Severity Detection of Diabetic Retinopathy

Ranjana Kasangeri(170050086), Ujjval Goury(170050108), Mayanka Medhe(160050027)

24 November 2019

Abstract

Diabetic retinopathy is a medical condition in which damage occurs to the retina due to diabetes mellitus. It affects up to 80 percent of those who have had diabetes for 20 years or more. At least 90% of new cases could be reduced with proper treatment and monitoring of the eyes. The longer a person has diabetes, the higher his or her chances of developing diabetic retinopathy. Each year in the United States, diabetic retinopathy accounts for 12% of all new cases of blindness. It is also the leading cause of blindness in people aged 20 to 64. In this project, we are proposing a machine learning architecture to detect severity of diabetic retinopathy. We are comparing the Multinomial Logistic Regression, Support vector machines, Random forests, voting classifier and Convolutional Neural Network for their accuracy.

Key words- Classification, Multinomial Logistic Regression, Support vector machines, Random forests, voting classifier, Convolutional Neural Network

1 Introduction

Diabetic retinopathy can be detected and analyzed from the retinal fundus images. These images are used to determine the stage of the disease. Because fundoscopic images are the main sources for diagnosis of diabetic retinopathy, manually analyzing those images can be time-consuming and unreliable, as the ability of detecting abnormalities varies by years of experience. We are proposing a method of extracting useful features from these images and using various machine learning model to classify the images on the following scale: No DR (0), mild (1), moderate (2) and severe (3).

2 Dataset

We have used the dataset from MESSIDOR [1] which stands for Methods to Evaluate Segmentation and Indexing Techniques in the field of Retinal Ophthalmology (in French). It contains images belonging to the 4 classes of severity- No DR (0), mild (1), moderate (2) and severe (3) (Refer to figures 1-4)

3 Methodology

Our model outputs if diabetic retinopathy is present, along with its level of severity. We are first preprocessing the images to get the required format. Then we are extracting features from them using various feature extraction techniques. These features are then used to train the model. Each step is explained in details below-

3.1 Preprocessing

We are first preprocessing the images by cropping and resizing them to 256*256 size. Then we are applying various filtration and enhancement techniques like Contrast-Limited Adaptive Histogram Equalization (CLAHE), Extraction of Green channel and subtraction of median filtered image. The image after every filter is shown on in figure 5-8.

3.2 Feature extraction

These preprocessed images are then used to extract useful features from them using least squares method namely Histogram of Oriented Gradients (HOG) and Oriented FAST and rotated BRIEF(ORB). The HOG technique counts the occurrences of gradient orientation in localized portions of an image. ORB is based on the FAST keypoint detector and the visual descriptor BRIEF (Binary Robust Independent Elementary Features). The feature histogram image of fig 4 is shown in the figure 9-10-

Figure 1: No DR

Figure 2: Mild DR

Figure 3: Moderate

Figure 4: Severe DR

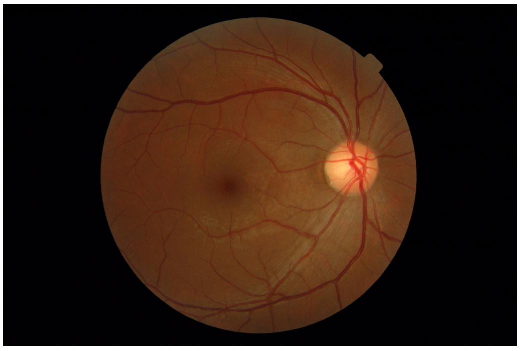

Figure 5: Original image (1488, 2240, 3)

Figure 6: Cropped and resized (256, 256, 3)

