Gaps Identified

- Retinal disease screening needed previous segmentation stages and would handle only one disease at a time
- High dimensionality of the feature vectors.
- Big feature size of the proposed matching schemes.
- Values to determine the accuracy of the normal and Glaucoma discrimination were not provided.

3.Objectives

To develop an automatic GSM enabled screening system capable of discriminating between normal hea lthy fundus, Glaucoma and AMD. The proposed work performs automatic screening of the diseases for feasible assessment of the at-risk population. The system will use digital image processing done in MATLAB and the final result will be sent to patient via GSM module.

. The above mentioned objective of our project work can be achieved using the following steps one by one as follows :

Glaucoma:

1. Loading of fundus image
2. Cup extraction using IP operations:
   - Extract RGB component of fundus image
   - Median filtering on R component
   - Thresholding
   - Cup region extraction
3. Disc extraction
   - Locate ROI
   - Extract RGB component
   - Create circular mask image
   - Convolution of G component and mask image
   - Erosion & Opening
   - Disc approximation and smoothinig
4. Calculate CDR
5. Analyze result

AMD:

1. Training of diseased dataset
2. Training of normal dataset
3. Validation of model
4. Loading of test image/directory of test images
5. Feature extraction of test image
6. Analyzing result

4. Motivation

Glaucoma and AMD are two different chronic eye condition that majorly leads to vision loss. As it cannot be cured, detecting the disease in time is important. Current tests for glaucoma detection includes using intraocular pressure (IOP), are not sensitive enough for population based glaucoma screening. Survey by glaucoma society of India have revealed that glaucoma is the third leading cause of blindness in India. Approximately 12 million people are affected accounting for 12.8% of countries blindness.While AMD in 2015 it affected 6.2 million people globally. In 2013 it was the fourth most common cause of

blindness after cataracts, preterm birth, and glaucoma. It most commonly occurs in people over the age of fifty and in the United States is the most common cause of vision loss in this age group. About 0.4% of people between 50 and 60 have the disease, while it occurs in 0.7% of people 60 to 70, 2.3% of those 70 to 80, and nearly 12% of people over 80 years old.

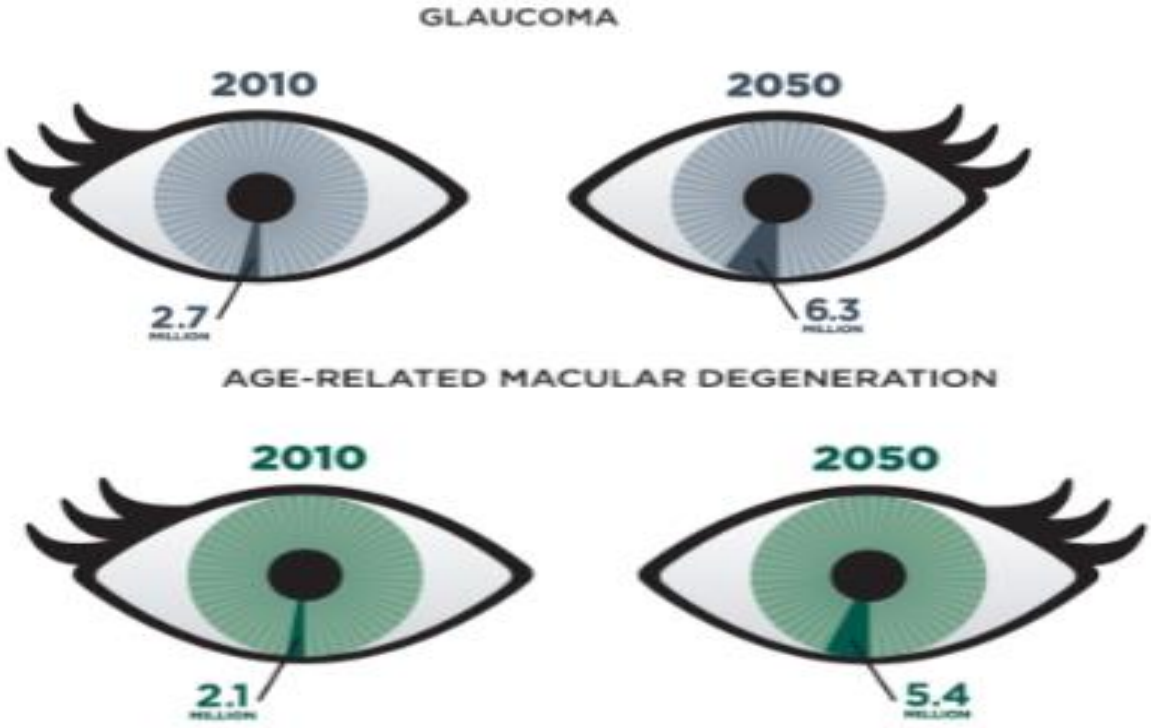


Fig 4.1: Prediction by NEI between 2010 and 2050 affected number of people will double

Hence our goal is to design and develop an automatic computer-aided diagnosis screening software system capable of discriminating between a healthy normal fundus (without pathology), Glaucoma, and age related macular degeneration.

5.Proposed Methodology

The methodology adopted for carrying out the project involves following steps:

- ✓ Image datasets from different databases composed of Fundus images are obtained and for each dataset subsets are generated for age related macular degeneration, and Glaucoma.

- ✓ Image datasets with vascular network, salt and pepper noise, images with highlights around the vessels associated with young retinas, doubtful diagnosis are been excluded as they do not comply to certain quality criteria.
- ✓ The images are resized and masking of the image region is performed to obtain the region of interest.
- ✓ After exclusion the resulting dataset is divided into subsets of two, one for trainin g and other for testing (validation set).
- ✓ Local binary patterns are used for texture description, the images are resized to a standard size to obtain comparable local binary pattern texture descriptors in the preprocessing stage.
- ✓ The local binary patterns and gray level co-occurrence matrix statistical measurement features are obtained in the feature extraction stage and histograms are computed
- ✓ Once the features are extracted data of the model set is preprocessed with data normalization and data resampling tasks carried out before the classification stage.
- ✓ External cross validation also called nested cross validation is performed on the model set to reduce the dimensionality of the data by feature selection before being passed on to a classifier.
- ✓ For subset selection of the feature set a final classifier is obtained for the whole model and this model set is been used for training the classifier.
- ✓ On the final classifier the validation set is tested and the results are obtained.

5.1. Glaucoma

Glaucoma Detection involves two main methodologies which include cup and disc extraction, which are as follow:

1. Cup Extraction
2. Disc Extraction

Cup Extraction Flow:

Separate RGB components

↓

Median Filtering on R component

↓

Cup Estimation based on Threshold

↓

Cup region Extraction

Disc Extraction Flow:

Finding ROI

↓

Finding 7th Bit plane of Red Index

↓

Erosion after Convolution

↓

Erosion after Opening

↓

Disc Region Extraction

5.2. AMD Detection

Age Macular Degeneration(AMD) involves two methodologies as follows:

1. Process of Detection
2. Training Methodology for Normal and diseased images

Process of Detection Flow:

Train Diseased Database
↓
Train Normal Database
↓
Validate using SVM
↓
Load Test Image
↓
Result

Training Methodology Flow :

Load Image
↓
Morphological Processing (Opening)
↓
Edge Detection
↓
Vessel Detection
↓
Feature Extraction for Optical disc removed image

6.Tools

6.1. Hardware

I. Node Micro controller Unit

Fig 6.1.1 Node Micro controller unit

The **Node** Micro Controller Unit is an open source software and hardware development environment that is built around a very inexpensive System-on-a-Chip (SoC) called the ESP8266. Main features include

- Wi-Fi Module: ESP-12E module similar to ESP-12 module but with 6 extra GPIOs
- USB: Micro USB port for power, programming and debugging
- Headers: 2x 2.54mm 15-pin header with access to GPIOs, SPI, UART, ADC and power pins
- Miscellaneous: Reset and flash buttons
- Dimensions: 49 x 24.5 x 13mm

We connect one end to a USB port of our pc and inserted the other end to the USB port of Node mcu and set the com port number in Arduino IDE.

II. GSM SIM800

Fig 6.1.2 Gsm module sim800A

GSM is a mobile communication modem; it is stands for global system for mobile communication (GSM). SIM800 is a complete Quad-band GSM solution in a SMT type which can be embedded in the customer applications

Node mcu communicate with Gsm module and it sends sms on Sim 800 and forward that sms to the patient.

It can perform the following operations:

1. Receive, send or delete SMS messages in a SIM.
2. Read, add, search phonebook entries of the SIM.
3. Make, Receive, or reject a voice call.

6.2. Software

1. Windows Operating System 10
    - It is the latest and greatest update of all the windows editions. Even critically acclaimed.
    - Stability and Performance has been greatly increased.
2. MATLAB R2017a
    - MATLAB can control popular microcontrollers like Arduino and Raspberry Pi, acquire images from webcams, and even collect data from the sensors built into your smartphone.
3. Arduino IDE (Integrated development environment)
    - Arduino IDE that runs on our computer is used to write and upload computer code to the physical board.

7.Algorithm

7.1. Glaucoma

Purpose: Pre-processing of the input image

Input: Fundus image

Output: Preprocessed grayscale image

Algorithm steps:

1) Load the image into the workspace
2) Extract the RGB components
3) Obtain the median filter with R index
4) Perform Hist/Gray threshold analysis
5) Extract cup based on thresholding
6) Perform morphological processing and get the properties of an image
7) Obtain the outer circle and perform edge detection

7.2. Age Related Macular Degeneration

Purpose: To extract local binary pattern features

Input: Gray scale image

Output: LBP Features of fundus image

Algorithm steps:

1) Load the training dataset for feature extraction
2) The examined window is divided into cells (e.g. 8 × 8 pixels for each cell).
3) Pixels of its 8 neighborhood is compared with each cell and the pixels are followed along a circle, i.e. counterclockwise or clockwise.
4) On the neighbor's value greater than the center pixels value, count as "1" otherwise, count "0", outcome is an 8-digit binary number that has been decimal converted.

