PROJECT REPORT

<u>BLINDNESS DETECTION: SEVERITY CLASSIFICATION OF DIABETIC RETINOPATHY USING YOLOV8</u>

Submitted in fulfilment of the requirement for the completion of my Internship

AT

CodeClause

Submitted By

Nathan Philip Bulla (Vellore Institute of Technology, Amaravati**)**

ABSTRACT

Diabetic retinopathy (DR) is a leading cause of vision impairment and blindness among diabetic individuals. Early detection and accurate severity classification of DR are critical for timely intervention and prevention of irreversible vision loss. This project leverages a comprehensive dataset from a Kaggle competition: *APTOS 2019 Blindness Detection*, comprising a diverse set of retinal images, to develop an advanced deep learning model for multi-class classification of diabetic retinopathy severity.

1. INTRODUCTION

As the primary cause of visual impairment and blindness in diabetics, diabetic retinopathy (DR) presents a serious global health concern. Accurate and effective techniques for identifying and categorising the severity of Diabetic Retinopathy (DR) are desperately needed, as the prevalence of the disease is rising globally. Early intervention is essential to preventing irreparable vision loss, underscoring the need for the medical community to develop reliable diagnostic techniques as soon as possible.

Using a variety of datasets from a Kaggle competition, this project uses deep learning approaches to address this difficulty. With the help of this diverse collection of retinal images, We want to create a sophisticated deep learning model that can be used for multi-class classification. The goal is to identify the various stages of diabetic retinopathy severity, from No DR, to Mild, to Moderate, to Severe to Proliferative DR.

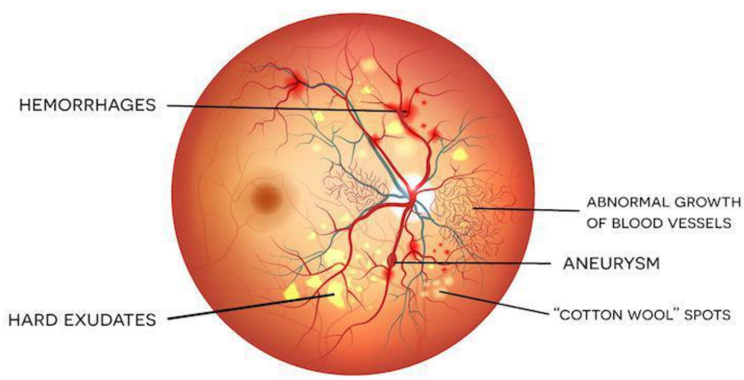

Fig 1. The different indicators related to Diabetic Retinopathy. [1]

1.1. Problem statement

The diagnosis and severity classification of diabetic retinopathy traditionally rely on manual assessment by ophthalmologists, a process that is time-consuming and subject to inter-observer variability. Leveraging the advancements in deep learning, this project aims to automate and enhance the accuracy of the diagnostic process. However, the complexity of retinal image analysis presents several challenges, such as subtle feature extraction and accurate severity level categorization.

The problem at hand involves constructing a deep learning model that can not only effectively learn and extract intricate features from retinal images but also classify them into distinct severity categories. To address this, the project employs preprocessing methods, including grayscale conversion, Gaussian blur, and cropping, to enhance the model's ability to discern relevant features from the images. Furthermore, the algorithm chosen for the classification task is YOLOv8, a state-of-the-art object detection model known for its accuracy and efficiency.

Through the integration of advanced preprocessing techniques and the utilisation of YOLOv8, We aim to develop a model that not only demonstrates superior performance in multi-class diabetic retinopathy severity classification but also showcases its potential for real-world clinical applications. The success of this project has the potential to significantly impact the field by providing a reliable and automated tool for early DR diagnosis, enabling timely medical interventions and ultimately improving patient outcomes.

2. METHODOLOGIES

2.1. Deep Learning

Deep learning, a subset of machine learning, has revolutionised the field of image classification by mimicking the human brain's intricate neural networks. At the core of deep learning for image classification lies the use of convolutional neural networks (CNNs), specialised architectures adept at automatically extracting hierarchical features from images. This enables the model to discern intricate patterns and representations, learning to differentiate between various objects or categories within images. Deep learning excels in image classification tasks due to its capacity to process vast amounts of data, automatically learn complex features, and adapt to diverse datasets. Through training on labelled examples, deep learning models iteratively refine their parameters, enhancing their ability to generalise and accurately classify unseen images. This transformative technology has found applications across numerous domains, from medical diagnosis to autonomous vehicles, establishing itself as a cornerstone in the realm of image classification.

