JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Efficient and Early Diabetic Retinopathy Detection

[1]Suraj Gupta, [2]Dr. Sneha Soni
[1]Mtech Scholar,SIRTE, Bhopal, India
[2] Asso. Prof. & HOD, SIRTE, Bhopal, India

Abstract : Diabetic retinopathy (DR) is a complex issue affecting diabetic patients, leading to retinal damage and potential blindness. This condition disrupts the blood vessels in the retina, causing fluid leakage and severe vision distortion. DR is a prevalent eye disease associated with chronic diabetes and is the leading cause of blindness among working-age adults worldwide, potentially affecting over 93 million individuals. This research introduces an automated classification system capable of analyzing fundus images with varying illumination and fields of view. It employs machine learning models, including Otsu, Random Forest, and Clehe algorithm, to generate severity grades for diabetic retinopathy (DR).

The use of machine learning models, such as Random Forest, offers the advantage of high variance and low bias, enabling the classifier to potentially diagnose a broader range of nondiabetic diseases as well. Visualizations of the features learned by the Random Forest classifier and GLMC (Gray Level Matrix Co-occurrence) reveal that the signals used for classification are predominantly located in clearly observable parts of the image. Moderate and severe diabetic retinal images exhibit macroscopic features at a scale that aligns with the architecture's training accuracy and validation accuracy. This research presents a promising approach to automated DR severity classification, offering potential benefits for early diagnosis and intervention in diabetic patients' eye health.

***IndexTerms* - Diabetic Retinopathy, Classification, Image Processing, Deep Learning, Segmentation, Severity Grade**

I. INTRODUCTION

The field of medical image analysis has witnessed significant advancements in recent years, driven by the integration of machine learning techniques. Among the various medical conditions that benefit from these innovations, diabetic retinopathy (DR) stands out as a critical concern. DR is a diabetes-related eye disease that damages the retina's blood vessels, potentially leading to blindness if not diagnosed and treated promptly. As the leading cause of blindness among working-age adults worldwide, DR imposes a substantial global healthcare burden. The key challenge in managing DR effectively is the timely detection and classification of its severity levels. Early diagnosis and intervention are crucial for preventing vision loss in diabetic patients. This research delves into the development of an automated classification system for DR severity assessment using deep neural networks, Random Forest, and the Clehe algorithm. We aim to leverage the power of machine learning to analyze fundus images, accounting for variations in illumination and fields of view. Our objective is to provide an accurate and efficient tool that can assist healthcare professionals in diagnosing and managing DR effectively.

In this study, we explore the capabilities of deep neural networks, Random Forest, and the Clehe algorithm in classifying DR severity levels. We also investigate the interpretability of these models by visualizing the features they learn from fundus images. The insights gained from this research could have far-reaching implications for early DR detection and the broader field of medical image analysis.

We recognize that the challenges posed by diabetic retinopathy extend beyond just medical diagnosis. The societal and economic impacts of this condition are profound, affecting millions of individuals and straining healthcare resources. Hence, our research seeks to address these broader issues by proposing an automated system that could potentially revolutionize the management of DR.

The primary objectives of our study are as follows:

Automated DR Severity Classification: We aim to create a robust system that can automatically classify the severity of diabetic retinopathy in fundus images. This would enable healthcare providers to assess the condition quickly and accurately.

Machine Learning Models: We explore the capabilities of deep neural networks, Random Forest, and the Clehe algorithm as potential tools for DR severity classification. By comparing and contrasting these models, we seek to determine the most effective approach.

Interpretability and Visualization: In addition to achieving high classification accuracy, we investigate the interpretability of these models. Visualization techniques will help us understand which features are crucial for classification, providing insights into how the models arrive at their decisions.

DR is a complication caused by diabetes due to damage of the bloodvessels of the light-sensitive tissue at the back of the eye [6]. Toomuch sugar in the blood leads to blockage of the tiny blood vessels,cutting off its blood supply and the eye attempts to grow new bloodvessels. Early DR is called non-proliferative diabetic retinopathywhere new blood vessels are not growing yet, but the walls in theblood vessels weakens resulting in tiny bulges protruding fromvessel walls and sometimes exudates due to leakages of fluid andblood into the retina. Larger vessels begin to dilate and results inirregular diameter. As the condition progresses, the blood vesselgets blocked and retina swells. This results in growth of abnormalnew blood vessels in the retina. These new vessels can result inleakage and scar tissue formation in the eye, which can finally leadto retinal detachment and glaucoma. Figure 1 shows an image withvarious types of diabetic retinopathy conditions. Hence, detectionof such exudates, scars and abnormal blood vessels are importantdiagnostic tasks in DR. Typical manual diagnosis takes 7 to 14 daysas shown in Fig. 2 and needs specialized professionals input. Thepurpose of this study is to predict DR and do automated analysisby assigning a score of severity based on high-resolution retinaimages. This will save time in manual diagnosis as well as providesupport where specialists are not available.