5) For the frequency of each “number” occurring over the cell, compute the histogram.
6) Optionally histogram is been normalized.
7) All cells histogram are concatenated to obtain the feature vector of the window.
8) Load the test image for feature extraction and repeat from step(2-7).
9) Compare the test image features to the training dataset features and classify the image.

8.Working

8.1. Glaucoma

8.1.1. Background

At the lowest level of abstraction both output and input image intensities are operated in pre-processing to enhance and improve the image data by suppression unwanted distortions and enhancing the important image features for further processing.

8.1.2. Preprocessing

Image pre-processing is a typical approach for pre-processing computations at the lowest level of abstraction for images; both output and input are intensity images. Preprocessing operations involve those functions which normally require prior main data analysis performance and information extraction. Images in pre-processing generally involve low frequency removal of background noise, removing reflections, the individual particle images are normalized with intensity level and masking portions of images.

The improvement of the image data is the main purpose of image pre-processing in which important image features are enhanced and unwanted distortion are suppressed for further processing.

The reliability can be significantly increased with pre-processing operations. Certain image details can be reduced with several filter operations for intensifying images that enable a faster or easier evaluation.

The image dataset is formed with different images obtained from various databases with normal, diabetic retinopathy and age related macular degeneration images.

Pseudo code

Objective of the Module: To perform image pre-processing

Input to Module: Color fundus image

Output of the Module: Pre-processed gray scale image

Description of the Module: The program performs pre-processing of the image by noise removal, filtering, obtaining the required properties of the image by morphological processing for extraction of information for further processing.

Step 1: Loading Image

Here the image is loaded from the database into the workspace and is been displayed. Pseudo code:

1. Read the image into workspace using command *'imread'*.
2. Display the digital input image using command *'imshow'.*

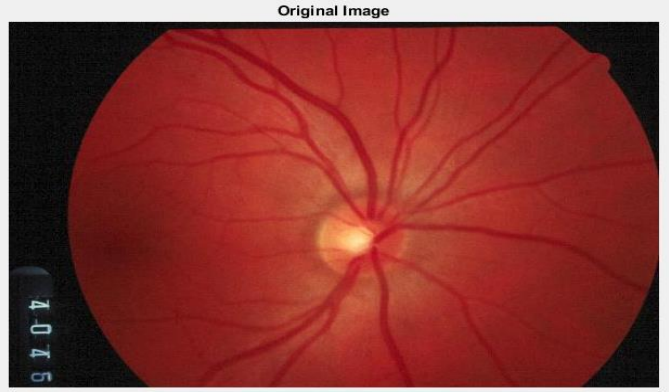


Fig 8.2.1.1: Image read

Step 2: Extract RGB Component

Here the extraction of red, green, blue channels of a color image is been done.

Pseudo code:

1. Define and obtain the Red color plane of the image
2. Display the red index of the image
3. Define and obtain the Green color plane of the image
4. Display the green index of the image
5. Define and obtain the Blue color plane of the image
6. Display the blue index of the image

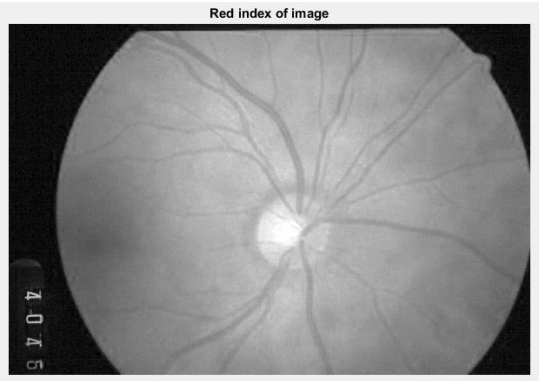


fig 8.2.1.2: Separation of RGB index

Step 3: Median Filter with R index

Performs median filtering, where each output pixel contains the median value in the m/n neighbourhood around the corresponding pixel in the input image.

Pseudo code:

1. Obtain the median value for the red index image with set window size
2. Display the median value of the red index image

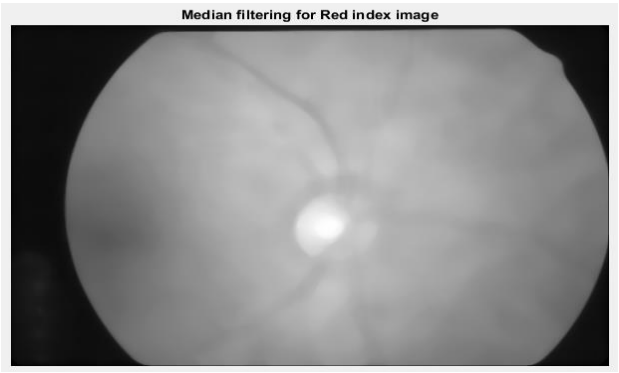


fig 8.2.1.3: Median Filtering

Step 4: Hist/ Gray threshold analysis

Here the image color channels are gray thresholded to obtain the binary image.

Pseudocode:

1. Color images are converted to grayscale before *level* is computed.
2. Given an image finds the optimal threshold value *level* for conversion to a binary image with im2bw.
3. An image histogram '*hist'* used to allow for preprocessing of the histogram.

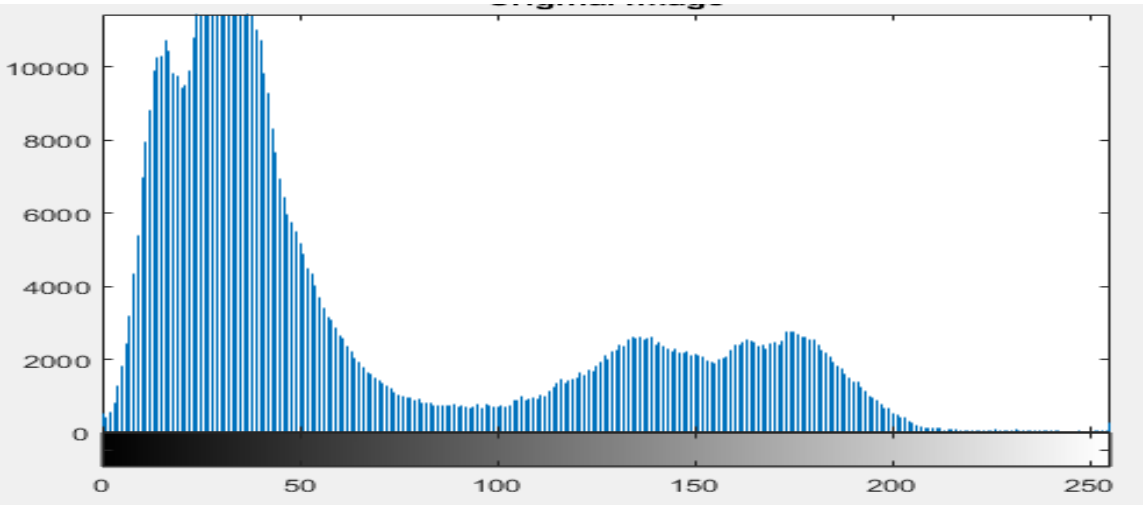


Fig 8.2.1.4: Input image histogram

Step 5: Extract Cup based on thresholding

Here the cup region is obtained to find the disk.

Pseudo code:

1. Obtain the R_med of the image
2. Define the mean adjusted dataset for the matrix[m.n]
3. For i = 1 to m and j= 1 to n
4. If image(i,j)<240
5. Set image(i,j) to zero
6. Display the image in the workspace

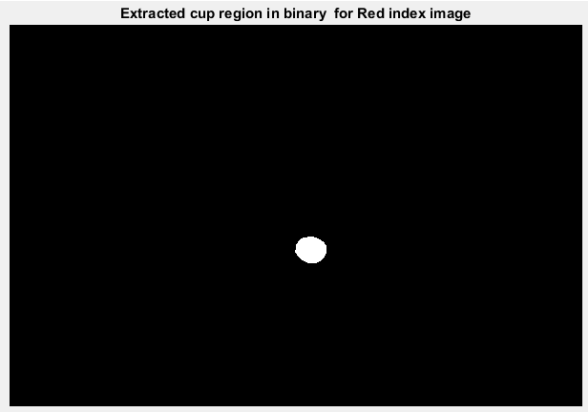


fig 8.2.1.5: Cup extraction based on threshold

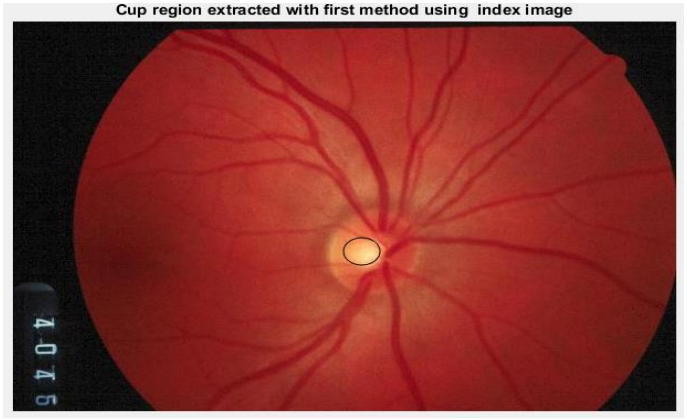


fig 8.2.1.6: Cup region extraction

Step 6: Morphological processing

Here the properties are obtained to get region of interest.