2.2. Dataset

The dataset utilised is the one provided by Asia Pacific Tele-Ophthalmology Society (APTOS) on Kaggle as part of their ‘APTOS 2019 Blindness Detection’ Competition.

The dataset contains 5590 images pooled from various patients, of which 3662 are provided as labelled. The class labels are based on the severity of the disease, and Range from 0-4, where 0 signifies No Diabetic Retinopathy; 1, Mild; 2, Moderate; 3, Severe; and 4 signifies the highest severity of Proliferative Diabetic Retinopathy.

2.3. Preprocessing

2.3.1. Balance

Balance of the dataset refers to the distribution of the target class values across the dataset.
Ideally, the count for each class must be equal, i.e. a weight of 1 for each class.
But in the chosen dataset, we encounter an imbalance.

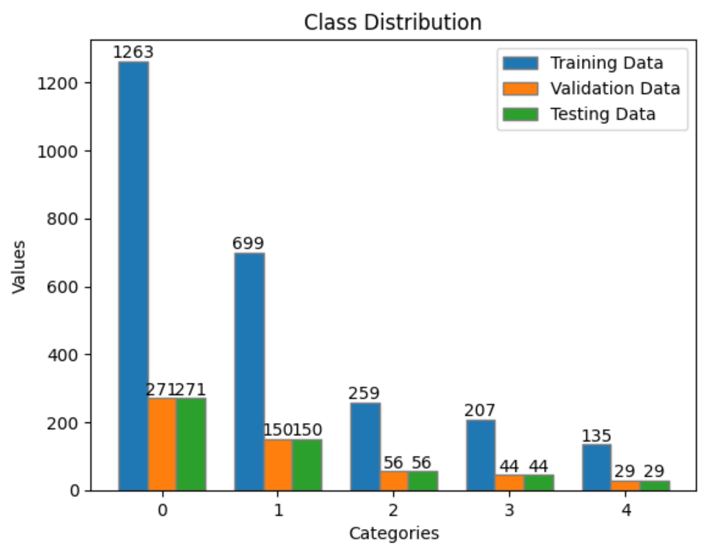

Fig.2 Class distribution

This can be tackled in many ways:

Upsampling: Refers to increasing the volume of the minority class data using techniques such as GAN (Generative Adversarial Networks), SMOTE (Synthetic Minority Oversampling Technique), Augmentation (in case of image or other unstructured data), etc.
Downsampling: Refers to decreasing the volume of the majority class to achieve balance, or utilising only a smaller subset of the majority class data.

In the proposed system, we use upscaling to tackle the class imbalance. But only in the Training dataset. The validation and test partitions are left as is to reflect the realistic distribution of cases.

2.3.2. Train Test Split

This technique consists of dividing the whole dataset into subsets of Train and Test, which will be used for Training and Testing the models respectively. It is one of the most fundamental steps in machine learning, which can help simulate the performance of the models created on unseen data, satisfied by evaluating the performance of the model on the testing dataset.

In this project, we have used a ratio of 0.3, meaning 70% of data considered as Training, and 30% for testing.

2.3.3. Colour Space Conversion

We use the cvtColor function found in the OpenCV (cv2) package to convert the colour space of the images.
We use it to convert them from BGR (Blue, Green, Red) format to RGB (Red, Green, Blue) format.

