



Faculty of Computer Science and Information Technology

**Classification of Diabetic Retinopathy from Fundus Images Using
Hybrid Deep Learning Feature Combination and Divide-and-Conquer
Approach**

Muhammad Mohsin Butt

**Doctor of Philosophy
2025**

Classification of Diabetic Retinopathy from Fundus Images Using Hybrid Deep Learning Feature Combination and Divide-and-Conquer Approach

Muhammad Mohsin Butt

A thesis submitted

In fulfillment of the requirements for the degree of Doctor of Philosophy

(Computer Science)

Faculty of Computer Science and Information Technology

UNIVERSITI MALAYSIA SARAWAK

2025

DECLARATION

I declare that the work in this thesis was carried out in accordance with the regulations of Universiti Malaysia Sarawak. Except where due acknowledgements have been made, the work is that of the author alone. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.



.....

Signature

Name: Muhammad Mohsin Butt

Matric No.: 17010199

Faculty of Computer Science and Information Technology

Universiti Malaysia Sarawak

Date : 06/02/2025

ACKNOWLEDGEMENT

First and foremost, I want to thank Almighty ALLAH the most merciful and gracious. I would also like to acknowledge all the people who have supported me throughout this work. I would like to express special appreciation to my parents and wife as their support and encouragement helped in completing the work presented in this thesis.

Next, I would like to acknowledge my mentor and advisor Associate Professor Ts. Dr. Dayang NurFatimah binti Awang Iskandar. The continuous support and guidance during the research proposal, thesis writeup and research publications played vital role in completing this work. Dr. Dayang's constructive advice, and insightful feedback were instrumental in shaping both the direction and conclusion of this thesis. Her support and encouragement were crucial in overcoming many challenges faced during my research. I truly felt privileged working under her supervision and for fostering an environment where my ideas and skills could flourish.

I would like to express my heartfelt thanks to the Centre for Graduate Studies and the management team at UNIMAS. Their guidance and support throughout my studies and residency at Universiti Malaysia Sarawak have been invaluable.

I would also like to thank my all colleagues and friends for their valuable feedback and assistance throughout the work. Lastly, I would like to dedicate the work to my late sister whose memory and influence have been a guiding light throughout this journey. Her unwavering belief in my abilities and her ever-present support, even in her absence, have been a source of strength and inspiration.

ABSTRACT

Diabetic Retinopathy (DR) is the leading cause of blindness in diabetic patients. Different ophthalmologists capture high-resolution fundus images of the eye with varying sizes and quality. The International Clinical Diabetic Retinopathy (ICDR) severity scales divide the progress of the DR into five stages. Multiclass classification of DR is a challenging task that relies on precisely identifying subtle anomalies. In this work, a multiclass classification system for DR from fundus images of the eye is proposed. The system considers the varying size and quality of the images and proposes a new pre-processing framework to deal with degraded fundus images. Two new frameworks are also proposed for multiclass DR classification. The processed images are used to extract high-quality features, which are passed on to different Machine Learning (ML) classifiers for training and testing. The first framework utilises the inherent feature extraction capabilities of Convolutional Neural Networks (CNN) to extract hybrid features that are used for classification. High-resolution fundus images can discern nuanced vessel textures and other critical features which can further improve multiclass DR classification but require high computational resources and memory requirements. The second framework introduces a memory efficient divide and conquer approach for extracting high-resolution fundus image features. The hybrid framework presented achieves a maximum accuracy of 92.69%, 88.02%, and 83.62% for the binary, three class, and five class DR classification respectively for the Diabetic Retinopathy Dataset (DDR) dataset. The divide and conquer approach further improve the accuracy of the system with values of 97.60%, 89.46%, and 85.64% for the binary, three class, and five class DR classification respectively for the DDR dataset. The results provide significant performance improvement compared to recent studies in the literature.

Keywords: Diabetic Retinopathy, features extraction, multiclass classification, machine learning, high resolution image features

Klasifikasi Retinopati Diabetes daripada Imej Fundus Menggunakan Gabungan Ciri Pembelajaran Dalam Hibrid dan Pendekatan Bahagi-dan-Takluk

ABSTRAK

Retinopati Diabetik (DR) adalah punca utama kebutaan pada pesakit diabetes. Pakar oftalmologi yang berbeza menangkap imej fundus mata resolusi tinggi dengan saiz dan kualiti yang berbeza-beza. Skala keterukan Retinopati Diabetik Klinikal Antarabangsa (ICDR) membahagikan kemajuan DR kepada lima peringkat. Pengelasan berbilang kelas DR ialah tugas mencabar yang bergantung pada mengenal pasti anomali halus dengan tepat. Dalam kerja ini, sistem klasifikasi berbilang kelas untuk DR daripada imej fundus mata dicadangkan. Sistem ini mempertimbangkan saiz dan kualiti imej yang berbeza-beza dan mencadangkan rangka kerja pra-pemprosesan baharu untuk menangani imej fundus yang terdegradasi. Dua rangka kerja baharu juga dicadangkan untuk klasifikasi DR berbilang kelas. Imej yang diproses digunakan untuk mengekstrak ciri berkualiti tinggi, yang dihantar kepada pengelas Pembelajaran Mesin (ML) yang berbeza untuk latihan dan ujian. Rangka kerja pertama menggunakan keupayaan pengekstrakan ciri bawaan Rangkaian Neural Convolutional (CNN) untuk mengekstrak ciri hibrid yang digunakan untuk pengelasan. Imej fundus resolusi tinggi boleh membezakan tekstur kapal bernuansa dan ciri kritikal lain yang boleh meningkatkan lagi klasifikasi DR berbilang kelas tetapi memerlukan sumber pengiraan yang tinggi dan keperluan memori. Rangka kerja kedua memperkenalkan pendekatan pembahagian dan penaklukan yang cekap memori untuk mengekstrak ciri imej fundus resolusi tinggi. Rangka kerja hibrid yang dibentangkan mencapai ketepatan maksimum 92.69%, 88.02% dan 83.62% untuk klasifikasi DR perduaan, tiga kelas dan lima kelas masing-masing untuk dataset Set Data Retinopati Diabetik (DDR). Pendekatan divide and conquer menambah baik lagi ketepatan sistem dengan nilai 97.60%,

89.46%, dan 85.64% untuk klasifikasi DR binari, tiga kelas dan lima kelas masing-masing untuk dataset DDR. Hasilnya memberikan peningkatan prestasi yang ketara berbanding kajian terkini dalam literatur.

Kata kunci: Retinopati Diabetik, pengekstrakan ciri, klasifikasi berbilang kelas, pembelajaran mesin, ciri imej resolusi tinggi

TABLE OF CONTENTS

	Page
DECLARATION	i
ACKNOWLEDGEMENT	ii
ABSTRACT	iii
<i>ABSTRAK</i>	v
TABLE OF CONTENTS	vii
LIST OF TABLES	xii
LIST OF FIGURES	xv
LIST OF ABBREVIATIONS	xix
CHAPTER 1 INTRODUCTION	1
1.1 Background	1
1.2 Motivation	4
1.3 Problem Statement	5
1.4 Research Questions	7
1.5 Research Objectives	7
1.6 Methodology	9
1.7 Expected Outcomes	11
1.8 Research Scope	12
1.9 Research Significance	12

1.10	Thesis Structure	13
1.11	Summary	13
CHAPTER 2	LITERATURE REVIEW	14
2.1	Diabetes Mellitus	14
2.2	Diabetic Retinopathy	16
2.2.1	Types of Diabetic Retinopathy	17
2.2.2	Fundus Images	18
2.2.3	Treatment of Diabetic Retinopathy	20
2.3	Image-based Pre-processing	20
2.3.1	Contrast Enhancement and Colour Space Transformation	21
2.3.2	Image Normalisation	23
2.3.3	Gaussian Filtering	23
2.3.4	Image Resizing and Thresholding	24
2.3.5	Data Augmentation	25
2.4	Features Extraction	27
2.4.1	Traditional Feature Extraction Methods	28
2.4.2	CNN based Feature Extraction	29
2.4.3	Transfer Learning based Feature Extraction of Fundus Images	30
2.5	Fundus Image Classification	32
2.5.1	Fundus Image Classification using Machine Learning	33