Pseudo code:

1. Display the image in the workspace
2. Obtain the image properties such as BoundingBox, Curvature, EdgeColor

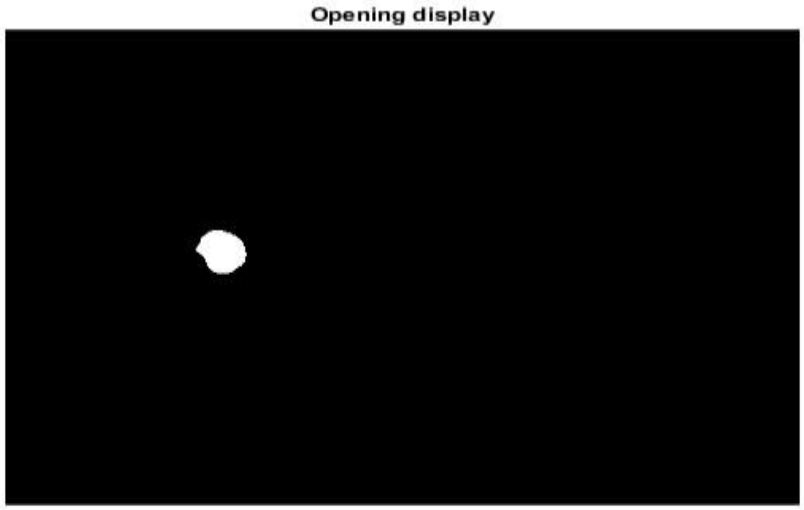


Fig 8.2.1.7:The morphological processing-opening

Step 7: Outer Circle

Here we will be repeating the process for green component.

Pseudo code:

1. Get image and mode from workspace

2. Apply filter on green component
3. Create averaging filter
4. Convert back from (0,1) to (0,255)
5. Display the image in the workspace

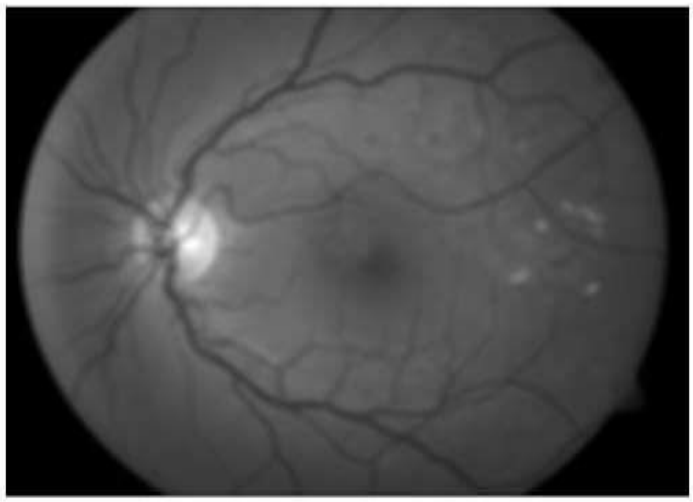


fig 8.2.1.8 :The outer circle of the green channel image loaded into workspace.

Step 8: Edge Detection

Here the disk is obtained as region of interest and rest of the area is been masked with zero.

Pseudo code:

1. Set the threshold value % default = 0.98
2. Obtain the binary mask or black and white threshold
3. Edge detection
4. Obtain the region properties shape, area, centroid, bounding box using region props
5. Define coordinates and radius
6. Generate grid with binary mask representing the circle.
7. Mask the original image

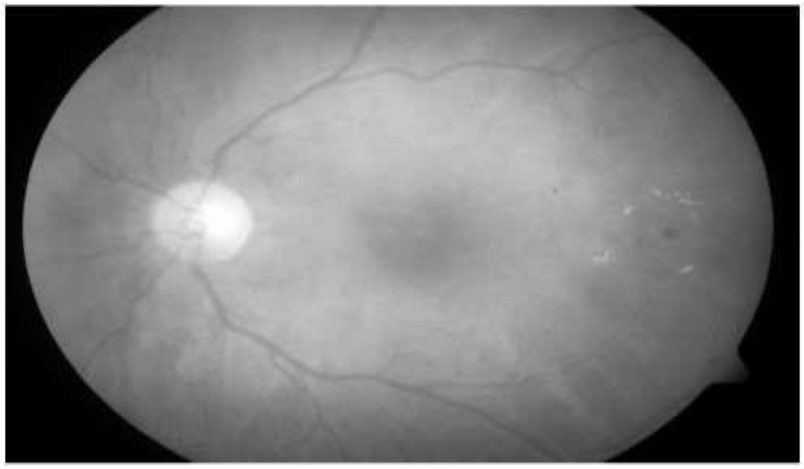


fig 8.2.1.9: The edge detection outcome

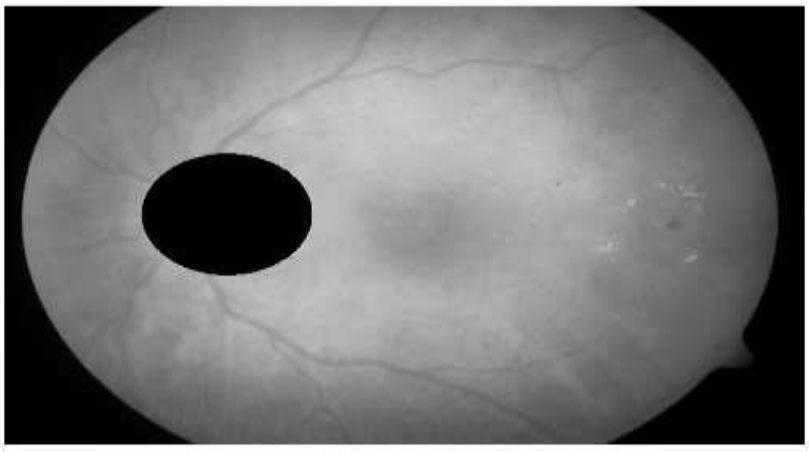


fig 8.2.1.10: The outcome of masking the disk area

8.2. Feature Extraction

8.2.1. Background

The spatial distribution information of total band variations are obtained from textural features and all the textural information is contained in the gray level co-occurrence matrices and local binary patterns hence all the textural features are extracted from local binary patterns and gray level co-occurrence matrices. The different statistical features of image extraction are:

- Entropy (A(X)) is statistical measurement of disorder in grayscale image, it is given by the equation.

$$A(X) = \sum_k P_k \, Log_2 \, P_k \ldots\ldots\ldots (1)$$

- Third moment (α_3) measurement is by asymmetry of the histogram.

$$\alpha_3 = \frac{\sum(Y_1 - Y)^3}{(M-1)^3} \ldots\ldots\ldots\ldots (2)$$

- Contrast is intensity contrast between a pixel and pixels neighbor, its value is zero for a constant image.

$$\sum_{kl} |k - l|^2 \, p(k, l) \ldots\ldots (3)$$

- Correlation is the extent measurement of interrelationship between a pixel and to its pixel neighbor for the entire image.

8.2.2. Local Binary Patterns

Local binary patterns (LBP) operators are powerful means used for texture description. Image pixels are labeled using the original version of the operator, the 3 × 3 neighborhood of each pixel are thresholded by the center value and threshold values are summed weighted by power of two.

Neighborhoods of different sizes can be used by extending the local binary patterns operator, to do this a circular neighborhood denoted by (S,N) is defined, here S represents the number of sampling points and N is the radius of the neighborhood. These sampling points around pixel (Q,R) lie at co-ordinates given by equation

$$(qs, rs) = \left(q + Rcos\frac{2\pi p}{s}, r - Rsin\frac{2\pi p}{s}\right) \ldots\ldots\ldots\ldots\ldots\ldots\ldots (4)$$

Pseudocode

Objective of the Module: To obtain texture features of an image.

Input to module: Color or Grayscale Image.

Output of the Module: Texture features of an image.

Description of Module: The program will compute the LBP features. If the input image is colored then it must be converted to grayscale to perform computation on the gray scale image. The original version of the operator labels the image pixels by thresholding the 3 × 3 neighborhood of each pixel with the center value and summing the thresholded values weighted by powers of two and achieving LBP invariant rotation operator.

//**input:** read a grayscale image

//**output:** features of the image

//input the converted gray scale image and derive statistics from LBP

Calculate Contrast, Correlation, Energy, Homogeneity, Mean, Standard_Deviation etc.

//store all the features of input fundus in a variable [Contrast, Auto-Correlation, Sum of squares: variance, Sum average, Difference variance, Sum Entropy, Difference Entropy, Information measure of correlation]; End;

8.2.3. Gray level co-occurrence Matrix

A co-occurrence matrix or co-occurrence distribution (less often co-occurrence matrix or co-occurrence distribution) is a matrix that is defined over an image to be the distribution of co-occurring values at a given offset. Mathematically, a co-occurrence matrix C is defined over an s × t image Y, parameterized by an offset (δu, δv) the co-occurrence matrix is defined as follows:

$C_{\delta u,\delta v}(m, n) = \sum^{s}_{q=1} \sum^{t}_{r=1}$ {1, Y (q, r) =m and Y (q+ δu, r+ δv) = n,

0, otherwise…… (5)

Where m and n are the image intensity values of the image, q and r are the spatial positions in the image Y and the offset (δu, δv) depends on the direction used and the distance at which the matrix is computed. Value of the image is originally referred as the gray scale value of the specified pixel, but could be anything, from a binary on/off value to and beyond 32-bit color.

In gray-level co-occurrence matrix the spatial relationship of pixels is examined considering statistical method, it is also known as the gray-level spatial dependence matrix. The texture of an image is characterized using GLCM functions by calculating pixel pairs occurring with specific values and how an specified spatial relationship occurs in image, creating a GLCM, and then by statistical measure extraction from this matrix.

The graycomatrix function is used to create a gray-level co-occurrence matrix (GLCM). A GLCM is created using graycomatrix function by calculating how often a pixel with the intensity (gray-level) value occurs in a specific spatial relationship to a pixel with the value. The spatial relationship is defined as the pixel of interest and the pixel to its immediate right (horizontally adjacent), but can specify other spatial

relationships between the two pixels. Each element (i , j) in the resultant GLCM is simply the pixel count of sum of the number of times pixel with value i occurred in the spatial relationship to the pixel with value of the input image. The size of the GLCM is determined by the number of gray levels in the image. The number of intensity values of an image is reduced to eight using graycomatrix with scaling. Certain properties of the gray levels of the texture image about spatial distribution can be obtained by gray-level co-occurrence matrix.

9. Training and Classification (SVM)

9.1. Overview

Classification divides the spectral or spatial feature space into several classes based on a decision rule to extract the pixels to be used for training the classifier to recognize certain categories or classes and determine the discriminant functions in the feature space.

