CS365:Artificial Intelligence

Course Project

Diabetic Retinopathy Detection Using Eye Images

Mohit Singh Solanki
12419
mohitss@iitk.ac.in

Supervisor:
Dr. Amitabha Mukherjee

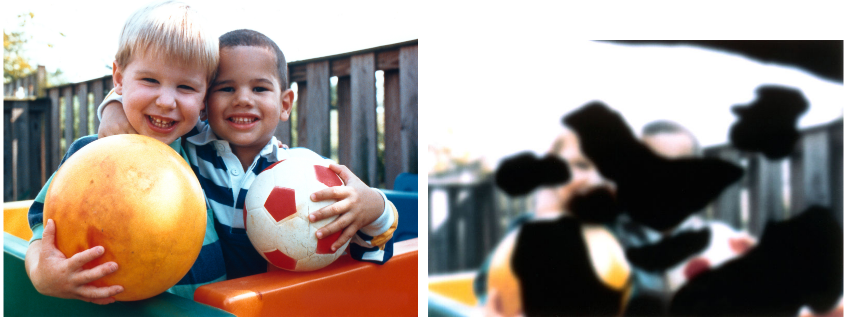
April 18, 2015

Abstract

Diabetic Retinopathy is one of the leading causes of blindness and eye disease in working age population of developed world. This project is a attempt towards finding a automated way to detect this disease in its early phase. In this project I am using supervised learning methods to classify a given set of images into 5 classes. For this task I am employing various image processing techniques and filters to enhance many important features and then using neural for classification.

1 Introduction

Diabetic Retinopathy is a disease which is caused due to long term diabetes. It is a ocular manifestation of diabetes and around 80 percent of population having diabetes for more than 10 or more years has some stage of the disease. Also the longer a person is in this disease there higher are the chances of of having DR in his visual system. Researches shows that it contributes around 5% of total cases of blindness. According to 'WHO' estimation 347 million of world population is having the disease diabetes and about 40-45% of them have some stage of the disease. By seeing below image one can see difference between image produced by normal eye and DR eye.[4]



(a) Normal Eye[4]

(b) DR affected eye[4]

There are various factors affecting the disease like age of diabetes, poor control, pregnancy but Researches shows that progression to vision impairment can be slowed or averted if DR is detected in early stage of the disease. One can see large no. of population suffering from the disease but still testing is done manually by trained professionals in real life which is quite time taking and lengthy process and usually due to miscommunication and delayed results eventually leads to delayed treatment and ignorance.

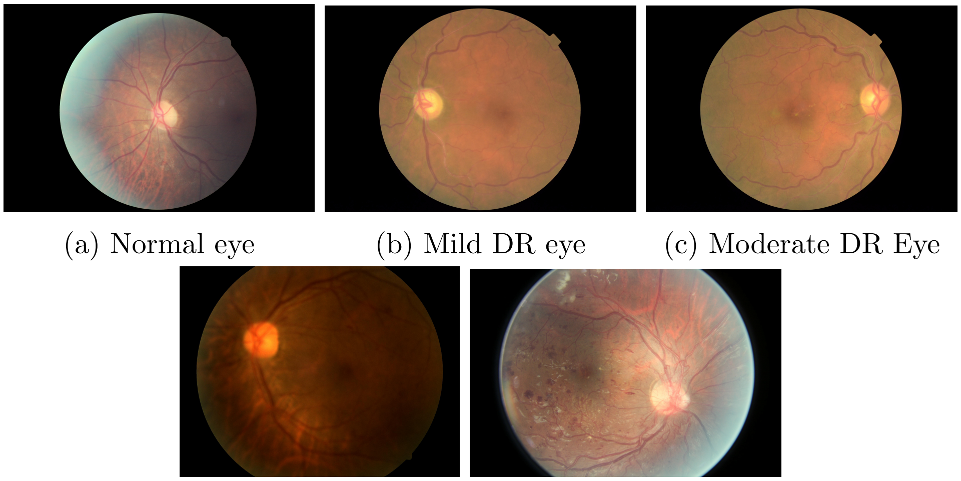
So aim of the project is to provide a a automated, suitable and sophisticated approach using image processing and pattern recognition so that DR can be detected at early levels easily and damage to retina can be minimized.

2 Dataset

This is an ongoing problem on kaggle[1] which tries to develop a model for DR detection. Dataset is taken from the challenge-data part. Data set consists of high resolution eye images and graded by trained professionals in 5 classes(0-4) which is according to below table and figure below that.