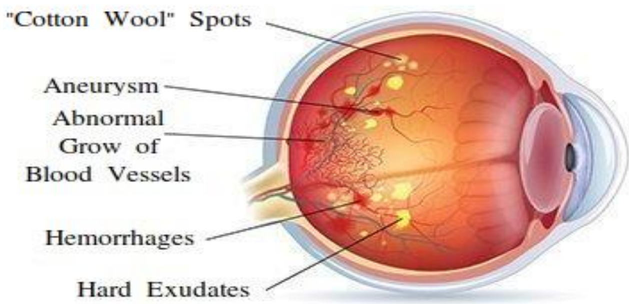

Figure 1.: Diabetic retinopathy conditions

II. PROPOSED METHODOLOGY

The Proposed system is a multistage classifier of Diabetic Retinopathy. This system overcomes the drawbacks of the existing system by classifying the disease in to four stages, namely, Normal, NPDR 1, NPDR 2 and PDR. This multistage classification is important because the disease itself progresses in multiple stages. The reoccurrence of the disease depends on the stage in which the treatment is provide, so it is not enough to classify the image as normal and abnormal. The pre-processing part of both the existing and proposed system remain similar, the difference comes in the segmentation and feature extraction stages. Existing system only segmented anomalies like microaneurysms, the problem with this is this anomaly occurs in the initial stage of the disease. Treatment cannot be given at this stage, hence this is a major drawback. The proposed system overcomes this by segmenting haemorrhages along with microaneurysms and also by considering a large feature set which includes the area and count of the segmented anomalies. The feature set also includes textural features like energy and correlation, and statistical features like mean and variance. This feature set is then used to classify the image into the respective severity.

DATA FLOW DIAGRAM

A data flow diagram (DFD) illustrates how data is processed by a system in terms of inputs and outputs. As its name indicates its focus is on the flow of information, where data comes from, where it goes and how it gets stored.

Applications of DFD

DFDs are a common way of modeling data flow for software development. For example, a DFD for a word-processing program might show the way the software processes the data that the user enters by pressing keys on the keyboard to produce the letters on the screen.

Significance of DFD

DFDs are popular for software design because the diagram makes the flow of data easy to understand and analyze. DFDs represent the functions a system performs hierarchically, starting with the highest-level functions and moving through various layers or levels of sub-functions. As a modeling technique, DFDs are useful for performing a structured analysis of software problems, allowing developers to spot and pinpoint issues in software development.

Every system is developed either to satisfy a need or to overcome the drawbacks of an existing system.

Level 0

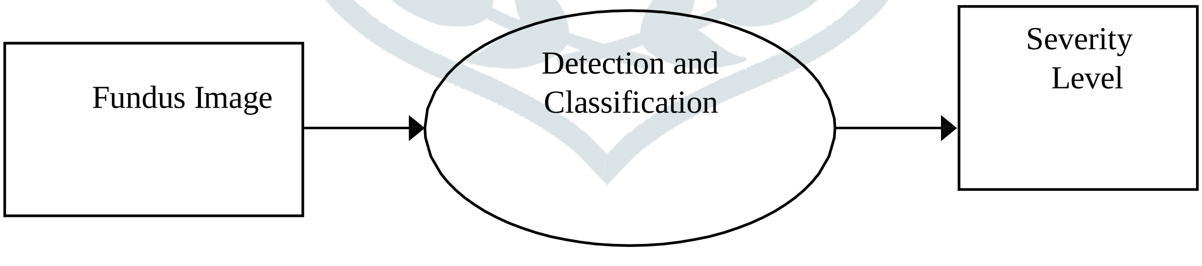

Figure 2 Level 0 DFD

Figure 3.1 shows the level 0 DFD. The fundus image of the eye is given as the input, the disease is detected and classified to give the severity level of the disease i.e. grades 0,1,2,3 corresponding to normal, mild, moderate and sever.

Level 1

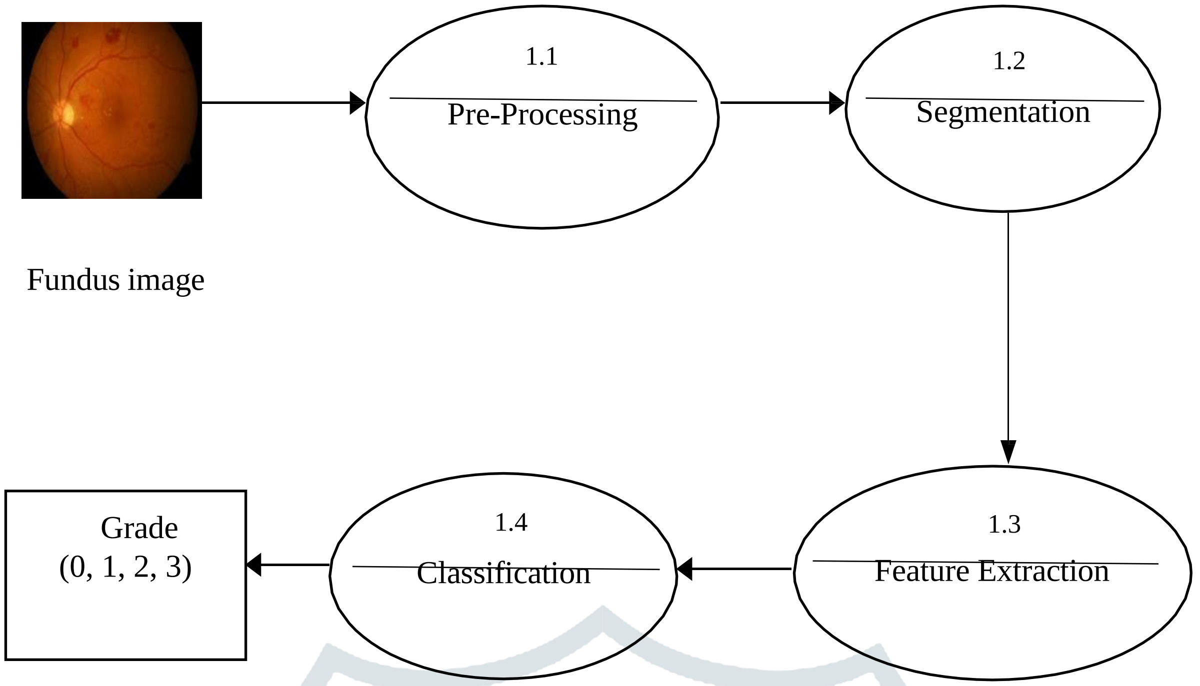

FIGURE 3 LEVEL 1 DFD

Each fundus image fed into the system undergoes a series of processes as shown in Figure 3.2, they are:

Level 1.1

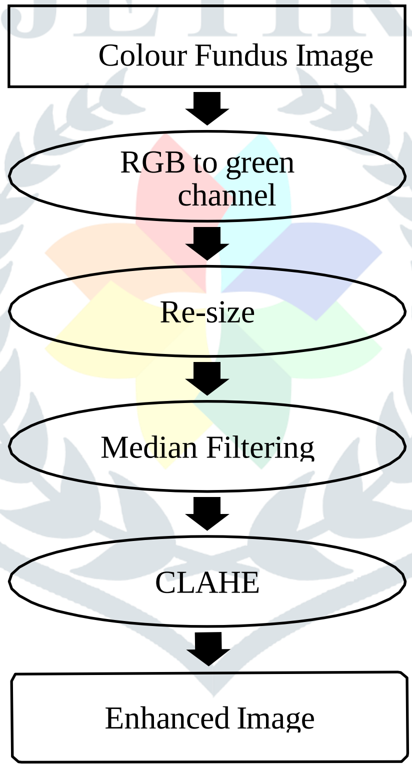

FIGURE 4 LEVEL 1.1 DFD

Preprocessing stage as shown in Figure 4 includes the following sub stages:

PRE-PROCESSING:

Patient movement, poor focus, bad positioning, reflections, inadequate illumination can cause a significant proportion of images to be of such poor quality as to interfere with analysis. In approximately 10% of the retinal images, artifacts are significant enough to impede human grading. Preprocessing of such images can ensure adequate level of success in the automated abnormality detection. In the retinal images there can be variations caused by the factors including differences in cameras, illumination, acquisition angle and retinal pigmentation. First step in the preprocessing is to attenuate such image variations by normalizing the color of the original retinal image against a reference image. Few of the retinal images acquired using standard clinical protocols often exhibit low contrast. Also, retinal images typically have a higher contrast in the center of the image with reduced contrast moving outward from the center. For such images, a local contrast enhancement method is applied as a second preprocessing step.

SEGMENTATION:

Involves the partitioning of an image or volume into Involves the partitioning of an image or volume into distinct (usually) non-overlapping regions in a meaningful way.

Segmentation-

- Identifies separate objects within an image Identifies separate objects within an image.
- Finds regions of connected pixels with similar properties.
- Finds boundaries between regions.
- Removes unwanted regions.

FEATURE EXTRACTION:

Feature extraction a type of dimensionality reduction that efficiently represents interesting parts of an image as a compact feature vector. This approach is useful when

image sizes are large and a reduced feature representation is required to quickly complete tasks such as image matching, classification and retrieval.

CLASSIFICATION:

Image classification analyzes the numerical properties of various image features and organizes data into categories. Classification algorithms typically employ two phases of processing: *training* and *testing*. In the initial training phase, characteristic properties of typical image features are isolated and, based on these, a unique description of each classification category, *i.e. training class*, is created. In the subsequent testing phase, these feature-space partitions are used to classify image features.

RGB TO GREEN CHANNEL:

The color image is converted to a gray scale image and then the green channel is extracted from it. Green channel is better than the red or blue channels because the red channel image is too bright and the blue channel image is too dark. All the anomalies are visible properly in the green channel image. A comparison of the images of the three channels is shown in Figure 5

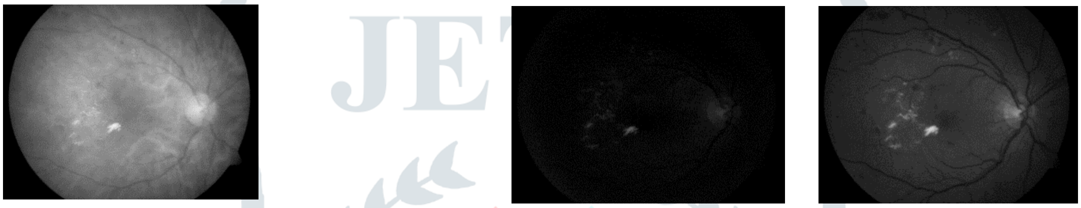

FIGURE 5 RED, BLUE AND GREEN CHANNEL IMAGES

Image resizing:
The green channel image is then resized to 560x720 Standard aspect ratio.

MEDIAN FILTERING:

One of the major advantages of pre-processing an image is to remove noise. Median filtering is one of the methods that is used for the same. Median filtering is a nonlinear method used to remove noise from images. It is widely used as it is very effective at removing noise while preserving edges. It is particularly effective at removing ‘salt and pepper’ type noise. The median filter works by moving through the image pixel by pixel, replacing each value with the median value of neighboring pixels. The pattern of neighbors is called the "window", which slides, pixel by pixel over the entire image 2 pixel, over the entire image. The median is calculated by first sorting all the pixel values from the window into numerical order, and then replacing the pixel being considered with the middle (median) pixel value.

CLAHE:

Contrast Limited Adaptive Histogram Equalization is one of the well-known enhancement techniques. In histogram equalization, the dynamic range and contrast of an image is modified by altering the image such that its intensity histogram has a desired shape. The intensity levels are changed such that the peaks of the histogram are stretched and the troughs are compressed. In contrast limited histogram equalization (CLHE), the histogram is cut at some threshold and then equalization is applied. Contrast limited adaptive histogram equalization (CLAHE) is an adaptive contrast histogram equalization method, where the contrast of an image is enhanced by applying CLHE on small data regions called tiles rather than the entire image. The output of this stage is an enhanced image.