9.2. Training

Training is the process of defining criteria by which spectral patterns are recognized.

There are three methods available for training the network.

- Supervised learning
- Unsupervised learning
- Reinforcement learning

Supervised learning is a method of learning in which the system is exposed to a set of inputs along with the actual values that are expected to be produced after processing the input vector. The data set considered for training process is termed as training set which is nothing but a collection of training examples. So, a pair of an input vector and the expected value of output constitute a training example. Initially each weight is assigned some random numbers between -1 and +1. The network is then made to process the input vectors and comparison is done between the predicted and the actual value, then the error calculation is done which will be propagated in backward direction to all layers from last to first. Depending on the error, all weights in the network are adjusted and the processing is repeated. The training is performed over the same set of data repeatedly until the error reduces to a considerable amount.

Unsupervised learning is also called as adaption learning. In this case, the network will be provided with only the input data but not the desired output and the network is expected to learn from the input data by observing the regularities existing in them.

Reinforcement learning makes the machines learn from the feedback they receive from the environment for their actions in order to maximize the performance. The feedback that the machine receives is a numerical reward and is termed as reinforcement signal.

The learning process depends completely on the decision making capacity of the machine as they are not provided with any other form of supervision except the reward.

9.3. Support vector machine

SVMs (Support Vector Machines) technique is most useful for data classification. Data is separated into training and testing sets involving classification task. Each instance in the training set contains several attributes (i.e. the features or observed variables or the features) and one \target value" (i.e. the class labels). When the test data attributes are given the target values of the test data are predicted by SVM model (based on the training data). A Support Vector Machine (SVM) is a discerning classifier precisely characterized by a separating hyperplane, the algorithm outputs an optimal hyperplane that categorizes new problems based on the supervised learning given labelled training data.

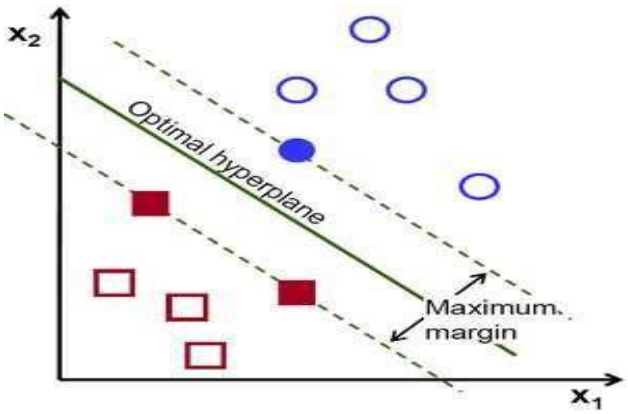


fig 9.3.1: Finding an optimal hyperplane

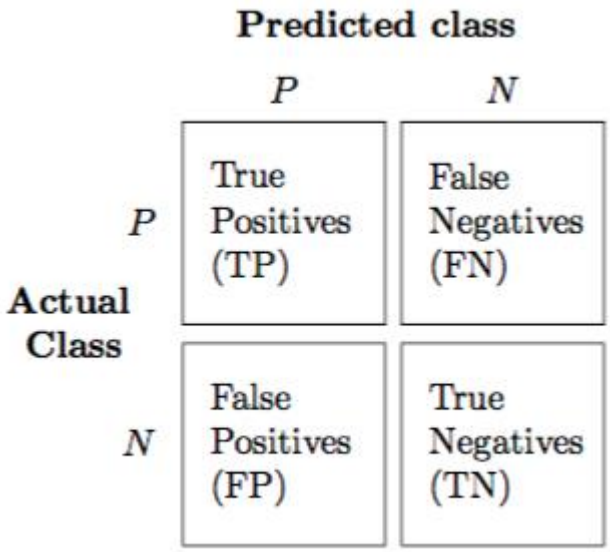


Fig 9.3.2 Confusion matrix

ALGORITHM: Support Vector Machine Classifier

Inputs: Input image, Training dataset (.mat files)

Outputs: Type of diseased or healthy fundus.

//Load the train data and select texture features for classification

Load the training dataset files //Load the test features into variable test diseaseset;

//To classifies the correct class use multisvm. Assign resultant to the variable

classes = svmclassify(svmStruct,data(test,:);

//Print the type of image by comparing the SVM value If (result==0) output "Normal"; else if(result==1) output "AMD diseased"; end

Pseudocode

Objective of the module: To obtain the class of input image.

Input to module: Grayscale image.

Output of the module: Class of input image.

Description of module: This module accepts the test image as input and matches it to one of the classes present in the training dataset, based on the feature values extracted from the input image. It classifies each row in test using the support vector machine classifier structure SVMSTRUCT created using svmtrain, and returns the predicted class level group. Test must have the same number of columns as the data used to train

the classifier in svmtrain. Group indicates the group to which each row of test is assigned. Based on that the result is assigned to variable result.

Pseudo code:

//input: input the test image and trained dataset features

//output: category of fundus input image

//input the test image and trained dataset features and store the result svmStruct = svmtrain(data(train,:),groups(train); classes=svmclassify(svmStruct,data(test,:); //compareresult and get the type of disease

if result_class==0

Normal;

else

AMD Diseased;

10. Testing

Each program components are tested for errors to discover defects in the testing stage. The components may be any of the program functions, objects or modules. The integrated components are used in system testing to form the complete system. In this stage testing must be focused to establish the system that meets functional requirements and must be ensured that system does not behave in an unexpected way. Test data are inputs that are been devised and trained to test the system whereas test cases are inputs used to test the system and if the system works as specified with the given input the output is specified for the given input, the behavior is examined in a cohesive system. The test cases are opted for ensuring that the system behavior is examined in all the possible combinations of conditions considered.

Accordingly, system behavior that is expected under various combinations of conditions is given. Therefore test cases are selected which have inputs and the outputs on expected lines, inputs that are not valid and for which suitable messages must be given and inputs that do not occur frequently which can be regarded as special cases.

10.1. Testing Strategy

The strategy that is used to perform unit testing is described below:

- **Features to be tested** – The features to be tested, most importantly include the operation of individual component for the proper execution of the entire program.
- **Items to be tested** – The items to be tested include all the individual units or functions, which collectively form the whole system. In case of unit testing the items to be tested, are the main graphical user interface, deploying the sensor nodes and handling the events in the sensor network.
- **Purpose of testing** – The purpose of the testing is to check the unit functionality of the main source of the project.
- **Pass/Fail Criteria** – The pass or fail criteria are designed with the basis of appropriate compilation of the main source file.

11. Experimental Result

The outcome of the project work could be summarized as follows

- Glaucoma detection (Matlab simulation)
- AMD detection (Matlab simulation)
- Gsm enabled (hardware implementation)