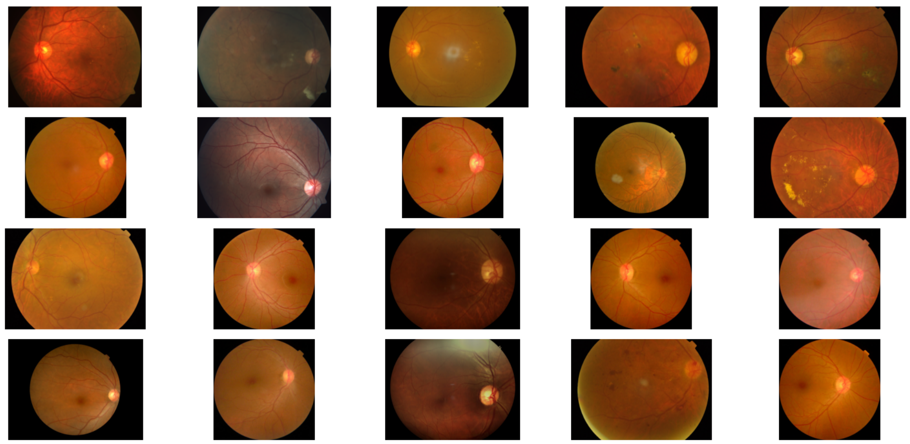

Fig. 3 After BGR2RGB

2.3.4. Grayscale

The ‘rgb2gray’ function is used to convert an RGB (Red, Green, Blue) image to grayscale. Grayscale images represent each pixel with a single intensity value, typically ranging from 0 (black) to 255 (white), where 0 represents the absence of light and 255 represents full intensity.

In the context of image processing, converting an RGB image to grayscale is a common operation. Grayscale images are often used when colour information is not essential for a particular task, and working with a single intensity channel simplifies the analysis or processing. In this case, using grayscale is useful for removing unnecessary colour information.

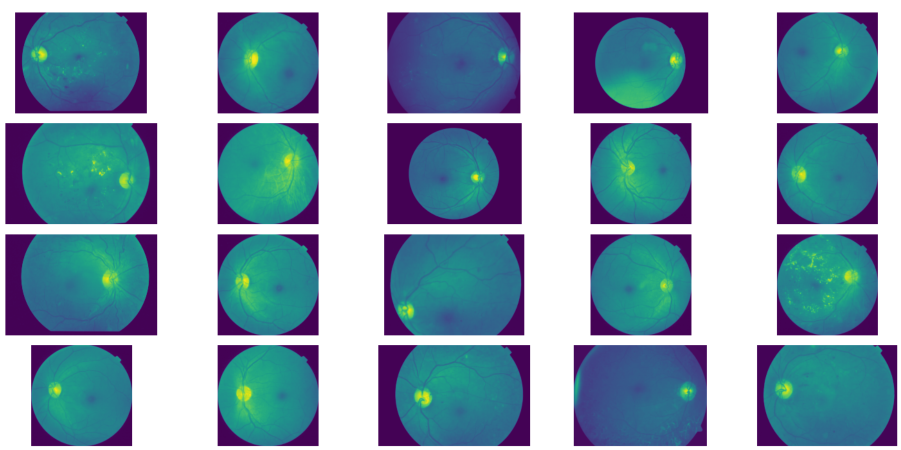

Fig. 4 After applying grayscale

2.3.5. Cropping

We use to reduce the amount of wasteful space found outside of the region of interest of the retinal image, signified by the black space found near the edges.

2.3.6. Weighted Blending and Gaussian blur

We use the cv2.addWeighted() function in OpenCV to perform a specific image manipulation. The source image undergoes a weighted blending operation with two components. The first component involves multiplying the entire source image by a weight of 4. The second component consists of applying a Gaussian blur to the source image using cv2.GaussianBlur() with a relatively high standard deviation (sigmaX=10). This blurred image is then multiplied by a weight of -4. Finally, the results of these two components are combined using the cv2.addWeighted() function, with an additional scalar of 128 added to the final output. The overall effect of this step is to enhance the edges and details of the image while controlling the brightness.

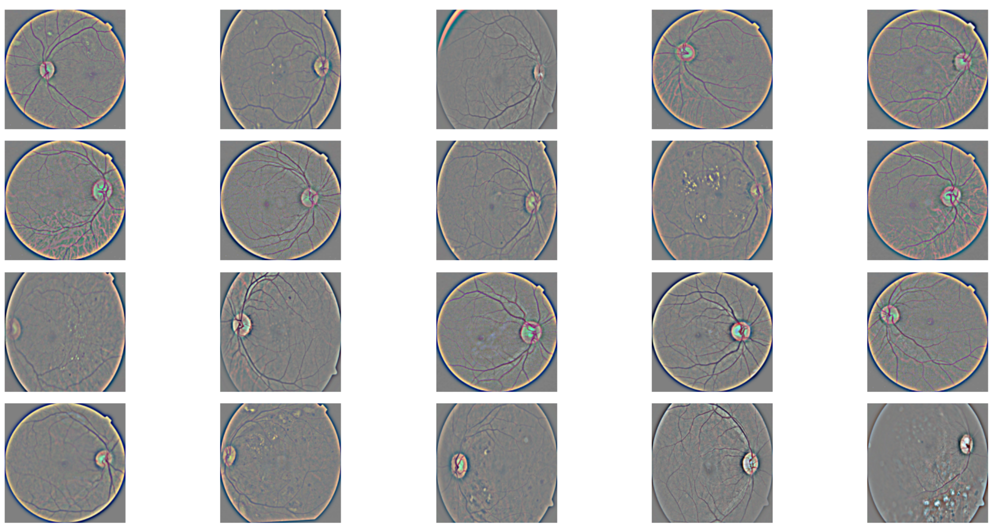

Fig. 5 After applying cropping, and weighted blending with gaussian blur.