Level 1.2

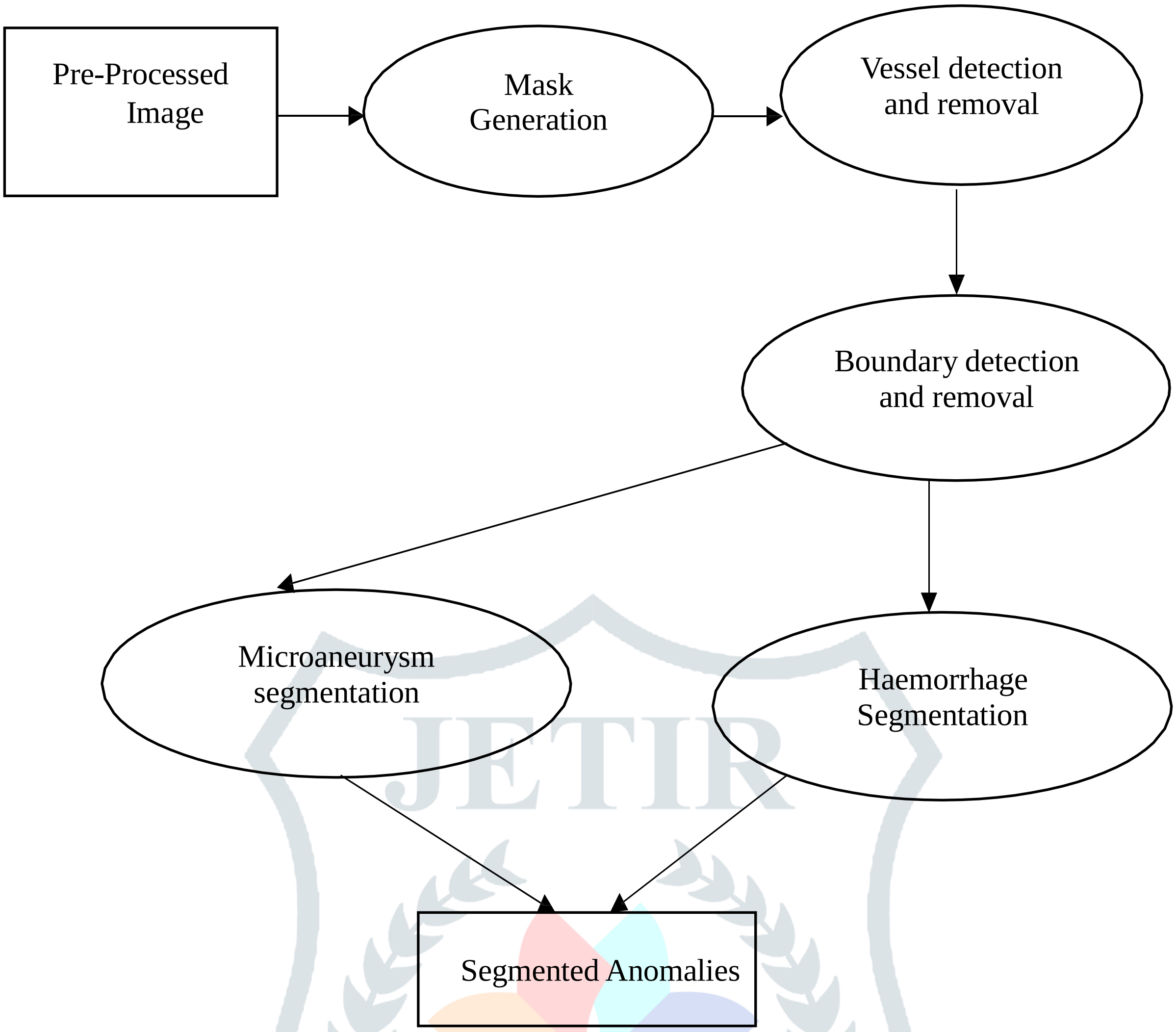

FIGURE 6 LEVEL 1.2 (SEGMENTATION) DFD

The enhanced image obtained in the previous stage is then segmented. Segmentation involves the following sub stages:

MASK GENERATION:

The mask is a binary image with the same resolution as that of fundus image whose positive pixels correspond to the foreground area. It is important to separate the fundus from its background so that the further processing is only performed for the fundus and not interfered by pixels belonging to the background. In a fundus mask, pixels belonging to the fundus are marked with ones and the background of the fundus with zeros. The fundus can be easily separated from the background by converting the original fundus image from the RGB to HSI color system where a separate channel is used to represent the intensity values of the image. The intensity channel image is threshold by a low threshold value as the background pixels are typically significantly darker than the fundus pixels. A median filter is used to remove any noise from the created fundus mask and the edge pixels are removed by morphological erosion. The final mask obtained is shown in Figure 3.6.

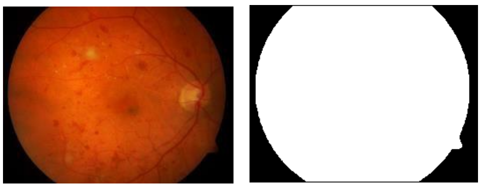

FIGURE 7 INPUT IMAGE AND THE GENERATED MASK

VESSEL DETECTION AND REMOVAL:

Blood vessel segmentation is a significant step in the red lesion detection. Since the blood vessels and red lesions namely microaneurysms and haemorrhages are both red in color the blood vessels need to be extracted out of the fundus image in order to effectively detect the microaneurysms and haemorrhages. A Contrast limited adaptive histogram equalization (CLAHE) is performed on the negative of green channel image. Top-hat filter operation is applied using a flat disc shaped structuring element. Top-hat filtering is the equivalent of subtracting the result of performing a morphological opening operation on

the input image from the input image itself. Suitable threshold is used to segment out the blood vessels. This threshold is selected based on the a priori knowledge of the quality of the image. The resultant image comprises of blood vessels along with haemorrhages, micro-aneurisms and other stray structures. After removing structures that have area less than a decided threshold the image containing only blood vessels is obtained.