2.5.2	Fundus Image Classification using Deep Learning	36
2.5.3	High Resolution Fundus Image Classification using CNN	40
2.6	Convolutional Neural Networks Models	43
2.6.1	AlexNet	46
2.6.2	GoogleNet	47
2.6.3	VGG16	50
2.6.4	ResNet	50
2.6.5	EfficientNet	52
2.7	Evaluation Metrics	53
2.7.1	Macro and Weighted Averages	56
2.8	Summary	57
CHAPTER 3	MATERIALS AND METHODS	59
3.1	Overview	59
3.2	Experimental Environment	61
3.3	Experimental Datasets	61
3.3.1	DDR Dataset	61
3.3.2	APTOS Dataset	63
3.3.3	EyePACS Dataset	65
3.3.4	Proposed Custom Dataset	67
3.4	Proposed Framework for Pre-Processing fundus images	68

3.5	Proposed Framework for DR Image Classification	74
3.5.1	Hybrid CNN Features based DR Image Classification	74
3.5.2	High-Resolution Memory Efficient Divide and Conquer approach DR image classification	77
3.6	Summary	80
CHAPTER 4	RESULTS AND DISCUSSION	81
4.1	Overview	81
4.2	DR Classification using proposed Pre-Processing Framework	82
4.3	Binary DR Classification using Pre-Processed Fundus Images	89
4.3.1	Binary DR classification using CNN Features extraction and Transfer Learning	89
4.3.2	Binary DR classification using Hybrid Features extraction and Transfer Learning	96
4.3.3	Binary DR Classification using localized Divide and Conquer approach	100
4.4	Multiclass DR Classification using Pre-Processed Fundus Images	111
4.4.1	Multiclass DR classification using CNN Features extraction and Transfer Learning	111
4.4.2	Multiclass DR classification using Hybrid Features extraction and Transfer Learning	117
4.4.3	Multiclass DR Classification using Localized Divide and Conquer Approach.	120
4.5	Five Stage Multiclass DR Classification using Pre-Processed Fundus Images	130

4.5.1	Five Stage Multiclass DR classification using CNN Features extraction and Transfer Learning	131
4.5.2	Five Stage Multiclass DR classification using Hybrid Features extraction and Transfer Learning	136
4.5.3	Five Stage Multiclass DR Classification using Localized Divide and Conquer Approach.	140
4.6	Summary	150
CHAPTER 5 CONCLUSION AND RECOMMENDATIONS		152
5.1	Introduction	152
5.2	Contributions	152
5.2.1	Pre-Processing Framework for Fundus Images of Varying Size and Quality	153
5.2.2	DR Image Classification using Hybrid Features	154
5.2.3	DR Image Classification using Divide and Conquer Approach	155
5.3	Limitations	156
5.4	Future Work	157
5.5	Summary	157
REFERENCES		159
APPENDICES		177

LIST OF TABLES

	Page
Table 2.1: Confusion Matrix for Binary DR classification	54
Table 3.1: Summary of Different DR Image Processing Datasets	67
Table 3.2: Results of Proposed framework for Pre-Processing Fundus Images	73
Table 4.1: Parameters used for Experimentation	82
Table 4.2: Classification Results of Different CNN Models without Proposed Pre-Processing	83
Table 4.3: Classification Results of Different CNN Models with Proposed Pre-processing	85
Table 4.4: Average Comparison between Different CNN Models With and Without Proposed Pre-processing	86
Table 4.5: Experimental Results of Different ML Classifiers for Binary Classification using Features Extracted from the GoogleNet Model	90
Table 4.6: Experimental Results of Different ML Classifiers for Binary Classification using Features Extracted from the ResNet Model	91
Table 4.7: Experimental Results of Different ML Classifiers for Binary Classification using Features Extracted from the EfficientNet Model	92
Table 4.8: Experimental Results of Different ML Classifiers for Binary Classification using Features Extracted from the VGG16 Model	93
Table 4.9: Experimental Results of Different Classifiers for Binary Classification using Hybrid Features Extracted from VGG16 and EfficientNet	97
Table 4.10: Experimental Results of Different Classifiers for Binary Classification using Hybrid Features Extracted from VGG16 and ResNet	98
Table 4.11: Experimental Results of ML Classifiers for Binary Classification using Proposed Divide and Conquer Framework on the GoogleNet Model	101
Table 4.12: Experimental Results of ML Classifiers for Binary Classification using Proposed Divide and Conquer Framework on the ResNet Model	103

Table 4.13: Experimental Results of ML Classifiers for Binary Classification using Proposed Divide and Conquer Framework on the EfficientNet Model	104
Table 4.14: Experimental Results of ML Classifiers for Binary Classification using Proposed Divide and Conquer Framework on the VGG16 Model	105
Table 4.15: Comparison of Binary Classification Results of Proposed Frameworks with Recent Works	109
Table 4.16: Experimental Results of Different ML Classifiers for Multiclass Classification using Features Extracted from the GoogleNet Model	112
Table 4.17: Experimental Results of Different ML Classifiers for Multiclass Classification using Features Extracted from the ResNet Model	113
Table 4.18: Experimental Results of Different ML Classifiers for Multiclass Classification using Features Extracted from the VGG16 Model	115
Table 4.19: Experimental Results of Different ML Classifiers for Multiclass Classification using Features Extracted from the EfficientNet Model	116
Table 4.20: Experimental Results of Different Classifiers for Multiclass Classification using Hybrid Features Extracted from VGG16 and ResNet models	119
Table 4.21: Experimental Results of ML Classifiers for Multiclass Classification using Proposed Divide and Conquer Framework on the GoogleNet Model	122
Table 4.22: Experimental Results of ML Classifiers for Multiclass Classification using Proposed Divide and Conquer Framework on the ResNet Model	123
Table 4.23: Experimental Results of ML Classifiers for Multiclass Classification using Proposed Divide and Conquer Framework on the VGG16 Model	124
Table 4.24: Detailed Experimental Results of SVM Classifier for Multiclass Classification using Proposed Divide and Conquer Framework with Features Extracted from ResNet Model	126
Table 4.25: Comparison of Multiclass Classification Results of Proposed Frameworks with Recent Works	129
Table 4.26: Experimental Results of Different ML Classifiers for Five Stage DR Classification using Features Extracted from the GoogleNet Model	131

Table 4.27: Experimental Results of Different ML Classifiers for Five Stage DR Classification using Features Extracted from the ResNet Model	133
Table 4.28: Experimental Results of Different ML Classifiers for Five Stage DR Classification using Features Extracted from the VGG16 Model	134
Table 4.29: Experimental Results of Different Classifiers for Five Stage Multiclass DR Classification using Hybrid Features Extracted from VGG16 and ResNet Models	137
Table 4.30: Experimental Results of ML Classifiers for Five Stage Multiclass Classification using Proposed Divide and Conquer Framework on the GoogleNet Model	140
Table 4.31: Experimental Results of ML Classifiers for Five Stage Multiclass Classification using Proposed Divide and Conquer Framework on the ResNet Model	142
Table 4.32: Experimental Results of ML Classifiers for Five Stage Multiclass Classification using Proposed Divide and Conquer Framework on the VGG16 Model	143
Table 4.33: Detailed Experimental Results of SVM Classifier for Five Stage Multiclass Classification using Proposed Divide and Conquer framework with Features Extracted from VGG16 Model	145
Table 4.34: Comparison of Five Stage Multiclass Classification Results of Proposed Frameworks with Recent Works	148