1. Glaucoma Detection

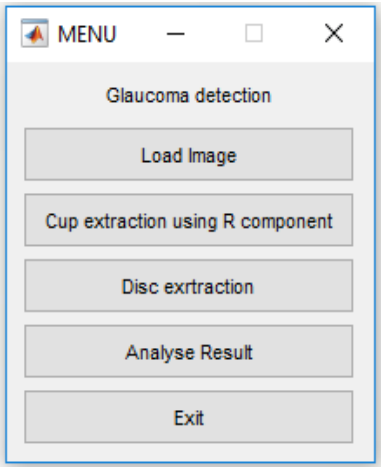


Fig11.1 Menu for Glaucoma detection

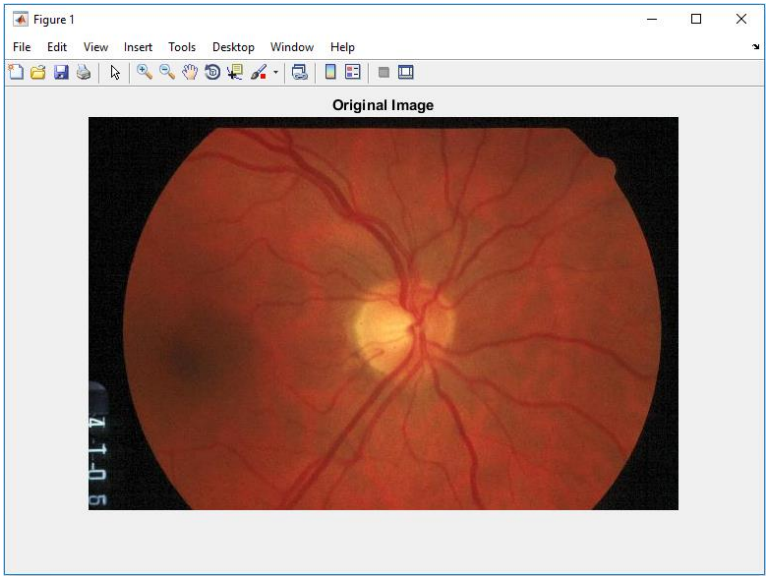


Fig11.2 Original input image

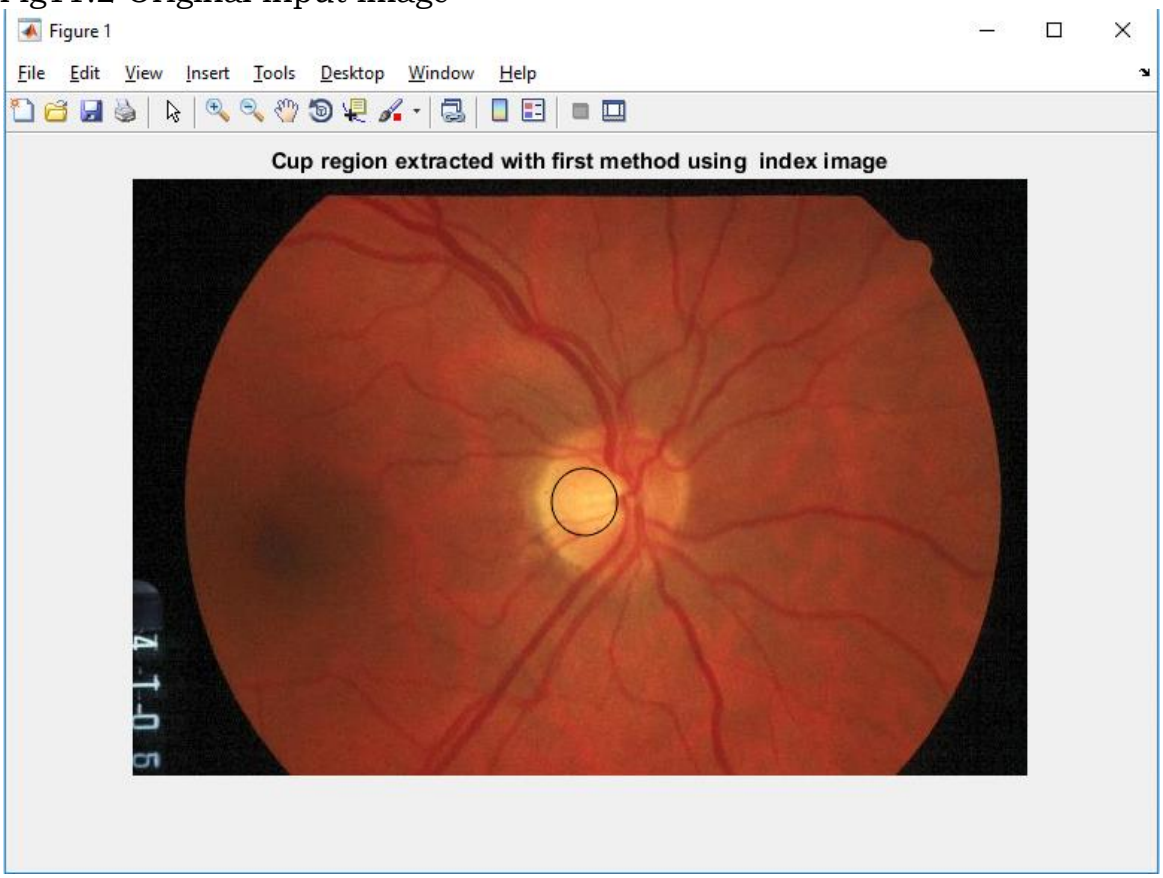
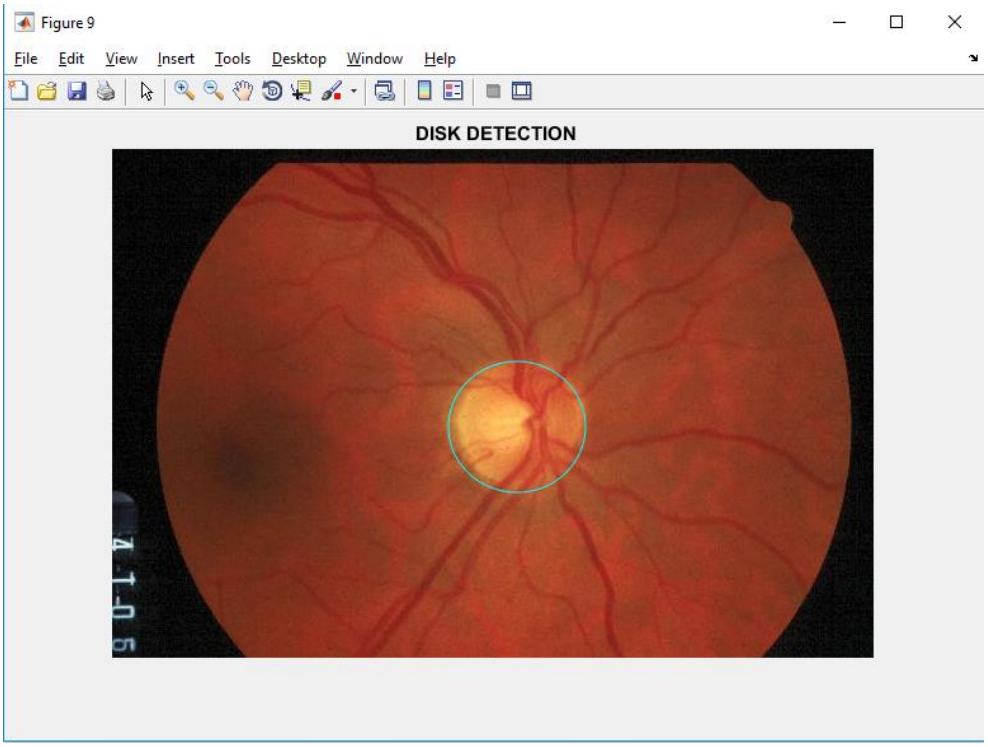


Fig11.3



12. Application of project

Our Project deals with detection of Age Macular Detection and Glaucoma which plays a major role in the medical field . It helps to create awareness among the people so that they could take initial measures.Our project can be implemented to use in rural areas for large scale detection of these diseases.

13. Conclusion

As a result of the work, an automatic disease screening software is developed for identifying and distinguishing the AMD, Glaucoma and normal images. GLCM and LBP are used for feature extraction which makes it a better approach compared to other methods due the availability of a large number of features. The system is made to learn from the training data and is tried with another set of data of testing phase with support vector machine for guaranteeing the system legitimate working.Glaucoma on the other hand is detected by calculating the CDR using and its result are not probabilistic.Thus making a system to determine the stage of disease easily.

14. Future Work

In our proposed project we use machine learning for detection of Age Macular Detection ,the training process for the images in the machine learning used by us is time consuming, future work can be done to improve the time constraints. Considering Glaucoma, our detection method deals with adults and not with children. Future work can be done on glaucoma detection in children.

15. References

[1] M. R. K. Mookiah, U. R. Acharya, H. Fujita, J. E. Koh, J. H. Tan, K. Noronha, S. V. Bhandary, C. K. Chua, C. M. Lim, A. Laude, and L. Tong, "Local configuration pattern features for age-related macular degeneration characterization and classification," Computers in Biology and Medicine, vol. 63, pp. 208 – 218, 2015.

[2]M. M. R. Krishnan and A. Laude, "An integrated diabetic retinopathy index for the diagnosis of retinopathy using digital fundus image features," Journal of Medical Imaging and Health Informatics, vol. 3, no. 2, pp. 306–313, 2013.

[3] M. Garnier, T. Hurtut, H. Ben Tahar, and F. Cheriet, "Automatic multiresolution age related macular degeneration detection from fundus images," in SPIE, Proceedings, vol. 9035, 2014, pp. 903 532–903 532–7.

[4]https://www.aao.org/eye-health/diseases/amd-macular-degeneration

[5]https://nei.nih.gov/health/diabetic/retinopathy

[6]http://www.scholarpedia.org/article/Local_Binary_Patterns

[7] T. Ojala, M. Pietikinen, and T. Menp, "A generalized local binary pattern operator for multiresolution gray scale and rotation invariant texture classification," in Advances in Pattern Recognition, 2nd International Conference on, 2001, pp. 397–406.

[8] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns:

Application to face recognition," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 28, no. 12, pp. 2037– 2041, 2006.

[9] T.Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 24, no. 7, pp. 971–987, 2002.

[10] Zhenhua Guo, Lei Zhang, David Zhang, "Rotation invariant texture classification using LBP variance (LBPV) with global matching", Pattern Recognition ,pp. 706–719, 2010.

[11] L. Nanni, A. Lumini, and S. Brahnam, "Local binary patterns variants as texture descriptors for medical image analysis," Artificial Intelligence in Medicine, vol. 49, no. 2, pp. 117 – 125, 2010.

[12] H.A. Quigley, A.T. Broman, The number of people with glaucoma worldwide in 2010 and 2020, Br. J. Ophthalmol. 90 (2006) 262–267.

[13] M.A. Kass , D.K. Heuer, E.J. Higginbotham, et al., The Ocular Hypertension Treat-ment Study: a randomized trial determines that topical ocular hypotensive medication delays or prevents the onset of primary open-angle glaucoma,

Arch.Ophthalmol. 120 (2002) 701–713, discussion 829–30.

[14] R. Komulainen, A. Tuulonen, P.J. Airaksinen, The follow-up of patients screened for glaucoma with non-mydriatic fundus photography, Int. Ophthalmol. 16(1992)

465–469.

[15] S.Y. Shen, T.Y. Wong, P.J. Foster, et al., The prevalence and types of glaucoma in Malay people: the Singapore Malay Eye Study, Invest. Ophthalmol. Vis. Sci. 49(2008) 846–851.

[16] R. Bock, J. Meier, L.G. Nyl, G. Michelson, Glaucoma risk index: automated glaucoma detection from color fundus images, Med. Image Anal. 14 (3)(2010)471–481.

[17] G. Michelson, S. Wärntges, J. Hornegger, B. Lausen, The papilla as screening parameter for early diagnosis of glaucoma, Dtsch. Arztebl. Int. 105 (34–35)(2008)583–589.

[18] Y. Fengshou, Extraction of Features from Fundus Images for Glaucoma Assessment, National University of Singapore, 2011, Master's Thesis.

[19] D.H. Sim, L.G. Goh, Screening for glaucoma in the Chinese elderly population in Singapore, Singapore Med. J. 40 (10) (1999) 644–647.

[20] M.L. Huang, H. Chen, J.J. Huang, Glaucoma detection using adaptive neuro-fuzzy inference system, Expert Syst. Appl. 32 (2) (2007) 458–468.

[21] G.L. Spaeth, J. Henderer, C. Liu, M. Kesen, U. Altangerel, A. Bayer, L.J. Katz, J.Myers, D. Rhee, and W. Steinmann, Disc damage likelihood scale: reproducibility of a new method of estimating the amount of optic nerve damage caused by glaucoma, Trans. Am. Ophthalmol. Soc. 100 (2002) 181–186.

[22] Dashtbozorg Behdad, Ana Maria Mendonc¸ a, Aurélio Campilho, Optic disc segmentation using the sliding band filter, Comput. Biol. Med. 56 (2015)1–12.