BOUNDARY DETECTION AND REMOVAL:

The boundary of the fundus image gets separated in the process of segmentation and has to be removed as it causes false detections. By using the generated mask the boundary is removed.

Proposed algorithm

Input: Fundus image
Output: Grade of severity (0, 1, 2, 3)
Step 1: Input fundus image is retrieved from the test set Step 2: Green channel of the image is extracted
Step 3: The image is then passed through median filter Step 4: CLAHE is applied to the output of previous step
Step 5: The image is then resized to a standard size of 576*720

Step 6: Morphological operations are applied to extract microaneurysms and Haemorrhages
Step 7: Area and count of these anomalies are extracted as features. Statistical and GLCM features are also extracted
Step 8: The feature set extracted is then provided to the Random forest classifier for classification of severity levels.

Otsu thresholding algorithm

Otsu thresholding divides image into Foreground and Background Pixels, thus assigning Pixels nearer to the black level as 0 and white level as 1, converting image to binary. The thresholding identifies minimum variance between these pixels to aptly identify them.

Otsu's thresholding method involves iterating through all the possible threshold values and calculating a measure of spread for the pixel levels each side of the threshold,
i.e. the pixels that either fall in foreground or background. The aim is to find the threshold value where the sum of foreground and background spreads is at its minimum.

Otsu thresholding algorithm has the following steps:
Input: Pre-processed image
Output: Binary image
Step 1: Read a gray scale image. Step 2: Calculate image histogram.
Step 3: Select a threshold and referred as t,
3.1 Calculate foreground variance.
3.2 Calculate background variance. Step 4: Calculate Within-Class variance.
Step 5: Repeat steps 3 and 4 for all possible threshold value.
Step 6: Final global threshold, T = threshold in MIN (Within-class variance) Step 7: Binarize Image = gray scale image > T

Random Forest algorithm

The Random Forests algorithm is one of the best among classification algorithms - able to classify large amounts of data with accuracy. Random Forests are an ensemble learning method (also thought of as a form of nearest neighbor predictor) for classification and regression that construct a number of decision trees at training time and outputting the class that is the mode of the classes output by individual trees. Random Forests are a combination of tree predictors where each tree depends on the values of a random vector sampled independently with the same distribution for all trees in the forest. The basic principle is that a group of “weak learners” can come together to form a “strong learner”. Random Forests are a wonderful tool for making predictions considering they do not over fit because of the law of large numbers. Introducing the right kind of randomness makes them accurate classifiers and regressors. Single decision trees often have high variance or high bias. Random Forests attempts to mitigate the problems of high variance and high bias by averaging to find a natural balance between the two extremes. Considering that Random Forests have few parameters to tune and can be used simply with default parameter settings, they are a simple tool to use without having a model or to produce a reasonable model fast and efficiently.
Random Forests grows many classification trees. Each tree is grown as follows:

Input: Feature set

Output: Grade of severity
Step 1: If the number of cases in the training set is N, sample N cases at random - but with replacement, from the original data. This sample will be the training set for growing the tree.

Step 2: If there are M input variables, a number mM is specified such that at each node, m variables are selected at random out of the M and the best split on these m is used to split the node. The value of m is held.

Step 3: Constant during the forest growing.
Step 4: Each tree is grown to the largest extent possible. There is no pruning.
s.

III. RESULT

In DILATION

Suppose A and B are sets of pixels. Then the dilation of A by B, denoted $A \oplus B$, is defined as $A \oplus B = \cup x\in B\ Ax$. This means that for every point $x \in B$, A is translated by those coordinates. An equivalent definition is that $A \oplus B = \{(x, y) + (u, v): (x, y) \in A, (u, v) \in B\}$. Dilation is seen to be commutative, that $A \oplus B = B \oplus A$. Figure 5.1a shows
an original fundus image before dilation and Figure 5.1b shows the same image after dilation with disk shaped SE of radius 8. Optic disc becomes more prominent and exudates can also be seen near macula.

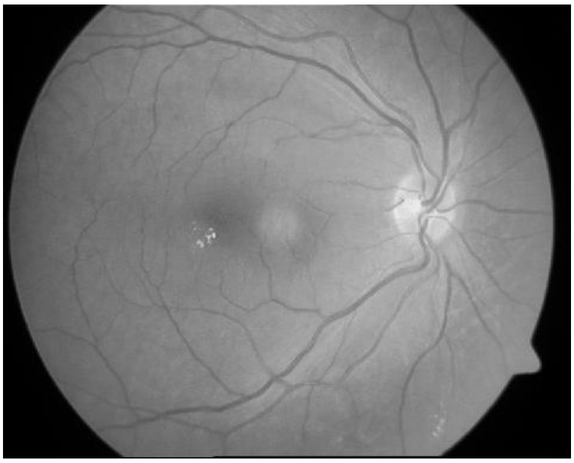

Figure 8.a: Original image

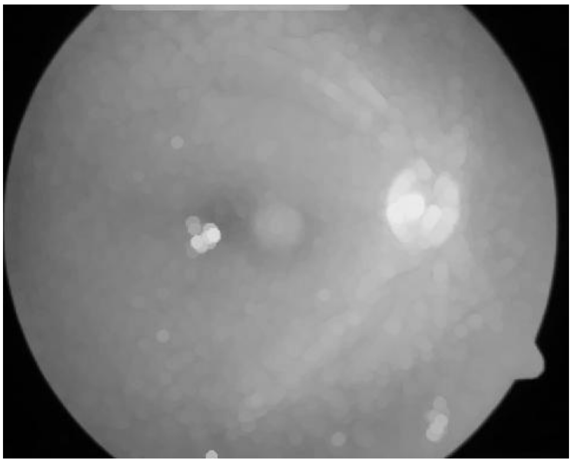

Figure 8.b: Image after dilation with disk shaped SE

EROSION

Given sets A and B, the erosion of A by B, written $A \ominus B$, is defined as $A \ominus B = \{w: Bw \subseteq A\}$. Figure 5.2a shows an original fundus image before dilation and Figure 9b shows the same image after erosion with disk shaped SE of radius 8. Blood vessels become more prominent.