LIST OF FIGURES

	Page
Figure 1.1: Image with Healthy (Left) and DR Affected (Right) Eye (Deepak et al., 2021)	2
Figure 1.2: Lesions Indicating Signs of DR in the Retina (Nneji et al., 2022)	3
Figure 1.3: Varying Quality of Retinal Images Captured from Different Sources (Rakhlin, 2017a)	6
Figure 1.4: Proposed Architecture for Classification of DR from Fundus Images	9
Figure 1.5: Proposed Work and Contribution for Detecting DR from Fundus Images	11
Figure 2.1: How Glucose is Extracted by the Body Cells	15
Figure 2.2: Different Type of Lesions in Fundus Images (Mohanty et al., 2023)	17
Figure 2.3: Canon CR-2 Fundus Camera (Canon, 2018)	19
Figure 2.4: General Workflow in Transfer Learning	31
Figure 2.5: Working of Support Vector Machines (Lu & Malan, 2023)	34
Figure 2.6: Working of a Perceptron in an Artificial Neural Network	37
Figure 2.7: Neural Network with Single Hidden Layer	38
Figure 2.8: Workflow in a Convolutional Neural Network	39
Figure 2.9: Level of Details in Fundus Images at Varying Resolution	41
Figure 2.10: Working of CNN Models	46
Figure 2.11: Convolution Layer in CNN	46
Figure 2.12: AlexNet Architecture for Image Classification (Butt et al., 2019)	47
Figure 2.13: Inception Module of GoogleNet CNN (Butt et al., 2022)	49
Figure 2.14: Architecture of GoogleNet utilising the Inception modules (Butt et al., 2022)	49
Figure 2.15: VGG 16 Architecture for Image Classification	50
Figure 2.16: Skip Connections in ResNet (Butt et al., 2022)	51

Figure 2.17: Architecture of ResNet-18 CNN Model (Butt et al., 2022)	52
Figure 2.18: Confusion Matrix for Multiclass Classification	55
Figure 2.19: Performance Metrics for Moderate DR (2) Class	56
Figure 3.1: Workflow Diagram of DR Classification from Fundus Images	60
Figure 3.2: Different Stages of DR in DDR dataset (Li et al., 2019b)	62
Figure 3.3: Data Distribution in each Class of DDR Dataset	63
Figure 3.4: Fundus Image Distribution of Different Classes in the APTOS Dataset	64
Figure 3.5: Various Stages of DR According to the ICDR Severity Scale in the APTOS Dataset (Karthik Maggie, 2019)	65
Figure 3.6: Dataset Distribution (image count) in EyePCS Dataset	66
Figure 3.7: Sample Images Suffering from Focus and Exposure Issues (Emma Dugas Jared, 2015)	66
Figure 3.8: Custom Dataset for DR Image Classification	68
Figure 3.9: Proposed Pre-Processing Framework for Enhancing Fundus Images	69
Figure 3.10: Effect of applying Otsu's Thresholding on Fundus Image	70
Figure 3.11: Cropped Retinal Structure from Input Fundus Image	70
Figure 3.12: Standard Deviation and Mean of Degraded and Reference Fundus Image	71
Figure 3.13: Colour Matching Input Fundus Image with Reference Image	72
Figure 3.14: CLAHE on Fundus Image using RGB to L*a*b Colour Transformation	72
Figure 3.15: Final Result of Pre-Processed Fundus Image	73
Figure 3.16: Proposed Algorithm for Hybrid Feature Extraction and Classification of Fundus Images	76
Figure 3.17: The Proposed Method for Hybrid Feature Extraction and Classification of Fundus Images	76
Figure 3.18: Proposed Divide and Conquer Approach for DR Image Classification	77
Figure 3.19: Divide and Conquer Approach for Extracting High Resolution Features from Input Images	78

Figure 3.20: Flowchart of proposed Memory Efficient Divide and Conquer Framework	80
Figure 4.1: Comparison between different CNN Models With and Without Proposed Pre-Processing Framework.	88
Figure 4.2: Ablation Study for Color Matching in the Pre-Processing Framework using VGG16 CNN Model.	89
Figure 4.3: Binary Classification Results with Features from VGG16 CNN Model for Different ML Classifiers.	96
Figure 4.4: Confusion Matrix for Binary Classification using Features Extracted from VGG16 for Different Classifiers	96
Figure 4.5: Binary Classification Results with Hybrid Features from VGG16 and ResNet for Different ML Classifiers	100
Figure 4.6: Confusion Matrix for Binary Classification using Hybrid Features Extracted from VGG16 + EfficientNet Models for Different Classifiers	100
Figure 4.7: Results from Binary DR Classification using Divide and Conquer Approach, Left: PDR Processed Image, Right: k Patches Extracted from the Image	101
Figure 4.8: Binary Classification Results of Proposed Divide and Conquer Framework using Features from the ResNet Model.	107
Figure 4.9: Sample Fundus Images from Binary DR Detection Correctly Classified as NDR, and PDR	108
Figure 4.10: Sample Fundus Images from Binary DR Detection Incorrectly Classified as NDR, and PDR	108
Figure 4.11: Confusion Matrix for Binary Classification using Proposed Divide and Conquer Approach with Features Extracted from ResNet Model	109
Figure 4.12: Three Class Classification Results with Fatures from ResNet CNN Model for Different ML Classifiers.	117
Figure 4.13: Three Class Classification with Hybrid Features from VGG16 and ResNet for Different ML Classifiers.	120
Figure 4.14: Confusion Matrix for Multiclass Classification using Hybrid Featuers Vector on Different ML Classifiers	120
Figure 4.15: Multiclass DR Classification Results of Proposed Divide and Conquer Framework using Features from the ResNet Model.	126

Figure 4.16: Confusion Matrix for Multiclass Classification using Proposed Divide and Conquer Approach with Features Extracted from ResNet Model	127
Figure 4.17: Sample Fundus Images from Multiclass DR Detection Correctly Classified as NDR, MDR and PDR	128
Figure 4.18: Sample Fundus Images from Multiclass DR Detection Incorrectly Classified as NDR, MDR and PDR	128
Figure 4.19: Five Stage Multiclass DR Classification Results with Features from VGG16 CNN Model for Different ML Classifiers.	136
Figure 4.20: Five Stage Multiclass DR Classification Results with Hybrid Features from VGG16 and ResNet for Different ML Classifiers.	139
Figure 4.21: Confusion Matrix for Five Stage Multiclass Classification using Hybrid Features Vector on Different ML Classifiers	139
Figure 4.22: Five Stage Multiclass DR Classification Results of Proposed Divide and Conquer Framework using Features from the VGG16 Model.	145
Figure 4.23: Confusion Matrix for Five Stage Multiclass Classification using Proposed Divide and Conquer Approach with Features Extracted from VGG16 Model	146
Figure 4.24: Sample Fundus Images from Five Stage DR Detection Correctly Classified as NDR, MiDR, MoDR, SDR, and PDR	147
Figure 4.25: Sample Fundus Images from Five Stage DR Detection that are Incorrectly Classified	147

LIST OF ABBREVIATIONS

CGS	Centre for Graduate Studies
UNIMAS	Universiti Malaysia Sarawak
DR	Diabetic Retinopathy
ICDR	International Clinical Diabetic Retinopathy
CNN	Convolutional Neural Networks
DM	Diabetes Mellitus
WHO	World Health Organization
MA	Microaneurysms
HE	Haemorrhage
EX	Exudates
DL	Deep Learning
SVM	Support Vector Machines
RF	Random Forest
NB	Naïve Bayes
MLP	Multi-Layer Perceptron
PDR	Proliferative Diabetic Retinopathy
NPDR	Non-Proliferative Diabetic Retinopathy
PRP	Pan Retinal Photocoagulation
VEGF	Vascular Endothelial Growth Factor
CAD	Computer Aided Design
AHE	Adaptive Histogram Equalisation
CLAHE	Contrast Limited Adaptive Histogram Equalisation