[23] M.C.V.S. Mary, Elijah Blessing Rajsingh, J. Kishore Kumar Jacob, D. Anandhi,Umberto Amato, S. Easter Selvan, An empirical study on optic disc segmentation using an active contour model, J. Biomed. Signal Process. Control 18 (2015)19–29.

[24] S. Morales, V. Naranjo, J. Angulo, M. Alca˜niz, Automatic detection of optic disc based on PCA and mathematical morphology, IEEE Trans. Med. Imaging 32(2013)
786–796.

[25] Aquino Arturo, Manuel Emilio Gegundez, Diego Marin, Detecting the optic disc boundary in digital fundus images using morphological, edge detection, and feature extraction techniques, IEEE Trans. Med. Imaging 29 (11) (2010)1860–1869.

[26] M. Lalonde, M. Beaulieu, L. Gagnon, Fast and robust optic disk detection using pyramidal decomposition and Hausdorff-based template matching, IEEE Trans.Med. Imaging 20 (2001) 1193–1200.

[27] Z. Liu, O. Chutatape, S.M. Krishnan, Automatic image analysis of fundus photograph, in: Proceedings of 19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 1997, 1997, pp.524–525,2.

[28] S. Tamura, Y. Okamoto, K. Yanashima, Zero-crossing interval correction in tracking eye-fundus blood vessels, Pattern Recognit. 21 (1988) 227–233.

[29] J. Xu, O. Chutatape, E. Sung, C. Zheng, P.C.T. Kuan, Optic disk feature extraction via modified deformable model technique for glaucoma analysis, Pattern Recognit. 40 (2007) 2063–2076.

[30] F. Mendels, C. Heneghan, J.P. Thiran, Identification of the optic disk boundary in retinal images using active contours, in: Proceedings of the Irish Machine Vision and Image Processing Conference, IMVIP 1999, 1999, pp. 103–115.

[31] D.T. Morris, C. Donnison, Identifying the neuroretinal rim boundary using dynamic contours, J. Image Vis. Comput. 17 (1999) 169–174

16. Appendix

Table : Pin Description of Node Micro Controller Unit

Symbol	Pin	Type	Description
RX	21	I	Receiver Input
TX	22	O	Transmitter output
D8/D7/D6/D5 HSPI	5/6/7/16	I/O	General purpose input/output analog pins. HSPI clk /MISO/MOSI/CS
D0	4	I/O	General purpose input/output digital pin for User.
D3	18	I/O	General purpose input/output analog pin for Flashing.
SD3	12	I/O	General purpose input/output digital pin.It can be used as an interface for input/output.
SD2	11	I	Secure digital input pin.
SD1	13	O	Secure digital output pin.
RSV	12	O	PWM(Pulse width modulator) analog output.
RST	03	I	Reset pin
EN	01	I	Enable
VSS	08,15,23,25	I	Ground:0v reference.
VDD	2,19,14	I	3.3v power supply:This is the power supply voltage for code and input/output port.
clk	28	I	Input to the oscillator and internal clock generator circuits

cmd	9	0	command pin.

17. Code

17.1. Matlab code for Glaucoma Detection

```
%% ENCODING CODE
function first_main_gui()
clc;
clear all;
close all;
warning off all
 repeat=1;
 while (repeat==1)
choice=menu('Final year project','Load Image',...
'Cup extraction using R component ',.....
'Disc extraction','vessel elimination','Analyse results','Exit');
switch choice
  case 1
     Load_image
     case 2
     cup_GUI;
       case 4
      disc_GUI;

    case 5
        close all
      vessel_method1_gui;


    case 6
        delete img.mat
        compare_result
```

```
   case 7
      clc
      clear all
      close all
      repeat=0;
end
fprintf('\n')
fprintf('\n')
 end
end
```

```
%Loading image
[FileName1, Path1] = uigetfile ('*.bmp; *.png; *.jpg; *.tiff','Select the Main image');
Source_img=strcat(Path1,FileName1);
img=imread(Source_img);
[m,n,p]=size(img);
opti_disk=img;
figure,imshow(img)
title('Original Image')
save img.mat img
```

```
%% Cup extraction
clc;
clear all;
close all;
warning off all
repeat=2
while (repeat==2)
choice=menu('  CUP extraction ','Extract RGB component',...
   'Median Filtering ','Extract Cup based on threshold',...
   'Cup region Extraction','Return');

case 1
```

```
Extract_RGB_component;
 case 2
       close all;
       Median_Filtering;
          case 3
        close all


     Extract_Cup_based_on_threshold;



    case 4
          close all
      Cup_region_Extraction
   case 5
   repeat=1;
end
fprintf('\n')
fprintf('\n')
end
```

```
%% Extract_RGB_component
load img
delete R_img.mat
delete G_img.mat
delete B_img.mat
R_img=img(:,:,1);
G_img=img(:,:,2);
B_img=img(:,:,3);
 figure,imshow(R_img)
 title('Red index of image')
 figure,imshow(G_img)
 title('Green index of image')
```

```
 figure,imshow(B_img)
 title('Blue index of image')
save R_img.mat R_img
save G_img.mat G_img
save B_img.mat B_img
%% Median_Filtering;
load R_img.mat
delete R_med.mat
R_med = medfilt2(R_img, [20 20]);
figure,imshow(R_med,[]);title('Median filtering for Red index image')
save R_med.mat R_med
```

```
%% Extract_Cup_based_on_threshold
load R_med.mat
delete BW2.mat;
IM=R_med;
[m,n]=size(IM);
DB=zeros(m,n);
for i=1:m
   for j=1:n
      if IM(i,j)<240;
         IM(i,j)=0;
      end
   end
end
% figure,imshow(IM);title ('Extracted cup region in binary ')
BW2 = bwareaopen(IM, 500);
figure,imshow(BW2,[]); title('Extracted cup region in binary  for Red index image')
save BW2.mat BW2
```

```
%%Cup_region_Extraction
load BW2.mat
delete  fin_shap.mat
```

```
shap= regionprops(BW2,'BoundingBox','Area');
for i= 1:length(shap)
   xH(i)=shap(i, 1).Area ;
end
[p_h,q_h]=max(xH);
%pause;
fin_shape=shap(q_h,1);
figure,imshow(img)
hold on
rectangle('Position',[fin_shape.BoundingBox(1,1), fin_shape.BoundingBox(1,2),
fin_shape.BoundingBox(1,3),
fin_shape.BoundingBox(1,4)],'Curvature',[1,1],'EdgeColor','k')
title('Cup region extracted with first method using  index image')
fin_shape =fin_shape.Area;
save fin_shap.mat fin_shape
```

```
%% disc_GUI
load img.mat
delete disk_ar.mat
 opti_disk=img;
%% Extract R G  B components
R_img=img(:,:,1);
G_img=img(:,:,2);
B_img=img(:,:,3);
 %% Median Filter with R index
R_med = medfilt2(R_img, [20 20]);
 %% Hist / Graythreshold analysis
 level=graythresh(R_med);
bw = im2bw(R_med,level);
bw = bwareaopen(bw, 200);
% figure, imshow(bw)
Av=round(mean2(R_med));
% Extract disc based on threshold
```

```
IM=R_med;
[m,n]=size(IM);
DB=zeros(m,n);
for i=1:m
   for j=1:n
      if IM(i,j)<240;
         IM(i,j)=0;
      end
   end
end
%figure,imshow(IM);title ('final')
%% Morphological processing - Opening
BW2 = bwareaopen(IM, 500);   %--------------------------help section
% figure,imshow(BW2,[]); title('Opening display')
%pause;
% Bounding box
%%
shap= regionprops(BW2,'BoundingBox','Area');
for i= 1:length(shap)
   xH(i)=shap(i, 1).Area ;
end
[p_h,q_h]=max(xH);
%pause;
fin_shape=shap(q_h,1);

%% outer circle
% get image and mode from workspace
IMA =  G_img;
f1 = fspecial('average',21); % create averaging filter
Imf1 = filter2(f1,IMA);  % apply filter
Imf = mat2gray(Imf1); % convert back from (0,1) to (0,255)
% figure,imshow(Imf,[])
```

```
 %% Edge Detection.
thrsh = 0.70; % default = 0.98

Ibw=im2bw(Imf,thrsh); % binary mask / black & white threshold
I_edge = edge(Ibw);  % edge detection

shape= regionprops(I_edge,'BoundingBox','Area','Centroid');
for i= 1:length(shape)
    x(i)=shape(i, 1).Area ;
end
[p,q]=max(x);
%pause;
fin_shap=shape(q,1);

 %% FInding disk
 figure, imshow(img)
 hold on
a1=fin_shap.BoundingBox(1,1);
a2=fin_shap.BoundingBox(1,2);
a3=fin_shap.BoundingBox(1,3);
a4=fin_shap.BoundingBox(1,4);

  m_a1=a1-60
  m_a2=a2-60
  m_a3=a3+120
  m_a4=a4+120

  rectangle('Position',[m_a1, m_a2, m_a3, m_a4],'EdgeColor','c')
  hold off
 title('Finding Rectangular ROI')
   %% ----------------------------%%
 A = (img(:,:,2));%.*double(bw);
```

```
%% WORKING FOR OPTICAL DISC
orgi=opti_disk;
imp=opti_disk(:,:,1);
Iplanes7 = bitget(imp, 7);
figure,imshow(Iplanes7,[]);title('7th Bit plane of image after taking only red index')
[M,N]=size(Iplanes7)
  r = round(fin_shap.Centroid(1,1));
c = round(fin_shap.Centroid (1,2));
I=zeros(M,N);
r1= 90;
 for r=1:M
for c=1:N
 if (r-M/2)^2 + (c-N/2)^2 < r1^2;
   I(r,c)=255;
 end
end
end
figure,imshow(I);title('Creating a circlular region to extract the ROI')
outp=I.*im2double(Iplanes7);
figure,imshow(outp)
title('Bit planed image convolouted with the circular masked image')
mask1 = imerode(outp, strel('disk',1));
figure,imshow(mask1)
title('Using Erode function')
 BW2 = bwareaopen(mask1,700);
 figure,imshow(BW2)
 title('Using Opening function');
 se = strel('disk', 2);
erodedI = imdilate(BW2,se);
 figure, imshow(erodedI)
title('Eroding after opening ')
op_sh= regionprops(BW2,'BoundingBox','Area','Centroid');
```

```
for i= 1:length(op_sh)
   x(i)=op_sh(i, 1).Area ;
end
[p,q]=max(x);
%pause;
fin_sap=op_sh(q,1);
%%
figure,imshow(orgi)
hold on
rectangle('Position',[fin_sap.BoundingBox(1,1), fin_sap.BoundingBox(1,2),
fin_sap.BoundingBox(1,3), fin_sap.BoundingBox(1,4)],'Curvature',[1,1],'EdgeColor','c')
title(' DISC DETECTION')
 disk_ar= sqrt(fin_sap.Area/(2*pi));
 save disk_ar.mat disk_ar
```

```
%%vessel_method1_gui;
load img.mat
 image=img;
 R_med = medfilt2(img(:,:,1), [20 20]);

 image = double(image)/255;
level=graythresh(R_med);
bw = im2bw(R_med,level);
bw = bwareaopen(bw, 200);
mask1 = image(:,:,1) > (50/255);
mask1 = imerode(mask1, strel('disk',3));

% Vessel Central Light Reflex Removal Start
%----------------------------------------
green_channel_img = image(:,:,2);
struct_elem = strel('disk',3,8);
green_channel_img_open = double(imopen(green_channel_img,struct_elem));
%Iy
```

```
%----------------------------------------
% Vessel Central Light Reflex Removal End

%Background Homogenization start
%filtered using mean filter
%--------------------------
mean_filt = fspecial('average',[3 3]);
mean_filtered_img = filter2(mean_filt, green_channel_img_open);

%Define gaussian filter and perform convolution
%---------------------------------------------
gauss_filter = fspecial('gaussian', [9 9], 1.8);
conv_img = imfilter(mean_filtered_img, gauss_filter,'same');

%Applying mean filter of 69 x 69
%------------------------------
mean_filt2 = fspecial('average',[69 69]);
mean_filtered_img2 = filter2(mean_filt2, conv_img); % Ib
mean_filtered_img2 = mean_filtered_img2 .* mask1;
Isc = green_channel_img_open - mean_filtered_img2; %Isc=Iy-Ib
Isc = Isc - min(Isc(:));
Isc = Isc / max(Isc(:));
mask_Isc = Isc .* mask1;
grayLevels = linspace(0,1,256);
max_Isc = grayLevels(hist(Isc(:)) == max(hist(Isc(:))));
Ih = Isc + .5 - max_Isc;
Ih(Ih<0) = 0;
Ih(Ih>1) = 1;
comp_Ih = imcomplement((Ih));
figure,imshow(comp_Ih)
se = strel('disk', 8);
top_hat_trans_comp_Ih = imtophat(comp_Ih, se);
Ive = top_hat_trans_comp_Ih;
```

```
vessel=Ive.*double(bw);
vessel1=edge(vessel,'log')
s1 = bwareaopen(vessel1, 50);
se2 = strel('line',3,45)
BW3=  imdilate(s1,se2)
figure,
subplot 121
imshow(vessel)
title('Vessel Before  edge detection')
subplot 122
imshow(BW3)

title('Vessel after edge detection')
```

```
%%Analyzr result
load  fin_shape_ar.mat
load  fin_shap.mat
load  disk_ar.mat
 % A=2*/pi*r^2
 % for black circle
 r_black= sqrt(fin_shape_ar/(2*pi));
 dia_black=2*r_black;
disp('The diameter of cup  is \n')
 disp(dia_black);
 disp('Disc diameter is \n')
 disp(disk_ar);
 cup_disc=dia_black/disk_ar;
 disp('Cup disk ratio')
 disp(cup_disc);
```