Figure 9.a: Original image

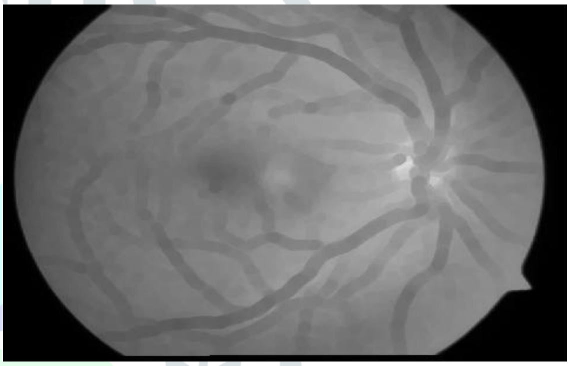

Figure 9.b: Image after erosion with disk shaped SE

Opening and Closing

Dilation and erosion are often used in combination to implement image processing operations. Erosion followed by dilation is called an open operation. Opening of an image smoothes the contour of an object, breaks narrow isthmuses ("bridges") and eliminates thin protrusions. Dilation followed by erosion is called a close operation. Closing of an image smoothes section of contours, fuses narrow breaks and long thin gulfs, eliminates small holes in contours and fills gaps in contours.

Opening operation of image is defined as $A \circ B = (A \ominus B) \oplus B$. Since opening operation of image consists of erosion followed by dilation, therefore it can also be defined as $A \circ B = \cup \{Bw : Bw \subseteq A\}$.

Closing operation of image is defined as $A \cdot B = (A \oplus B) \ominus B$. Figure 10a and Figure 10b shows the difference between opening operation and closing operation of fundus images.

Figure 10.a: Opening operation with disk shaped image

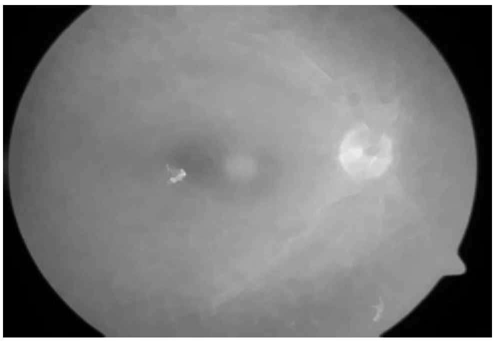

Figure 10.b: Closing operation with disk shaped SE image

Edge Detection

In an image, an edge is a curve that follows a path of rapid change in image intensity. Edges are often associated with the boundaries of objects in a scene. Edge detection refers to the process of identifying and locating sharp discontinuities in an image. It is possible to use edges to measure the size of objects in an image, isolate particular objects from their background, and to recognize or classify objects.

The Canny method finds edges by looking for local maxima of the gradient of I. The gradient is calculated using the derivative of a Gaussian filter. The method uses two thresholds, to detect strong and weak edges, and includes the weak edges in the output only if they are connected to strong edges. This method is therefore less likely than the others to be fooled by noise, and more likely to detect true weak edges. The Canny method performs better than the others due to the fact that it uses two thresholds to detect strong and weak edges and for this reason, Canny algorithm is chosen for edge detection.

Figure 11 a and b show original image and Canny edge detection methods respectively. It is apparent that by using Canny edge detection method, the weak fine blood vessels can be detected.

Figure 11.a: Original image

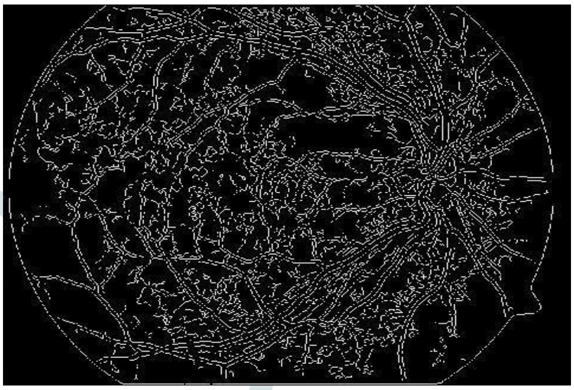

Figure 11.b: Canny edge detected image

Median Filtering

Median filtering is a nonlinear operation often used in image processing to reduce "salt and pepper" noise. A median filter is more effective than convolution when the goal is to simultaneously reduce noise and preserve edges. Median of a set is the middle value when values are sorted. For even number of values, the median is the mean of the middle of two. Figure 12a shows an illustration of a 3 x 3 median filter for a set of sorted values to obtain the median value.

55	70	57
68	260	63
66	65	62

55 57 62 63 **65** 66 68 70 260

65

Figure 12.a: Illustration of a 3 x 3 median filter

This method of obtaining the median value means that very large or very small values (noisy values) will be replaced by the value closer to its surroundings. Figure 12b shows
the difference before and after applying median filtering. The “salt and pepper” noise in the original image have been clearly reduced after applying the median filtering.