PDF	Probability Density Function
RIHE-RRVE	Radiance indicator-based histogram equalisation for recursive retinal vessel enhancement
RIHE-RVE	Radiance indicator-based histogram equalisation for retinal vessel enhancement
PSNR	Peak Signal-to-Noise Ratio
LBP	Local Binary Pattern
LTP	Local Ternary Pattern
LESH	Local Energy-based Shape Histogram
NM-LBP	Neighbour Mean – Local Binary Pattern
GM-LBP	Global Mean – Local Binary Pattern
APTOS	Asia Pacific Tele-Ophthalmology Society
DDR	Diabetic Retinopathy Dataset
IDRiD	Indian Diabetic Retinopathy Image Dataset
ResNet	Residual Neural Network
ODIRD	Ocular Disease Intelligent Recognition dataset
TL	Transfer Learning
kNN	k-Nearest Neighbour
ReLu	Rectified Linear Unit
ANN	Artificial Neural Network
RBF	Radial Basis Function
GPU	Graphics Processing Unit
TP	True Positives
TN	True Negatives
FP	False Positives

FN	False Negatives
TPR	True Positive Rate
TNR	True Negative Rate
RGB	Red, Green, Blue
HSV	Hue, Saturation, Value

CHAPTER 1

INTRODUCTION

1.1 Background

Diabetes Mellitus (DM) is a group of medical conditions in which the human body ends up having high blood sugar. There can be various causes of high blood sugar such as deficiency in insulin production or lack of cell response towards insulin (Ikram et al., 2024). The World Health Organization (WHO) indicated an increase in the DM in the future (Kropp et al., 2023). Diabetic Retinopathy (DR) is one of the complications that occur due to diabetes and major cause of blindness in diabetic patients. It mostly remains undetected until we reach the later stages of the disease. Therefore, its early detection is necessary to prevent vision loss (Bhulakshmi & Rajput, 2024). The increased sugar content in diabetic patients affect the vessels inside the retinal tissues. Capturing the tissue structure of the retina is of utmost importance in automatic detection of DR. Funduscopy is a medical imaging technique used to capture internal structure of the retina. The fundus images captured through this technique reveals different retinal structures of the eye. It includes details about the blood vessels inside the retinal tissues and the optic disc section of the eye.

The grading of DR images by an ophthalmologist is a lengthy process that requires meticulous examination. The different abnormalities caused by DR in the eye include red lesions like microaneurysms (MA) and intra retinal haemorrhages (HE). Besides these, white lesions that appear in the eye due to DR include exudates (EX) and cotton wool spots. A MA is a tiny aneurysm or swelling of a blood vessel (Agarwal & Bhat, 2023). These small aneurysms can weaken the capillary walls which can rupture and leak blood from the retinal vessels. The leaked blood due to microaneurysm causes haemorrhages (Saranya &

Umamaheswari, 2023) around the blood vessels inside the retina. There are many traditional image processing and machine learning techniques that are proposed in literature for isolating these lesions (Tsiknakis et al., 2021a).

The risk of DR increases with age and for people suffering from diabetes for a longer period. The progression of DR can be described in five stages according to the International Clinical Diabetic Retinopathy (ICDR) severity scale. The initial stage of ICDR severity scale i.e., No DR (Stage 0), indicates a healthy eye with no presence of DR. In next stage i.e., mild DR (Stage 1), microaneurysms begin to appear in fundus images. As the condition progresses to moderate DR (Stage 2), notable changes occur within the retinal vessels, leading to a loss of vessel integrity and compromised blood supply to the retina. Subsequent advancement to severe DR (Stage 3) entails blockages in blood supply to the vessels, resulting in the development of new vessels within the retinal tissues. The culmination of DR progression manifests in proliferative DR (Stage 4), where newly formed vessels extend into the vitreous gel. This advanced stage poses a substantial risk of permanent vision impairment, as the presence of these abnormal tissues can result in haemorrhaging and subsequent retinal damage. Figure 1.1 shows the view of an image experienced by a healthy eye and that of a patient suffering from stage 4 DR. The various lesions that can cause DR are shown in Figure 1.2.



Figure 1.1: Image with Healthy (Left) and DR Affected (Right) Eye (Deepak et al., 2021)

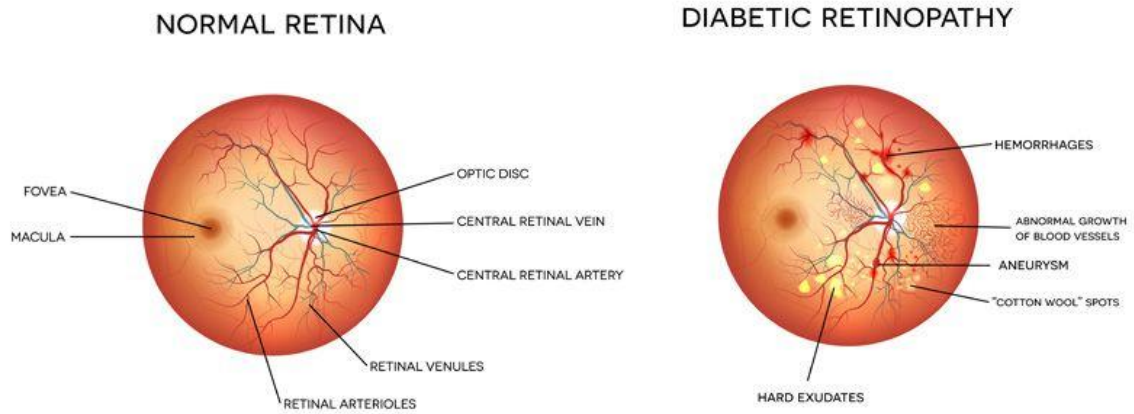


Figure 1.2: Lesions Indicating Signs of DR in the Retina (Nneji et al., 2022)

Multiclass DR classification is a challenging task due to the subtle variations between the DR classes and the complex retinal features. Modern approaches for DR detection use several machine learning and deep learning techniques to enhance the DR classification. Different architectures of CNN like ResNet, GoogleNet, EfficientNet etc., have been used primarily for extracting fundus image features and distinguishing between multiple DR classes (Mohanty et al., 2023). Some authors combine fundus images of left and right eye of the patient to extract meaningful insights that help address the interclass similarities (Gour & Khanna, 2021a). The input image resolution also plays vital role in extracting useful features for multiclass DR classification. However, higher resolution images require more network parameters to train and large memory bandwidth. Some custom architectures are also presented in literature that use attention mechanism and encoding part of the transformer models to extract image features that help in multiclass classification (Gour & Khanna, 2021a). For datasets with limited sample sizes, transfer learning on pre-trained models is utilized. Fine-tuning these pre-trained models for DR classification enhances their ability to extract relevant retinal features while mitigating the risk of overfitting due to insufficient

data (Gangwar & Ravi, 2021). Transfer learning can speed up the training process and provide ample features for problems with smaller datasets. In addition to CNN and transfer learning approaches, hybrid models are also used that combine feature extraction capability of CNN with the traditional ML classifiers like Support Vector Machines, Random Forrest etc. In summary, multiclass DR classification relies on a combination of deep learning architectures, transfer learning techniques, and hybrid models to address the challenges posed by limited data, subtle feature variations, and varying image quality of fundus images.