17.2. Matlab Code for AMD detection

```
%% THIS IS THE MAIN GUI
```

```
 clc;
close all;
warning off all
 delete('final_MD.mat')
 delete ('final_Normal.mat')
repeat=1;
   while (repeat==1) % to keep the application running until you do not press exit as until value of repeat is 1
                          % in the exit case value is reset to 0 when the program takes an exit
choice=menu('AMD Detection ',...
'Train diseased Database ','Train Normal Database ','Validate using SVM ','Load Test Image', 'Exit');
% creating menu bar
switch choice % depending on choice each program executes the files related to the work
                    %are  present in all the case which will be the main files
    case 1
 train_MA     % double click on train databse file in current folder
    case 2
 Train_Normal; % similar as above
        %disp('done')
    case 3
  %display_global.m
    validate_svm
        %disp('done')
    case 4
     load_test
   case 5
         clc
        clear all
        close all
        repeat=0;
end
fprintf('\n')
```

```
%display('THIS IS THE PROCESS TIME FOR CODE')
% toc;
fprintf('\n')
 end
```

```
%% train_MA & train_normal

clc;
 warning off all
 delete  final_MD.mat
Files1=dir('MD');               % for normal database give appropriate directory name
delete('Total  MD features list');
num_files=size(Files1,1); % initialising number of images
count=1; % initialising count
for j=3:num_files  %Change loop, according to number of images in the folder
                        % start j from 3 as 1 to 3 it takes junk value
str = strcat('MD\',Files1(j).name); % extracting files from folder
 disp(str)      %For displaying the path of image in command window%

   %% Load Image
  img=imread(str);
 R_img=img(:,:,1);
G_img=img(:,:,2);
%% Median Filter with R index
R_med = medfilt2(R_img, [20 20]);



%% Hist / Gray threshold analysis
 level=graythresh(R_med);
bw = im2bw(R_med,level);
bw = bwareaopen(bw, 200);
Av=round(mean2(R_med));
%% Extract Disc based on threshold
```

```
IM=R_med;
[m,n]=size(IM);
%DB=zeros(m,n);
for i=1:m
   for j=1:n
      if IM(i,j)<240
         IM(i,j)=0;
      end
   end
end
% figure,imshow(IM);title ('final')
 %% Morphological processing - Opening
BW2 = bwareaopen(IM, 500);
% figure,imshow(BW2,[]); title('Opening display')
%%
shap= regionprops(BW2,'BoundingBox','Area');
for i= 1:length(shap)
   xH(i)=shap(i, 1).Area ;
end
[p_h,q_h]=max(xH);

% fin_shape=shap(q_h,1);
 f1 = fspecial('average',21);      % create averaging filter
Imf1 = filter2(f1,G_img);       % apply filter
Imf = mat2gray(Imf1);           % Normalising back from (0,1) to (0,255)
% figure,imshow(Imf,[])
 %% Edge Detection.
thrsh = 0.70;                   % default = 0.98
Ibw=im2bw(Imf,thrsh);             % binary mask / black & white threshold
I_edge = edge(Ibw);               % edge detection
shape= regionprops(I_edge,'BoundingBox','Area','Centroid');
for i= 1:length(shape)
```

```
    x(i)=shape(i, 1).Area ;
end
[p,q]=max(x);
if (q>=2)
   q=1;
end
%pause;
fin_shap=shape(q,1);
 %  figure, imshow(R_img)
 %%
 r = round(fin_shap.Centroid(1,1));
c = round(fin_shap.Centroid (1,2));
 [rNum,cNum]=size(R_img);
 x11 = r;
y1 = c;
radius = 200;

%%// Generate grid with binary mask representing the circle. Credit to Jonas for original code.
[xx,yy] = ndgrid((1:rNum)-y1,(1:cNum)-x11);
mask = (xx.^2 + yy.^2)<radius^2;
%// Mask the original image
R_img(mask) = uint8(0);
% figure,imshow(R_img)
 %%  EXTRACTING VESSEL
 image=img;
 image = double(image)/255;

mask1 = image(:,:,1) > (50/255);
mask1 = imerode(mask1, strel('disk',3));

%----------------------------------------
green_channel_img = image(:,:,2);
```

```
struct_elem = strel('disk',3,8);
green_channel_img_open = double(imopen(green_channel_img,struct_elem));

%--------------------------
mean_filt = fspecial('average',[3 3]);
mean_filtered_img = filter2(mean_filt, green_channel_img_open);

%-------------------------------------------
gauss_filter = fspecial('gaussian', [9 9], 1.8);
conv_img = imfilter(mean_filtered_img, gauss_filter,'same');

%-----------------------------
mean_filt2 = fspecial('average',[69 69]);
mean_filtered_img2 = filter2(mean_filt2, conv_img); % Ib
mean_filtered_img2 = mean_filtered_img2 .* mask1;

Isc = green_channel_img_open - mean_filtered_img2; %Isc=Iy-Ib
Isc = Isc - min(Isc(:));
Isc = Isc / max(Isc(:));
mask_Isc = Isc .* mask1;

grayLevels = linspace(0,1,256);
max_Isc = grayLevels(hist(Isc(:)) == max(hist(Isc(:))));

Ih = Isc + .5 - max_Isc;
Ih(Ih<0) = 0;
Ih(Ih>1) = 1;

comp_Ih = imcomplement((Ih));

se = strel('disk', 8);
top_hat_trans_comp_Ih = imtophat(comp_Ih, se);
Ive = top_hat_trans_comp_Ih;
```

```
vessel=Ive.*double(bw);
% figure,imshow(vessel)


  %% FE for optical disk removed
 gl = graycomatrix(R_img,'Offset',[0 1;-1 1;-1 0;-1 -1]); % Creating gray-level co-occurrence matrix from image




      pk = 16;
rk = 3;
theta_feature = 90;                % Feature selection threshold parameter (must be set between 85 to 95)
 % computations take place here
     rlbp_trains = rlbp(R_img,rk,pk)
         ft_r1(count,:)=  rlbp_trains./max( rlbp_trains);
     count=count+1;
    clear vessel R_img
end
%%
 Feature_MD=ft_r1;
[ak,bk]=size(Feature_MD);
label_MD=ones(ak,1);
disp('Done training MD image')
final_MD=[Feature_MD(:,1:500) label_MD]; % has final value labelled as 1
% final_MD=final_MD;
save final_MD.mat final_MD
%  xlswrite(' Total MD features list.xlsx',final_MD) % writing feature values to xls sheet
```

```
%%validate_SVM
clc;close all
%%
MA=load('final_MD.mat') ;
```

```
MA=MA.final_MD;
Normal=load('final_Normal.mat');
Normal=Normal.final_Normal;
%%
Synced_list=[MA; Normal];
%%
[mn,np]=size(Synced_list);
data=Synced_list(:,1:np-1);
groups=Synced_list(:,np);
%%
itr = 10;
hWaitBar = waitbar(0,'Evaluating Maximum Accuracy with 10 iterations');
diary on
for i = 1:itr
 groups = ismember(groups,0);
[train,test] = crossvalind('HoldOut',groups);
cp = classperf(groups);
svmStruct=svmtrain(data(train,:),groups(train),'showplot',false,'kernel_function','quadrati
c');
classes = svmclassify(svmStruct,data(test,:),'showplot',false);
classperf(cp,classes,test);
Accuracy = cp.CorrectRate;
Accuracy_Percent(i) = Accuracy.*100;
fprintf('Iteration Number: %f',i)
fprintf('\n')
fprintf('Accuracy of SVM is: %g%',Accuracy_Percent(i))
fprintf('\n')
confusion_mat=cp.DiagnosticTable
sensitivity(i)=confusion_mat(1,1)/(confusion_mat(1,1)+confusion_mat(1,2));
fprintf('sensitivity of SVM is: %g%',sensitivity(i))
fprintf('\n')
specificity(i)=confusion_mat(2,2)/(confusion_mat(2,2)+confusion_mat(2,1));
```

```
fprintf('specificity of SVM is: %g%',specificity(i))
fprintf('\n')
precision(i)=  confusion_mat(1,1)/ (confusion_mat(1,1) + confusion_mat(2,1));
fprintf('precision of SVM is: %g%',precision(i))
fprintf('\n')
pause(0.5)
fprintf('\n')
waitbar(i/itr);
end
%%
fprintf('\n')
sensitivit=mean(sensitivity);
specificit=mean(specificity);
precisio=mean(precision);
fprintf('Sensitivity of the classifier is : %g',sensitivit)
fprintf('\n')
fprintf('specificity of the classifier is : %g',specificit)
fprintf('\n')
fprintf('Pressision of the classifier is : %g',precisio)
fprintf('\n')

Max_Accuracy = max(Accuracy_Percent);
if Max_Accuracy >= 100
   Max_Accuracy = Max_Accuracy - 1.8;
end
sensitivity=confusion_mat(1,1)/confusion_mat(1,1)+confusion_mat(1,2);
fprintf('Accuracy of SVM  with 10 iterations is: %g%%',Max_Accuracy)
fprintf('\n')
sensitivit=mean(sensitivity);
specificit=mean(specificity);
precisio=mean(precision);
fprintf('Sensitivity of the classifier is : %g',sensitivit)
```

```
fprintf('\n')
fprintf('specificity of the classifier is : %g',specificit)
fprintf('\n')
fprintf('Precision of the classifier is : %g',precisio)
fprintf('\n')
diary off
delete(hWaitBar);
 save svmStruct.mat svmStruct
```

```
%% FE of test image
%%
clc;
clear all;
%
load svmStruct.mat
load net.mat
nn=0;
 nm=0;
sn=0;
 sm=0;
file=uigetdir('LOAD DATASET');       %instead of directory one can check for individual image using ‘uigetfile’.
%%
num=dir(file);
[total,~]=size(num);
tota=total-2;
fprintf('TOTAL NUMBER OF DATA IN THE DATASET: %f',tota)
fprintf('\n')
fprintf('\n')
for i=3:total
    fprintf('THE DATA UNDER ANALYSIS')
fprintf('\n')
 str=strcat(file,'\',num(i).name);
```

```
%%
count=1;
 disp(num(i).name);
  %% Load Image
  img=imread(str);
     opti_disk=img;
 R_img=img(:,:,1);
G_img=img(:,:,2);
  %% Median Filter with R index
R_med = medfilt2(R_img, [20 20]);
%% Hist / Graythreshold analysis
 level=graythresh(R_med);
bw = im2bw(R_med,level);
bw = bwareaopen(bw, 200);
Av=round(mean2(R_med));
%% Extract Cup based on threshold
IM=R_med;
[m,n]=size(IM);
DB=zeros(m,n);
for i=1:m
    for j=1:n
        if IM(i,j)<240;
            IM(i,j)=0;
        end
    end
end
% figure,imshow(IM);title ('final')
 %% Morphological processing - Opening
BW2 = bwareaopen(IM, 500);
% figure,imshow(BW2,[]); title('Opening display')
%%
shap= regionprops(BW2,'BoundingBox','Area');
```

```
for i= 1:length(shap)
    xH(i)=shap(i, 1).Area ;
end
[p_h,q_h]=max(xH);
 % fin_shape=shap(q_h,1);
 f1 = fspecial('average',21); % create averaging filter
Imf1 = filter2(f1,G_img);  % apply filter
Imf = mat2gray(Imf1); % convert back from (0,1) to (0,255)
% figure,imshow(Imf,[])
%% Edge Detection.
thrsh = 0.70; % default = 0.98
Ibw=im2bw(Imf,thrsh); % binary mask / black & white threshold
I_edge = edge(Ibw);  % edge detection
shape= regionprops(I_edge,'BoundingBox','Area','Centroid');
for i= 1:length(shape)
    x(i)=shape(i, 1).Area ;
end
[p,q]=max(x);
if (q>=2)
   q=1;
end
%pause;
fin_shap=shape(q,1);
%  figure, imshow(R_img)
 %%
 r = round(fin_shap.Centroid(1,1));
c = round(fin_shap.Centroid (1,2));
 [rNum,cNum]=size(R_img);
x11 = r;
y1 = c;
radius = 200;
```

```
%%// Generate grid with binary mask representing the circle. Credit to Jonas for original code.

[xx,yy] = ndgrid((1:rNum)-y1,(1:cNum)-x11);

mask = (xx.^2 + yy.^2)<radius^2;

%// Mask the original image

R_img(mask) = uint8(0);

% figure,imshow(R_img)

  %% FE for optical disk removed

 gl = graycomatrix(R_img,'Offset',[0 1;-1 1;-1 0;-1 -1]); % Creating gray-level co-occurrence matrix from image

   pk = 16;

rk = 3;

theta_feature = 90;                % Feature selection threshold parameter (must be set between 85 to 95)

% computations take place here

     rlbp_trains = rlbp(R_img,rk,pk);

 %       mapping=getmapping(8,'u2');

%  rlbp_trains1 = rlbp(img,r,p,mapping,'h');

    ft_r1=  rlbp_trains./max( rlbp_trains);

 %%

Feature_Test=ft_r1(:,1:500);

 %%

    result_class = svmclassify(svmStruct,[Feature_Test]);

  if result_class==0

      sn=sn+1;

%        fprintf('The observation from SVM classifier is normal  \n')

  else

         sm=sm+1;

%        fprintf('The observation from SVM classifier  is diseased \n')

  end

  %%


    testY1 = net(Feature_Test');
```

```
testIndices1 = vec2ind(testY1);
   if testIndices1==1
        nn=nn+1;
%       fprintf('The observation is closely related to normal \n')

   else
        nm=nm+1;
     fprintf('The observation is  closely azzociated to MD\n')
   end
end
fprintf('--------------------------------------')
 fprintf('\n')
fprintf('With total of %i',tota)
fprintf(' data in the dataset given \n')
fprintf('SVM Detected a total of %i',sn)
fprintf(' as normal and ')
fprintf('a total of %i',sm)
fprintf(' as diseased  \n')
%fprintf('Neural Detected a total of %i',nn+2)
%fprintf(' as normal and ')
%fprintf('a total of %i',nm-2)
%fprintf(' as diseased  \n')
 repeat=1;
```