Class name	Meaning
Class 0	Normal Eye
Class 1	Mild DR eye
Class 2	Moderate DR Eye
Class 3	Severe DR eye
Class 4	Poliferative DR Eye

Table 1: Class name descriptions



(a) Normal eye (b) Mild DR eye (c) Moderate DR Eye

(d) Severe DR eye (e) Poliferative DR Eye

3 Methodology

Method used in this project can be classified in two steps

- Image Processing and Feature Extraction
- Supervised learning

3.1 Image Processing and Feature Extraction

This is the most important step of the project as textures obtained will be taken as input material for neural nets which will classify the images in their respective classes.

3.1.1 Image compression

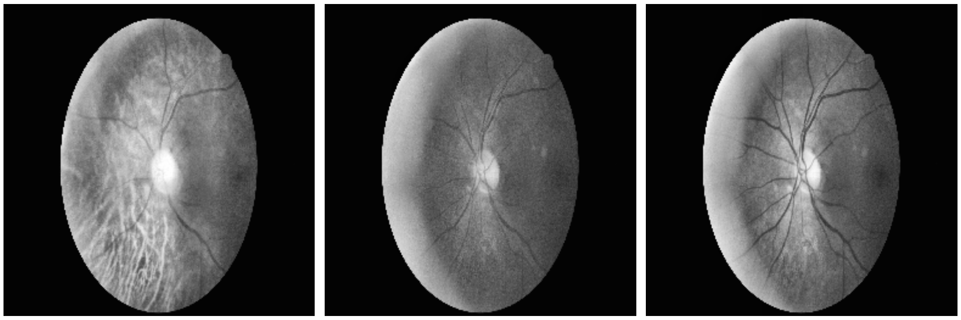
As one can see there are different types of images in dataset with different resolution, different camera quality and different sizes My work is to classify them in different classes. So first problem I faced was related to heterogeneity of the dataset. For this compressed all my training and test images in 256*256 format.

3.1.2 Layer separation

In later parts we are going to use 6 features as input to classifier namely Red layer of parameter,Blue layer of parameter, Green layer of parameter, Red layer of area, Green layer of area, Blue layer of area so in this step all 3 layers of namely Red, Green and Blue are separated from the images.

3.1.3 Equalization

After last step there are large intensity variations in the image and one can see that veins and other eye features are not clearly seen there. For making intensity variations uniform I applied histogram equalization to the image. Histogram equalization is technique which identifies various intensity variations in the given image and increases its global contrast.For equalization I tried both Histogram Equalization and Contrast Limited Adaptive Histogram Equalization but Contrast Limited Adaptive Histogram Equalization giving a little better features than simple one. So in this step I have used CLAHE object for equalization purpose.[2, 3, 6, 7]



(a) Red Layer (b) Blue Layer (c) Green layer

3.1.4 Morphological operations

In this part various morphological operations are employed to enhance blood vessels and to remove noise in the background. I used method proposed in (use cite here) to enhance to required features. Blood vessel rupture are main element of the disease DR. So it is important to extract and distinguish them from the background and remove background noise as much as possible. Two types of structuring elements are used in this step.[2, 8]

- Diamond like structure(for clearer veins)
- Disk like structure(to remove noise)