1.2 Motivation

Early diagnosis of DR in diabetic patients is essential for providing suitable treatment and preventing vision loss. However, the availability of specialised ophthalmologists and infrastructure remains a significant challenge, particularly in rural and underdeveloped regions. Such areas usually lack the necessary healthcare facilities and specialists capable of performing routine screening and diagnosis of DR. In this context, an automated DR classification system using fundus images offers a promising solution. Such a system can leverages various machine learning, and deep learning techniques to accurately detect and classify DR stages. The classification system can bridge the gap in healthcare access, enabling early and cost-effective screening even in remote areas where specialist care is not readily available. Automatic classification of DR into multiple stages is a challenging task (Gour & Khanna, 2021a). Variation in image quality such as differences in illumination, contrast resolution, and artefacts, can significantly affect the performance of DR diagnosis. Furthermore, the presence of inter-class similarities between different stages of DR poses a substantial challenge. Distinguishing between various stages, especially between stage 1-2, and 3-4, can be particularly difficult due to overlapping features and subtle differences in image characteristics. Recent approaches use Deep Learning (DL) techniques to extract the

low, mid, and high-level features of the input fundus images and classify the images into different DR stages (Gangwar & Ravi, 2021). The low-level features of an image correspond to minor details like lines, colour, contrast, edges, and corners etc. The mid-level features correspond to small structural objects within the image. The high-level features correspond to shapes and forms of objects within the image, which help in identifying the classes. The complexity in fundus images and importance of early detection provide the necessary motivation for the development of robust and accurate classification algorithms capable of identifying the different stages of DR. From a computer science perspective, the research seeks to provide useful feature extraction techniques, model optimisations, and improvements in the computational efficiency of the classification process that be generalised to diverse CAD-based medical imaging tasks.

1.3 Problem Statement

The aim of this research is to automate the detection of DR using colour fundus images. The fundus images of patients suffering from DR provided in various datasets are captured by different ophthalmologists using various hardware equipment operating under different environmental conditions. This results in colour fundus images of varying quality, including low contrast, uneven illumination, and varying resolution etc as shown in Figure 1.3. The common methods used in research to enhance the fundus images of the eye use different histogram equalisation techniques, However, limited work is done in utilising the individual Red (R), Green (G), or Blue (B) channels of the fundus image. The variance of blue channel can play important role in identifying degraded fundus images (C & R, 2023). The various image degradations need to be handled in the pre-processing stage of the proposed classification system.

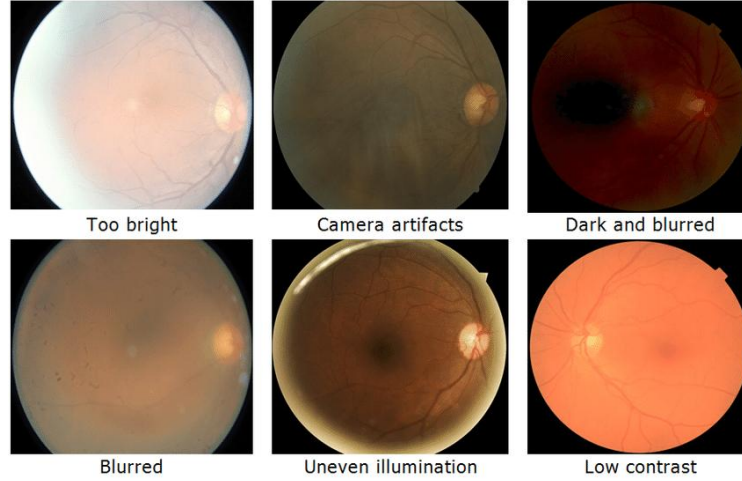


Figure 1.3: Varying Quality of Retinal Images Captured from Different Sources (Rakhlin, 2017a)

For image classification, most of the work in automatic DR detection focuses on binary classification of the given multiclass data by extracting different features from the fundus images (Sebastian et al., 2023). Although some research focuses on detecting the more than two stages of DR, the dataset used are of limited number of images that do not scale well with unseen data (Kar & Maity, 2018a). However, there are some datasets available that contain large number of fundus images e.g., EyePACS (Emma Dugas Jared, 2015) , but they suffer from major imbalance and quality issues. In some cases, balanced datasets are created using images from various other datasets to solve the imbalance problem (Monteiro, 2023).

The main limitations faced by researchers in multiclass DR classification is acquiring high quality image features that provide useful insights for fundus images. Some approaches use the classification layer of CNN and different ML models to classify DR images (Gour & Khanna, 2021a). However, the performance of the DR grading system is limited due to the use of single CNN during feature extraction. Combining features from multiple CNNs is a new approach that will be addressed in this work. Another approach in extracting high

quality features from fundus images is to increase the input image size used for the CNN models. However, increasing input image size puts a constraint on the available Graphics Processing Units (GPUs) memory. This work aims to address the problem of high GPU load when using high-resolution fundus images as input to DR classification system. To summarize, the frameworks proposed in this work will address the problems of low performance due to single CNN usage, and secondly, high GPU memory usage when using high-resolution images for DR classification.

1.4 Research Questions

- i. How to accurately pre-process the degraded images captured by different ophthalmologists with varying size and quality by utilising the individual channels of an RGB image?
- ii. How to extract high quality features from fundus images for multiclass DR classification using hybrid feature vector approach that uses multiple CNN models?
- iii. What modifications to existing CNN models can effectively handle high-resolution fundus image inputs, mitigate GPU memory constraints, and improve multiclass classification performance for diabetic retinopathy?

1.5 Research Objectives

The objectives of this research aim to enhance the detection of Diabetic Retinopathy using advanced computational techniques. The work aims to provide two main frameworks for multiclass DR classification that uses different image processing and machine learning

techniques to process and extract useful features from fundus images that are used by the learning systems. The list of objectives is given below:

- i. Propose an effective pre-processing method to enhance the input fundus images using individual color channels.
- ii. Propose a reliable framework using image processing, and machine learning methods that extracts hybrid high-quality image features from multiple CNN for multiclass DR classification to improve various performance metrics.
- iii. Propose a new framework based on recent CNN models, transfer learning and layers customisation to use high-resolution fundus images for multiclass DR classification without putting constraint on the limited GPU memory.