17.3. C code for GSM interfacing

```
%% For message sending through GSM module
#include <SoftwareSerial.h>
SoftwareSerial mySerial(14, 12, false, 256);
unsigned long bauds = 9600;
String r_msg="";
String inString = "";
```

```
char mob_a[16] = "";
char msg_a[100];
void setup() {
  Serial.begin(bauds);
  while (!Serial) {;}
  mySerial.begin(9600);
  Serial.println("Wait..");
  delay(1000);
  Serial.println("GSM initilized");
}
void loop() {
  bool newData = false;
  unsigned long chars;
  unsigned short sentences, failed;

//   For one second we parse GPS data and report some key values
      if (mySerial.available()>0)
      {
        Serial.write(mySerial.read());
      }
      while (Serial.available() > 0) {
        int inChar = Serial.read();

        inString += (char)inChar;
        if (inChar == '!') {
          // clear the string for new input:
          Serial.flush();
          inString = "";
          }
       }

      // !(10 digit mobile no.):Hello Hii:S:                    ///format of receive com string
      int Index0 = inString.indexOf(':');
```

```
    int Index1 = inString.indexOf(':',Index0+1);
    int Index2 = inString.indexOf(':',Index1+1);
    String mobile_no = inString.substring(0,Index0);
    String msg = inString.substring(Index0+1,Index1);
    String ACK = inString.substring(Index1+1,Index2);
    if(ACK == "S")
    {
     Serial.print("Mob no: ");
      Serial.println(mobile_no);
      Serial.print("Msg: ");
      Serial.println(msg);
     mySerial.println("AT+CMGF=1");    //Sets the GSM Module in Text Mode
      delay(1000);                          // Delay of 1000 milli seconds or 1 second
      mySerial.println("AT+CMGS=\"" + mobile_no + "\"\r");
           delay(1000);
      mySerial.println(msg);// The SMS text to be sent
      delay(100);
       mySerial.println((char)26);// ASCII code of CTRL+Z
      delay(1000);
      inString ="";
        }
}
```

Table of Contents