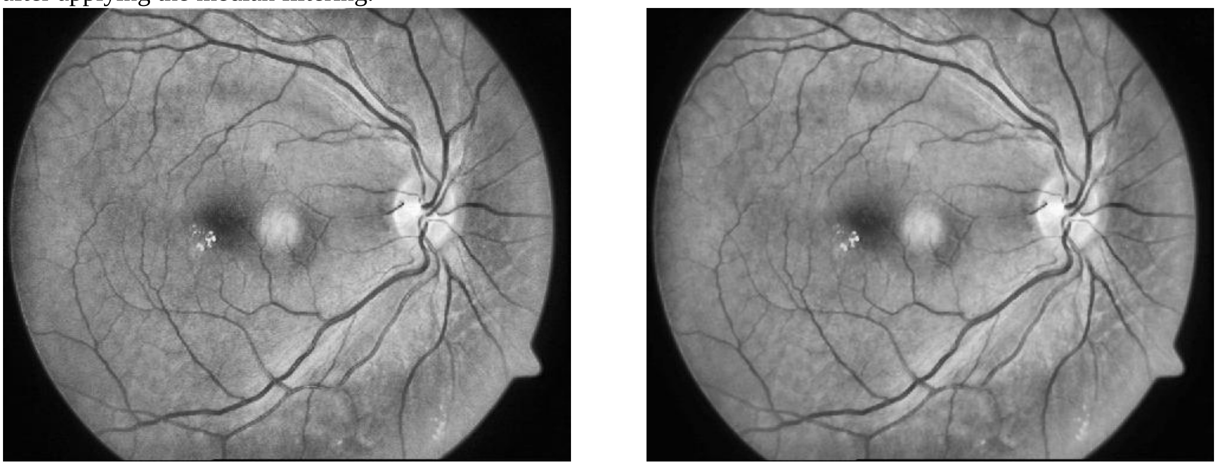

Figure 12.b: Original image (left) and image after median filtering (right)

Feature Extraction Texture Analysis

Texture describes the physical structure characteristic of a material such as smoothness and coarseness. It is a spatial concept indicating what, apart from color and the level of gray, characterizes the visual homogeneity of a given zone of an image. Texture analysis of an image is the study of mutual relationship among intensity values of neighboring pixels repeated over an area larger than the size of the relationship. The main types of texture analysis are structural, statistical and spectral.

Mean, standard deviation, third moment and entropy are statistical type. Mean, standard deviation and third moment are concern with properties of individual pixels.

The Gray Level Co-occurrence Matrix (GLCM) method is a way of extracting second order statistical texture features. A GLCM is a matrix where the number of rows and columns is equal to the number of gray levels, G, in the image. The matrix element P (i, j |

Δx, Δy) is the relative frequency with which two pixels, separated by a pixel distance (Δx,

Δy), occur within a given neighborhood, one with intensity 'i' and the other with intensity 'j'. The matrix element P (i, j | d, ө) contains the second order statistical probability values for changes between gray levels 'i' and 'j' at a particular displacement distance d and at a particular angle (ө). Using a large number of intensity levels G implies storing a lot of temporary data, i.e. a G × G matrix for each combination of (Δx, Δy) or (d, ө). Due to their large dimensionality, the GLCM's are very sensitive to the size of the texture samples on which they are estimated. Thus, the number of gray levels is often reduced. GLCM matrix formulation can be explained with the example illustrated in Figure 5.6 for four different gray levels. Here one pixel offset is used (a reference pixel and its immediate neighbor). If the window is large enough, using a larger offset is possible. The top left cell will be filled with the number of times the combination 0,0 occurs, i.e. how many time within the image area a pixel with grey level 0 (neighbor pixel) falls to the right of another pixel with grey level 0(reference pixel).

neighbour pixel value ---> ref pixel value:	0	1	2	3
0	0,0	0,1	0,2	0,3
1	1,0	1,1	1,2	1,3
2	2,0	2,1	2,2	2,3
3	3,0	3,1	3,2	3,3

Figure 13 GLCM calculation.

IV. Conclusion

The fast and efficient early detection of Diabetic Retinopathy is only possible if there is an effective method for segmenting the diabetic features in the fundus image. The proposed system presents a fast, effective and robust way of detecting diabetic features in the fundus images which can be used for classification of the images based on the severity of the disease. The retinal images are subjected to gray scale conversion, preprocessing and feature extraction steps. The extracted features are fed to a Random Forest classifier which will classify the images into different severity levels. Thus this Random forest technique has given a successful DR screening method which helps to detect the disease in multiple stages.

REFERENCES

[1.] Nguyen, Q. H., Muthuraman, R., Singh, L., Sen, G., Tran, A. C., Nguyen, B. P., & Chua, M. (2020, January). Diabetic retinopathy detection using deep learning. In Proceedings of the 4th international conference on machine learning and soft computing (pp. 103-107).

[2.] Fellah, K. M., Tigane, S., &Kahloul, L. (2023). Diabetic Retinopathy Detection Using Deep Learning. In International Symposium on Modelling and Implementation of Complex Systems (pp. 234-246). Springer, Cham.

[3.] Butt, M. M., Iskandar, D. A., Abdelhamid, S. E., Latif, G., &Alghazo, R. (2022). Diabetic Retinopathy Detection from Fundus Images of the Eye Using Hybrid Deep Learning Features. Diagnostics, 12(7), 1607.

[4.] Parthasharathi, G. U., Premnivas, R., & Jasmine, K. (2022). Diabetic Retinopathy Detection Using Machine Learning. Journal of Innovative Image Processing, 4(1), 26-33.

[5.] Bilal, A., Zhu, L., Deng, A., Lu, H., & Wu, N. (2022). AI-Based Automatic Detection and Classification of Diabetic Retinopathy Using U-Net and Deep Learning. Symmetry, 14(7), 1427.