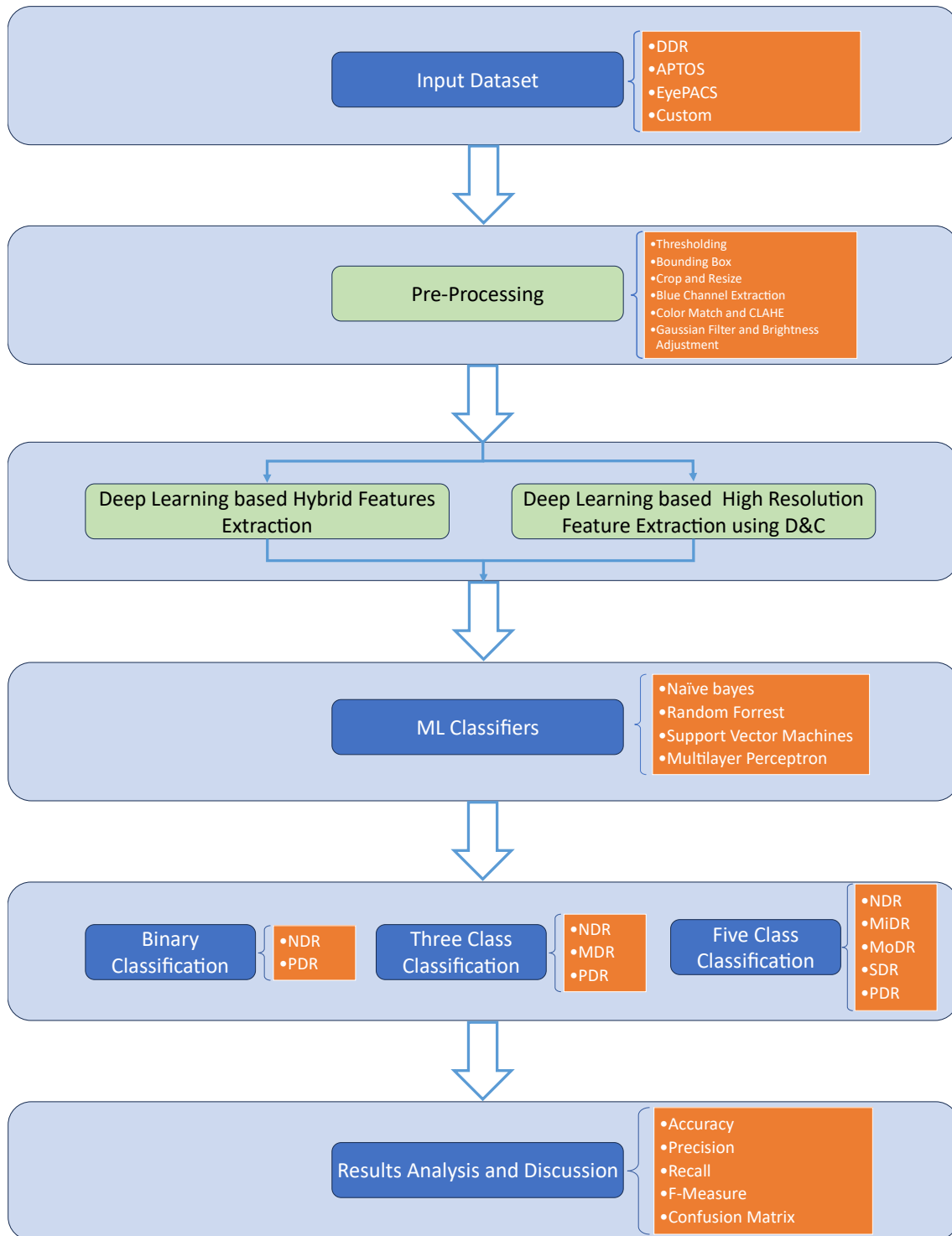


Figure 1.4: Proposed Architecture for Classification of DR from Fundus Images

1.6 Methodology

In this work, an automated Diabetic Retinopathy detection system is presented that will utilise various features of the fundus images. The system will use different image

processing and deep learning methods to classify DR images into various stages of DR corresponding to the ICDR severity scale. Figure 1.4 shows the proposed architecture for the detection of DR images.

In the pre-processing stage, the input images of varying quality are enhanced using different image processing methods. First, image thresholding and bounding box techniques are used to extract the retinal structure from the fundus image. Afterwards, the retinal structure is filtered to remove different artefacts. In the next stage, different features of the images are extracted, and classification is performed using various machine learning and deep learning techniques i.e., Convolutional Neural Networks based, Transfer Learning (TL) based and other machine learning techniques like Support Vector Machines (SVM) (Hearst et al., 1998), Multi-Layer Perceptron (MLP) (Gardner & Dorling, 1998), Naïve Bayes (NB) (Rish, 2001), and Random Forest (RF) (Tin Kam Ho, n.d., 2002). The work introduces two new frameworks for extracting high quality features from the fundus images of the eye. The first method generates a hybrid feature vector that is acquired by the help of TL and feature concatenation. Different CNN models such as AlexNet, ResNet, GoogleNet, EfficientNet and VGG16, are used to extract features from the fully connected layer at the end of the network. These features are combined and passed on to the ML classifiers for training and testing. In the second method for feature extraction, a new divide and conquer approach is proposed. The method modifies the recent CNN models, such as AlexNet, ResNet, GoogleNet, EfficientNet and VGG16, to extract local features from the fundus images that are helpful in differentiating inter-class similarities between different stages of DR. The second method presented in this work for feature extraction is based on principle of using high-resolution fundus images for better quality features. However, increasing the resolution of fundus images results in higher GPU memory usage as well as the training time

for different models. The proposed divide and conquer approach resolve this issue by dividing the high-resolution image into various patches and applying feature extraction on individual patches. The features acquired from the patches are then concatenated to create a feature vector that is used by different ML classifiers for training and testing. Figure 1.5 shows the work distribution and contribution of this research.

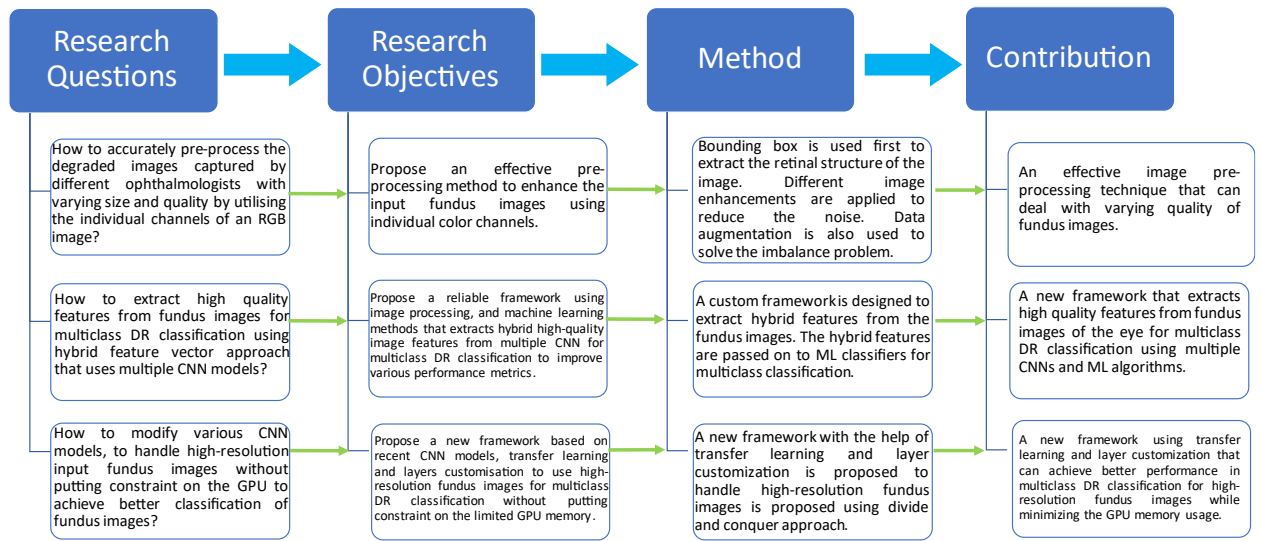


Figure 1.5: Proposed Work and Contribution for Detecting DR from Fundus Images

1.7 Expected Outcomes

The research outcomes of this study provide significant advancements for the automatic detection of Diabetic Retinopathy (DR) from fundus images of the eye. These outcomes include development of new techniques and framework designs to improve the accuracy and efficiency of DR diagnosis. The expected outcomes are as follows:

- i. An effective image pre-processing technique that can handle varying size and quality of fundus images.

- ii. A new framework that extracts high quality features from fundus images of the eye for multiclass DR classification using multiple CNNs and ML algorithms.
- iii. A new framework using transfer learning and layer customization that can achieve better performance in multiclass DR classification for high-resolution fundus images while minimizing the GPU memory usage.

1.8 Research Scope

The scope of this research work covers several key aspects of Diabetic Retinopathy (DR) classification. The research focuses on:

- i. Classifying DR into five stages using fundus images, with proposed frameworks for pre-processing and feature extraction.
- ii. Exploring the uses of multiple CNNs and memory efficient frameworks to improve the performance of multiclass DR classification.
- iii. The limitations of the proposed work related to datasets and time constraints are also highlighted.

1.9 Research Significance

The significance of the work presented in this thesis is listed below:

- i. Enhancing multiclass DR classification accuracy helps in timely discovery of the disease that helps in preventing vision loss.

- ii. Automated DR classification systems provide a cost-effective screening option for areas where specialized ophthalmologists are not available for screening.
- iii. The divide and conquer framework presented in this work provides resource-efficient deployments in locations where high processing power is not available.

1.10 Thesis Structure

The structure of the thesis provides a logical presentation of the research conducted on Diabetic Retinopathy detection. Chapter 1 covers the introduction, background, and motivation of the research. Chapter 2 provides the literature review in the field of automatic DR detection using fundus images of the eye. The proposed algorithms and methodology are further explained in Chapter 3. The results of the proposed techniques are discussed in Chapter 4.