2.4. Algorithms and Techniques

2.4.1. YOLOv8

The proposed model to be used is the You Only Look Once version 8 (YOLOv8), a State-of-the-art object detection algorithm that can perform a multitude of object detection, image classification, and image segmentation tasks.

YOLOv8 is a complex model, consisting of the Backbone, a neck, and multiple heads.
It is reliable and fast, making it ideal for real time applications.

The backbone of the model is based on the CSPDarknet53 architecture, and is responsible for feature extraction at high resolution.
The neck connects the backbone to the heads. It uses a C2f module to improve the object detection prediction.
The heads perform the function of prediction of the bounding boxes, class labels and the confidence scores. The model has three heads, each predicting a bounding box of a different scale, enabling the detection of differently sized objects.

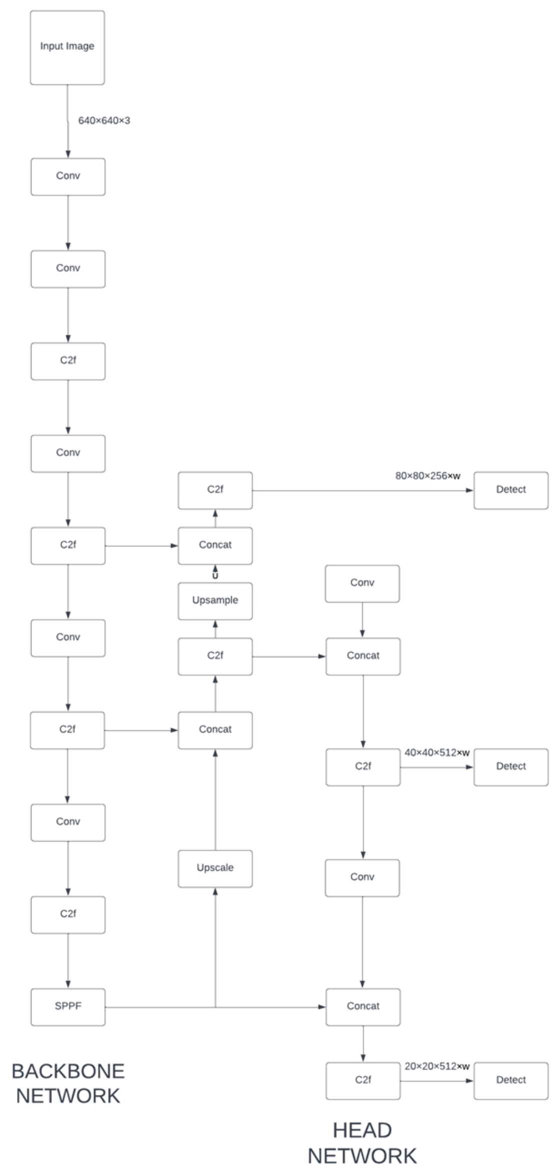

Fig. 6 Architecture of the YoloV8 model

3. RESULT ANALYSIS

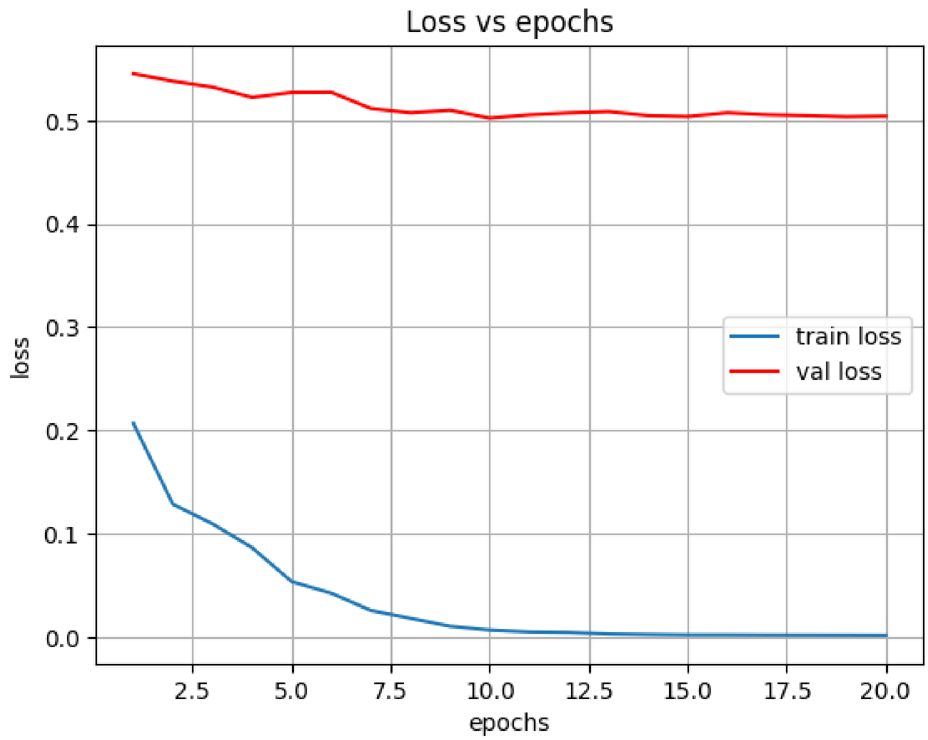

Fig. 7 Training and Validation Loss graph for 20 epochs

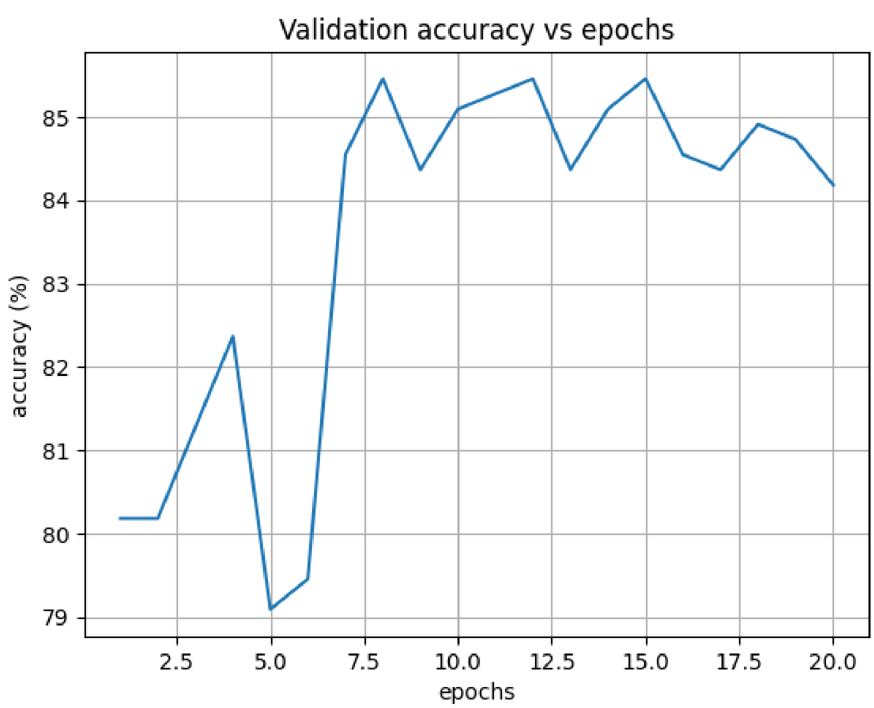

Fig. 8 Validation Accuracy Graph

The highest validation accuracy recorded in the course of the experiment, after testing multiple permutations of image sizes and cropping techniques was 85.5%