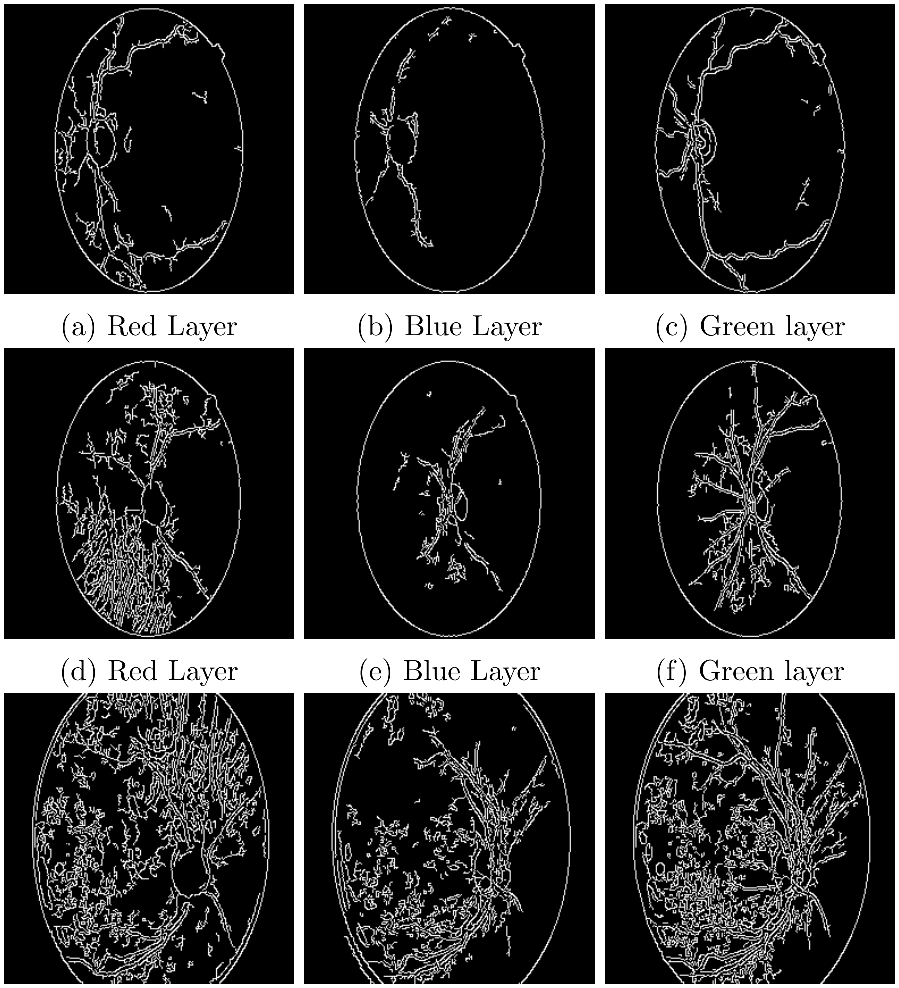
For this part I have used morphological openings. In this part I first used disc SE with R=5 then I used diamond of R=3.

3.1.5 Feature extraction

This is final image processing step for the project. In this step I will first extract perimeter from all three layers and then extract area of three layers.

- Canny edge detection

In this step we proceed towards finding perimeters of all 3 layers. This is done by canny edge detection. In canny edge detection gaussian filters are applied then using double threshold edge of intensity variation part is detected.[2, 3, 9]



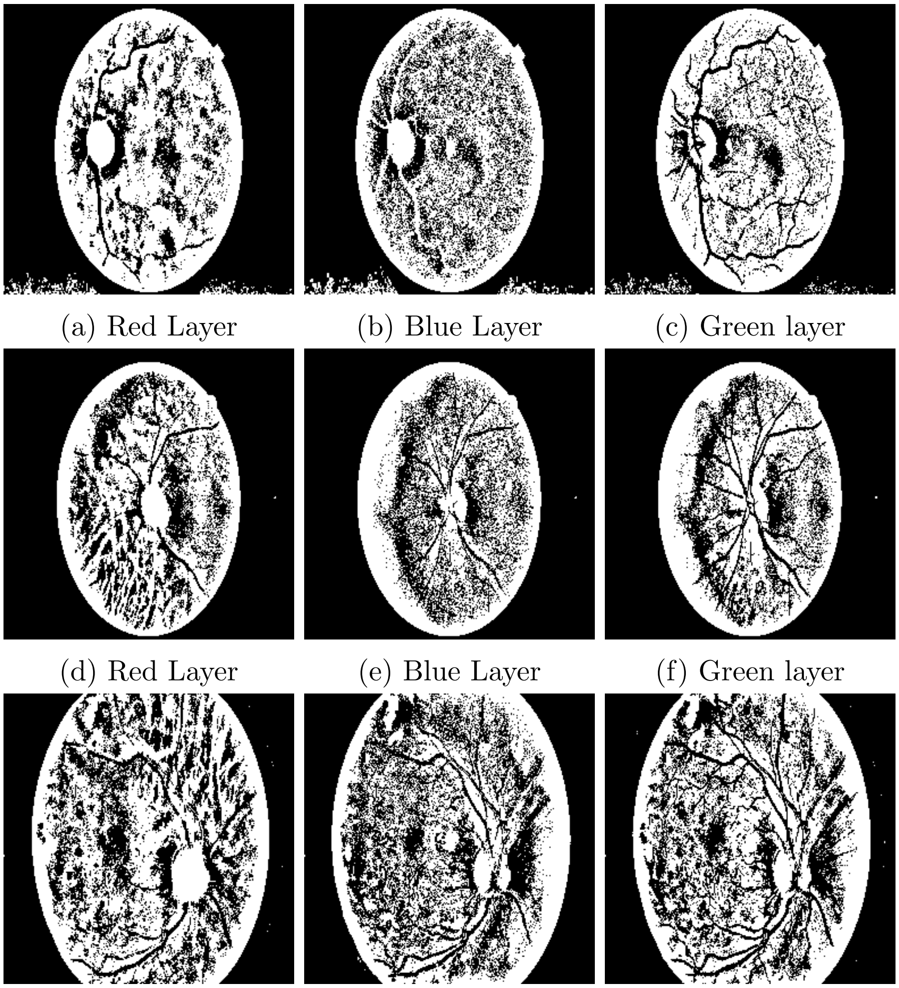
(a) Red Layer (b) Blue Layer (c) Green layer