[6.] Mane, D., Londhe, N., Patil, N., Patil, O., &Vidhate, P. (2022). A Survey on Diabetic Retinopathy Detection Using Deep Learning. In Data Engineering for Smart Systems (pp. 621-637). Springer, Singapore.

[7.] Ghaskadvi, M., Khochare, S., Gonsalves, R., &Dhamanskar, P. (2022). Pneumonia and Diabetic Retinopathy Detection Using Deep Learning Algorithm. In Sentimental Analysis and Deep Learning (pp. 155-175). Springer, Singapore.

[8.] Abbood, S. H., Hamed, H. N. A., Rahim, M. S. M., Rehman, A., Saba, T., &Bahaj, S. A. (2022). Hybrid Retinal Image Enhancement Algorithm for Diabetic Retinopathy Diagnostic Using Deep Learning Model. IEEE Access, 10, 73079-73086.

[9.] Gangwar, A. K., & Ravi, V. (2021). Diabetic retinopathy detection using transfer learning and deep learning. In Evolution in Computational Intelligence (pp. 679-689). Springer, Singapore.

[10.] Qureshi, I., Ma, J., & Abbas, Q. (2021). Diabetic retinopathy detection and stage classification in eye fundus images using active deep learning. Multimedia Tools and Applications, 80(8), 11691-11721.

[11.] Khan, Z., Khan, F. G., Khan, A., Rehman, Z. U., Shah, S., Qummar, S., ...& Pack, S. (2021). Diabetic retinopathy detection using VGG-NIN a deep learning architecture. IEEE Access, 9, 61408-61416.

[12.] Mushtaq, G., &Siddiqui, F. (2021, February). Detection of diabetic retinopathy using deep learning methodology. In IOP Conference Series: Materials Science and Engineering (Vol. 1070, No. 1, p. 012049). IOP Publishing.

[13.] Jadhav, A. S., Patil, P. B., &Biradar, S. (2021). Optimal feature selection-based diabetic retinopathy detection using improved rider optimization algorithm enabled with deep learning. Evolutionary intelligence, 14(4), 1431-1448.

[14.] Mishra, S., Hanchate, S., &Saquib, Z. (2020, October). Diabetic retinopathy detection using deep learning. In 2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE) (pp. 515-520). IEEE.

[15.] Alyoubi, W. L., Shalash, W. M., &Abulkhair, M. F. (2020). Diabetic retinopathy detection through deep learning techniques: A review. Informatics in Medicine Unlocked, 20, 100377.

[16.] Hemanth, D. J., Deperlioglu, O., &Kose, U. (2020). An enhanced diabetic retinopathy detection and classification approach using deep convolutional neural network. Neural Computing and Applications, 32(3), 707-721.

[17.] Karakaya, M., &Hacisoftaoglu, R. E. (2020). Comparison of smartphone-based retinal imaging systems for diabetic retinopathy detection using deep learning. BMC bioinformatics, 21(4), 1-18.

[18.] Tymchenko, B., Marchenko, P., &Spodarets, D. (2020). Deep learning approach to diabetic retinopathy detection. arXiv preprint arXiv:2003.02261.

[19.] Qummar, S., Khan, F. G., Shah, S., Khan, A., Shamshirband, S., Rehman, Z. U., ...&Jadoon, W. (2019). A deep learning ensemble approach for diabetic retinopathy detection. Ieee Access, 7, 150530-150539.

[20.] Arora, M., &Pandey, M. (2019, February). Deep neural network for diabetic retinopathy detection. In 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon) (pp. 189-193). IEEE.

[21.] Sengupta, S., Singh, A., Zelek, J., &Lakshminarayanan, V. (2019, September). Cross-domain diabetic retinopathy detection using deep learning. In Applications of Machine Learning (Vol. 11139, pp. 233-239). SPIE.

[22.] Zhang, W., Zhong, J., Yang, S., Gao, Z., Hu, J., Chen, Y., & Yi, Z. (2019). Automated identification and grading system of diabetic retinopathy using deep neural networks. Knowledge-Based Systems, 175, 12-25.

[23.] Chetoui, M., Akhloufi, M. A., &Kardouchi, M. (2018, May). Diabetic retinopathy detection using machine learning and texture features. In 2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE) (pp. 1-4). IEEE.

[24.] Suriyal, S., Druzgalski, C., &Gautam, K. (2018, March). Mobile assisted diabetic retinopathy detection using deep neural network. In 2018 Global Medical Engineering Physics Exchanges/Pan American Health Care Exchanges (GMEPE/PAHCE) (pp. 1-4). IEEE.

[25.] Ramachandran, N., Hong, S. C., Sime, M. J., & Wilson, G. A. (2018). Diabetic retinopathy screening using deep neural network. Clinical & experimental ophthalmology, 46(4), 412-416.

[26.] Gao, Z., Li, J., Guo, J., Chen, Y., Yi, Z., &Zhong, J. (2018). Diagnosis of diabetic retinopathy using deep neural networks. IEEE Access, 7, 3360-3370.

[27.] Wan, S., Liang, Y., & Zhang, Y. (2018). Deep convolutional neural networks for diabetic retinopathy detection by image classification. Computers & Electrical Engineering, 72, 274-282.

[28.] Gargeya, R., &Leng, T. (2017). Automated identification of diabetic retinopathy using deep learning. Ophthalmology, 124(7), 962-969.

[29.] Kanungo, Y. S., Srinivasan, B., &Choudhary, S. (2017, May). Detecting diabetic retinopathy using deep learning. In 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT) (pp. 801-804). IEEE.

[30.] SK, S. (2017). A machine learning ensemble classifier for early prediction of diabetic retinopathy. Journal of Medical Systems, 41(12), 1-12.