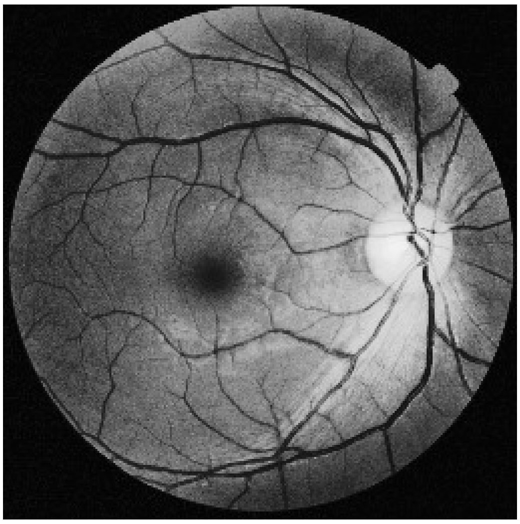

Figure 7: CLAHE and Green channel extraction

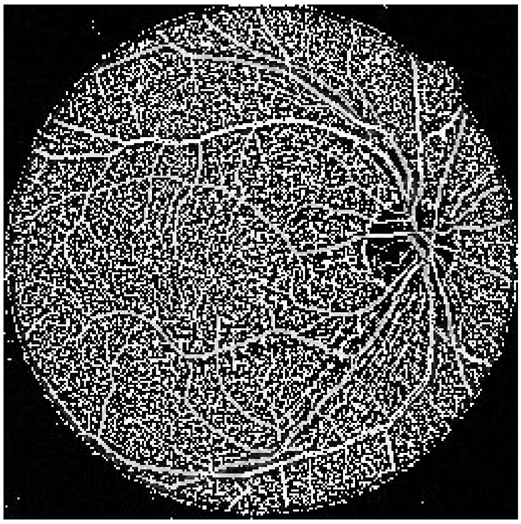

Figure 8: Subtracted median filtered image

3.3 Classification

These features are used to train the models. We have trained the features on the following models- Multinomial logistic regression, Support vector machines(SVMs), Random Forests, Voting Classifier and Convolutional neural network.

3.3.1 Multinomial logistic regression

Multinomial logistic regression is a simple extension of binary logistic regression that allows for more than two categories of the dependent or outcome variable. We have implemented it using the LogisticRegression model from sklearn package in python.

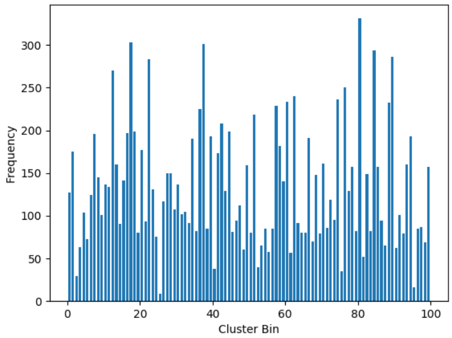

Figure 9: Feature Histogram for HOG

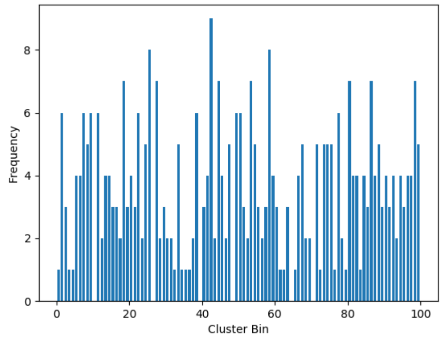

Figure 10: Feature Histogram for ORB

3.3.2 Support vector machine

Support-vector machines (SVMs) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. We have implemented it using the SVM model from sklearn package in python.

3.3.3 Random Forests

Random forests are an ensemble learning method for classification that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) of the individual trees.

3.3.4 Voting Classifier

Finally, we combine the classifiers above with a Voting Classifier, taking hard voting i.e selecting the class with maximum votes. The accuracy obtained is better than each of the accuracy obtained before.

3.3.5 Convolutional neural network

Convolutional networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers. Model used ResNet50 from keras library.

4 Results and discussion

We used 600 images containing all the categories(0-4). We used 30% of images for testing and 70% on training. We used both HOG and ORB features to determine our results. HOG gave better results than ORB hence we chose that as for our subsequent experimentation. The test and train accuracy of all the models with HOG features are shown in table 1. Table 2 gives the confusion matrix with HOG features. We observed that on increasing the number of clusters while detecting features the accuracy increased but at the cost of computation time.

Table 1: Train and test accuracy with HOG

Model	**Train Accuracy (%)**	**Test Accuracy (%)**
SVM	65	60.20
MLR	61.67	59.4
RF	98.6	65.8
Voting	**99.44**	**63.3**
CNN	56	51.4

5 Conclusion and future work

We have observed that Voting (SVM + RF + MLR) perform better than all the other models in all cases. The next closest to it was Random forest. Since voting takes votes from all its models it was expected to give more accuracy than any other model alone. We propose to improve the model further by using deep neural networks. We also believe that using a larger dataset will improve its accuracy.

6 Acknowledgement

We would like to thank Prof. Ganesh Ramakrishnan for giving us the opportunity to work on this project. We would also like to thank all the TAs for their support and guidance.

References

[1] Messidor dataset - "http://www.adcis.net/en/third-party/messidor/," (Accessed on 23 Nov, 2019).

[2] "https://medium.com/@burhanahmeed/image-recognition-with-svm-and-local-binary-pattern-289cc19ba7fe".

[3] "https://opencv-python-tutroals.readthedocs.io/en/latest/".

[4] "http://faroit.com/keras-docs/1.2.0/".

[5] "http://cs229.stanford.edu/proj2016/report/HonnungarMehraJoseph-DRISC-report.pdf".