(d) Red Layer (e) Blue Layer (f) Green layer

(g) Red Layer (h) Blue Layer (i) Green layer

- Thresholding

This step is applied on morphed images which gives area of the 3 layers. This is done by adaptive thresholding. I have also tried using otsu's thresholding and simple thresholding but later is giving better areas then other two.[2, 3, 10]



(a) Red Layer (b) Blue Layer (c) Green layer

(d) Red Layer (e) Blue Layer (f) Green layer

(g) Red Layer (h) Blue Layer (i) Green layer

	class 0	**class 1**	**class 2**	**class 3**	**class 4**
class 0	170	53	20	14	0
class 1	41	69	14	6	1
class 2	18	26	25	4	0
Class 4	3	8	5	8	1
Class 5	0	2	6	2	4
	232	158	70	34	6

Table 2: Confusion Matrix

3.1.6 Classification

This is the last part of the whole process. Here I have used deep neural nets with 3 convolution layer. In this part first balanced classes are created by using above images and then they are given as input in neural nets along with labels. This on output gives predicted labels. In this step I have used deep neural net library provided by Graphlab and used github code(http://bit.ly/1Eefvvd)[5] as base.

4 Results

By using 500 images as training input and same number of images for test we got these results on classification

In this way we got descent results for our analysis. By table-2 we can see that it gives around 55%accuracy in the results. The code for neural networks took around 6-7 hours to run over the images.

5 Future Work

In future in order to improve efficiency and better resultsI would like to employ below modifications in the project

- Increase size of test and training data set
- To use better morphological analysis algorithms to get clearer features.

- Implement neural nets in a better and efficient way
- In place of direct image features may try to use gradient of RGB layers

References

[1] kaggle Challenge *Diabetic Retinopathy Detection.* https://www.kaggle.com/c/diabetic-retinopathy-detection

[2] Wong Li Yun , U. Rajendra Acharya, Y.V. Venkatesh, Caroline Chee, Lim Choo Min, E.Y.K. Ng *Identification of different stages of diabetic retinopathy using retinal optical images.* July 2007

[3] Jagadis h Naya k, P Subba nna Bhat, Rajen dra Achar ya U,C. M. Lim, Manjunath Kagathi
Automated Identification of Diabetic Retinopathy Stages Using Digital Fundus Images November 2007

[4] Diabetic Retinopathy http://en.wikipedia.org/wiki/Diabetic_retinopathy

[5] Starter code Startercode,http://bit.ly/1Eefvvd

[6] Equalization http://opencv-python-tutroals.readthedocs.org/en/latest/py_tutorials/py_imgproc/py_histograms/py_histogram_equalization/py_histogram_equalization.html

[7] Equalization http://en.wikipedia.org/wiki/Histogram_equalization

[8] Morphological operations http://opencv-python-tutroals.readthedocs.org/en/latest/py_tutorials/py_imgproc/py_morphological_ops/py_morphological_ops.html#morphological-ops

[9] Canny edge detection http://opencv-python-tutroals.readthedocs.org/en/latest/py_tutorials/py_imgproc/py_canny/py_canny.html

[10] Thresholding http://opencvpython.blogspot.in/2013/05/thresholding.html

[11] R.ManjulaSri , M.Raghupathy Reddy , and K.M.M.Rao *Image Processing for Identifying Different Stages of Diabetic Retinopathy.* July 2007