1.11 Summary

To summarise, this chapter provides the basic introduction and motivation for automatic classification of DR from fundus images of the eye. The research questions and objectives are formulated based on addressing the fundus image variations and extracting high-quality features for multiclass DR classification. This work proposes a new framework for fundus image pre-processing. For feature extraction, two new frameworks are proposed that extract high-quality fundus image features for DR classification. Next chapter discusses the literature review in the field.

CHAPTER 2

LITERATURE REVIEW

2.1 Diabetes Mellitus

Diabetes is a medical condition that affects millions of people around the world. The number of people affected from Diabetes Mellitus (DM) is expected to reach around 700 million by the year 2045 (Tsiknakis et al., 2021b). Glucose is the fundamental source of energy for the cells that make up various tissues and muscles of the body. Insulin is the hormone that is responsible for transferring the glucose, present in the food intake, from the bloodstream into the body cells. The process is illustrated in Figure 2.1. The blood glucose stimulates the pancreas to generate insulin. The insulin further excites the body cells which then extract glucose from the bloodstream. Diabetes is caused when not enough glucose is extracted from the blood stream which results in high glucose level in the bloodstream. There can be various causes such as deficiency in insulin production or lack of cell response towards insulin (Poly et al., 2019; Zimmet et al., 2016). There are two main types of diabetes, which are listed below:

Type 1: In Type 1 diabetes, the pancreas is unable to produce enough insulin to trigger the sufficient glucose extraction from the bloodstream. It is an autoimmune condition in which the cells responsible for producing insulin in the pancreas are destroyed. This leaves the body cells with a deficiency of glucose meanwhile glucose levels in the bloodstream remains elevated. This type of diabetes condition occurs in childhood and adolescence. The immune system targets the insulin production cells in the pancreas which are necessary for regulating blood sugar levels (Banzi et al., 2021). Type 1 diabetic patients need to maintain insulin level at regular intervals to keep the blood sugar level stable.

Type 2: This type of DM is a chronic condition that develops in adulthood. In Type 2 diabetes, the body cells become resistant to insulin stimulation. This also results in the cells not extracting enough glucose from the blood resulting in high blood sugar level. Type 2 diabetes is a metabolic disorder which is caused due to various lifestyle factors such as lack of activity, obesity, and bad diet.

Both types of diabetes lead to various health complications if the treatment is not performed at an appropriate time. The health conditions resulting from diabetes include heart disease, kidney disease, damage to retina and nerves. Despite all the complications caused by diabetes, the condition can still be managed by keeping the blood glucose levels intact. This can be done by using appropriate medication, maintaining a regular workout routine, and keeping a healthy diet.

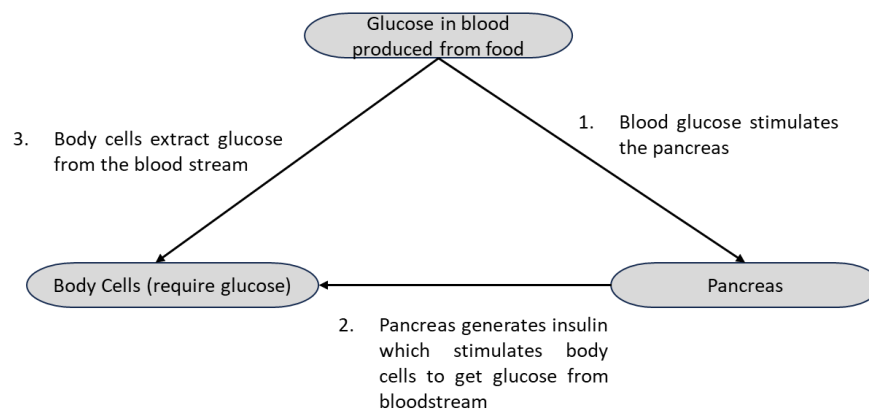


Figure 2.1: How Glucose is Extracted by the Body Cells

2.2 Diabetic Retinopathy

Diabetic Retinopathy (DR) is a complication that is caused by diabetes. The increased blood sugar levels of the diabetic patients can affect the blood vessels inside the retina of the eye. DR is a major cause of blindness in diabetes patients and often remains undetected in the early stages until the disease has progressed. Regular eye examination are necessary to avoid the complications caused by DR. In later stages of DR, the patient suffer from blurred vision, dark spots in vision, and eventually vision loss (Ghamdi, 2020). High blood sugar in diabetic patients leads to retinal vessel damage causing problems in light perception and the transfer of visual information to the brain. The different abnormalities caused by DR, such as Microaneurysm (MA), Haemorrhage (HE), Exudates (EX) and cotton wool spots, are explained below.

Microaneurysm (MA): A microaneurysm (Quellec et al., 2008) is a tiny aneurysm or swelling in the side of a blood vessel. These small aneurysms can weaken the capillary walls, which can rupture and leak blood from the blood vessel. Microaneurysms are one of the earliest signs of DR. They appear as small red dots in retinal photographs and are usually round or oval.

Haemorrhage (HE): A haemorrhage is caused by the leakage of blood from retinal blood vessels affected by microaneurysm (Gilliland & Folberg, 1992b). In retinal images, haemorrhages appear as dark reddish spots or blotches and may vary in size and position. They can obstruct normal vision if they appear in crucial regions and are a sign of blood vessel injury.

Exudates (EX): Exudates are white or yellowish deposits that appear in retinal images as a sign of DR. They usually appear as tiny, round, or oval-shaped lesions and are

the result of blood vessel leakage. Lipids, proteins, and other compounds called exudates seep from injured blood vessels and build up in the retinal tissues. Their presence can be interpreted as an indication of retinal tissue damage and frequently indicates the severity of diabetic retinopathy. Exudates can occasionally cluster or take on a "hard exudate" appearance, which may require treatment to prevent further retinal injury.

Various computer-based image processing techniques have been proposed in the literature for isolating these lesions (Chetoui et al., 2018a; Kar & Maity, 2018b). The various lesions that can cause DR are shown in Figure 2.2.

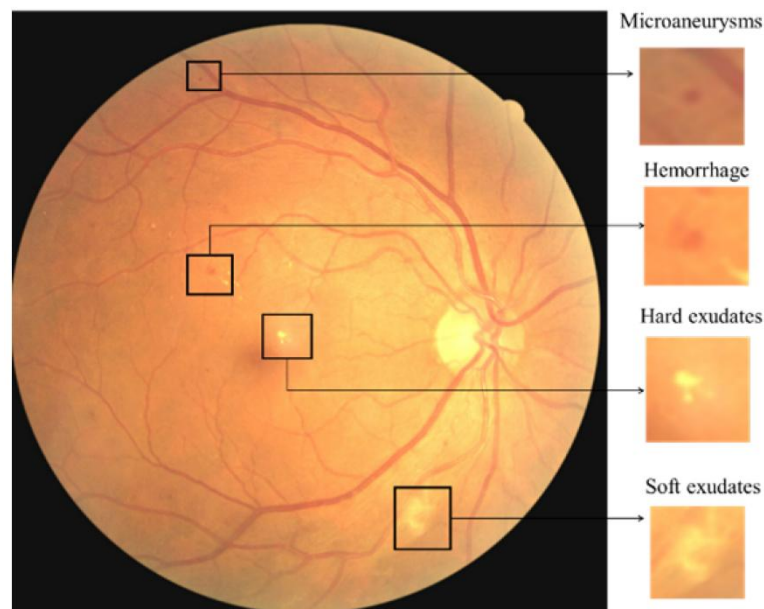


Figure 2.2: Different Type of Lesions in Fundus Images (Mohanty et al., 2023)

2.2.1 Types of Diabetic Retinopathy

There are two main types of diabetic retinopathy: Non-Proliferative Diabetic Retinopathy (NPDR) and Proliferative Diabetic Retinopathy (PDR). NPDR is the early stage of DR in which vessels inside the retina weaken and start leaking some blood into the retina.