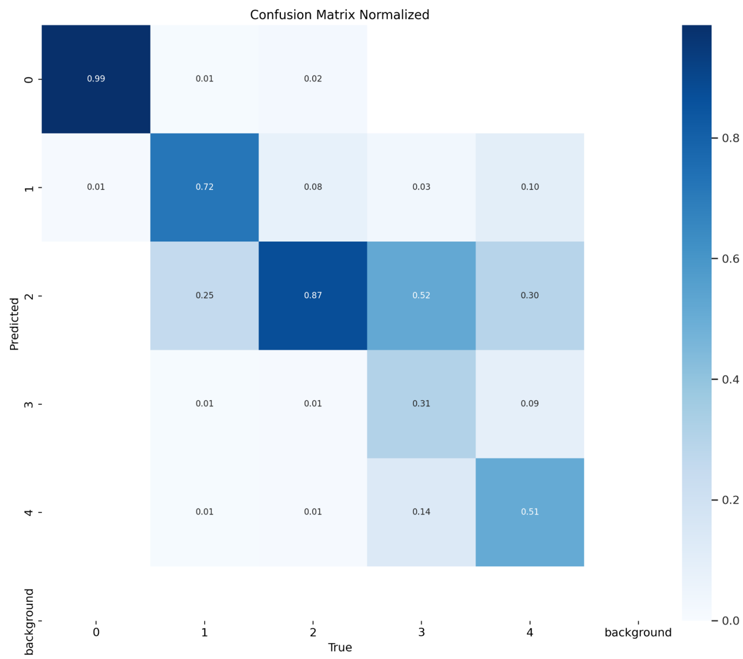

Fig. 9 Confusion Matrix

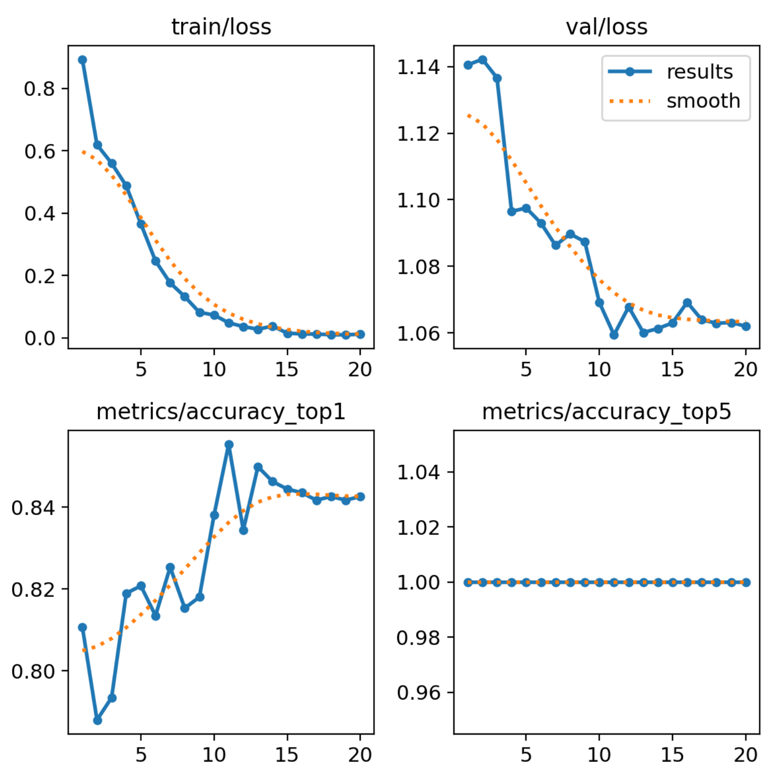

Fig. 10 Graph of top1 and top5 accuracies

7. CONCLUSION

In conclusion, this project represents a significant stride in leveraging deep learning techniques for the multi-class classification of diabetic retinopathy severity. The urgency to address the challenges posed by diabetic retinopathy as a leading cause of vision impairment and blindness necessitates innovative solutions, and our approach utilising a Kaggle competition dataset has yielded promising results. The application of advanced preprocessing methods, including grayscale conversion, Gaussian blur, and cropping, enhanced the model's ability to discern nuanced features from retinal images. The adoption of YOLOv8 as the classification algorithm further bolstered the model's performance, showcasing its efficiency in handling complex image data. The final validation accuracy of 85.5% attests to the robustness of our deep learning model in accurately categorising diabetic retinopathy severity levels. While there is room for further refinement and exploration, these findings underscore the potential of deep learning in automating and improving the diagnostic process for diabetic retinopathy, thereby contributing to early intervention and better patient outcomes in the realm of ophthalmic healthcare.

8. BIBLIOGRAPHY

- [1] Image showing indicators of Diabetic Retinopathy
https://www.eyeops.com/contents/our-services/eye-diseases/diabetic-retinopathy

- [2] APTOS DR dataset for blindness detection
https://www.kaggle.com/competitions/aptos2019-blindness-detection/data

- [3] Tensorflow Documentation
https://www.tensorflow.org/

- [4] OpenCV documentation
https://opencv.org/