Usually, the red lesions like MA or HE start appearing in NPDR. Proliferative DR is an advanced stage of the disease in which the damaged blood vessels close, causing growth of new abnormal vessels inside the retina. These abnormal vessels are weak and can cause blood leakage into the vitreous gel of the eye, leading to permanent vision loss. The new vessels can also damage the optic nerve that carries the visual information to the brain. The main types of DR are grouped into five stages according to ICDR (Haga et al., 2005) severity scale.

Stage 0: No DR: Indicates a healthy with no presence of DR.

Stage 1: Mild DR: In mild DR, some MA start appearing in the fundus image.

Stage 2: Moderate DR: In moderate DR, the retinal vessels lose their shape and ability to supply sufficient blood to the retina. At least one haemorrhage or microaneurysm and white lesions starts appearing in this stage.

Stage 3: Severe DR: In this stage, the blood supply is blocked, causing the growth of new vessels inside the retinal tissues. Multiple lesions are present in the fundus image at this stage.

Stage 4: Proliferative DR: In this advance stage of DR, the newly formed vessels inside the retinal tissues grow into the vitreous gel. These new tissues can cause permanent vision loss by leaking blood into the retina.

2.2.2 Fundus Images

Fundoscopy is a well-known and widely used imaging technique that is used to capture the back of the eye, known as the fundus (Chatziralli et al., 2012). In this imaging technique, the patient's eyes are dilated, and fundus camera is used to capture high-resolution

colour image of the fundus. The fundus image includes details about the retina, optic disc, and the blood vessels. The images produced by the camera are called fundus images. The process is simple and requires minimal risk. Fundus images can help in identifying abnormal bleeding of the vessels, scar tissues and other irregularities in the eye. These images are used by the ophthalmologists in diagnosing various stages of DR. A series of fundus photographs can provide a visual timeline in analysing the progression of the disease through various stages. Figure 2.3 shows the Canon CR-2 digital retinal fundus camera that comes with a 15-megapixel sensor having 45-degree angle of view. Besides this camera, there are various other fundus imaging cameras like Topcon NW500, and ZEISS VISUCAM 224. Topcon comes with a 12-megapixel sensor having a 50 – degree field of view. The ZEISS camera provides a better image quality with a 24-megapixel sensor having a 45 – degree field of view. Different cameras are available for capturing fundus images of the eye. The main differentiators between the cameras are the resolution, field of view, and cost.



Figure 2.3: Canon CR-2 Fundus Camera (Canon, 2018)

2.2.3 Treatment of Diabetic Retinopathy

The treatment of DR depends on the severity of the condition. Various methods are used to manage the complications caused by DR such as scatter laser treatment, surgical methods, and injections-based treatments.

Pan Retinal Photocoagulation (PRP): PRP is a scatter laser treatment used to treat leaking blood vessels in patients suffering from proliferative DR. In this treatment, a scattered laser is applied to various parts of the retina to target abnormal blood vessels and reduce their growth.

Vitrectomy: This is a surgical procedure performed to address blood leakage and scar tissues in the vitreous gel of the eye. During the procedure, the vitreous gel is removed from the eye and replaced with clear saline solution.

Anti-vascular endothelial growth factor (anti-VEGF) injections: These injections are used to counter the VEGF, which is protein produced by the cells for the generation new blood vessels. High blood sugar in diabetic patients causes an increase in this protein, resulting in new vessels generated inside the retina. Anti VEGF injections are applied to stop the abnormal growth of new blood vessels.

2.3 Image-based Pre-processing

In CAD-based classification systems for detecting DR, the most fundamental step is to pre-process the input images. The fundus images are captured using a variety of hardware devices under different environmental conditions. Additionally, different ophthalmologists can produce varying quality of fundus images, which affects the performance of the system. To avoid such heterogeneity in the input data, various image pre-processing techniques are

used in the literature to reduce the noise present in the input image. Contrast enhancement and color space transformations are some of the most widely used techniques for fundus image enhancement (Daniel & Anitha, 2015).

2.3.1 Contrast Enhancement and Colour Space Transformation

Contrast enhancement is a very common image pre-processing technique that is used to highlight the foreground details of an image. A simple approach for contrast enhancement of fundus images is called histogram equalisation (Zhang et al., 2019). The histogram of an image represents the number of pixels for each intensity value within the image. A narrow width histogram of an image indicates a low contrast image. Histogram equalisation redistributes the pixel intensities to spread out the width of histogram plot of the image. The process increases the global contrast of the image but neglects the local variations.

Adaptive Histogram Equalisation (AHE): This method accounts for the local variations of the image by computing several histograms, each corresponding to a separate section of the image. This enhances the local features and edges within the fundus image (Stark, 2000).

Contrast Limited Adaptive Histogram Equalisation (CLAHE): Adaptive histogram equalisation can result in amplifying the noise alongside contrast in the input image. To overcome this problem, CLAHE divides the image into subsections called tiles. In each tile, adaptive histogram equalisation is performed by setting a clip limit to avoid over amplification. The excess is redistributed using various mapping function like Rayleigh, exponential or uniform. The neighbourhood tiles are then combined using bilinear interpolation (Zuiderveld, 1994).

Colour space transformation is another common process used to adjust the contrast of fundus images. CIELAB colour space also referred as L*a*b colour space decouples brightness from chromatic information, allowing for more precise contrast adjustments. For colour input images, the image is first converted to L*a*b colour space and the contrast enhancement is performed on the luminance channel. CLAHE in the L*a*b colour space facilitates targeted enhancement, mitigating the risk of over-amplifying noise or artefacts present in the image. Conversion of RGB to L*a*b colour space involves multiple steps to represent colours accurately (Kuru & Girgin, 2011). These steps are shown through Equation 2.1 to 2.5. First, each channel in the RGB colour image is normalised between the range [0, 1]. The normalised channels are then transformed into intermediate XYZ components using pre-defined transformation matrix given by Equation 2.1.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \cdot \begin{bmatrix} R_n \\ G_n \\ B_n \end{bmatrix} \quad \text{Equation 2.1}$$

The XYZ values are normalised using the International Commission of Illumination (CIE) standard daylight condition given by the Equation 2.2.

$$\begin{bmatrix} X_n \\ Y_n \\ Z_n \end{bmatrix} = \begin{bmatrix} X/X_w \\ Y/Y_w \\ Z/Z_w \end{bmatrix} \quad \text{Equation 2.2}$$

Where $X_w = 0.95047$, $Y_w = 1$, $Z_w = 1.08883$

Following normalization, the L^*a^*b values are computed using specific formulas that account for the non-linear perception of colour by the human visual system.

$$L = 116 * f(Y_n) - 16 \quad \text{Equation 2.3}$$

$$a^* = 500 * (f(X_n) - f(Y_n)) \quad \text{Equation 2.4}$$

$$b^* = 200 * (f(Y_n) - f(Z_n)) \quad \text{Equation 2.5}$$

$$\text{Where } f(t) = \begin{cases} t^{1/3} & \text{if } t > 0.008856 \\ 7.787 * t + 16/116 & \text{otherwise} \end{cases}$$

2.3.2 Image Normalisation

Image normalisation is another method used in pre-processing the fundus images. It helps in standardising the data to a specific range, for example [0,1] (Singh Sisodia et al., 2017). This helps in avoiding bias or increased training times for the classification system. The normalised pixel value p' can be calculated based on the following formula:

$$p' = \frac{p - p_{min}}{p_{max} - p_{min}} \quad \text{Equation 2.6}$$

Where p is the original pixel value and p_{min} and p_{max} represent the minimum and maximum values in the image.

2.3.3 Gaussian Filtering

Noise removal is one of the most frequently used steps during fundus image enhancement methods to remove any noise or artefacts present in the image. Gaussian filter is typically employed for noise removal in fundus images. The process involves performing convolution of a gaussian kernel with the input image to smooth out the noise while

preserving the details. The standard deviation of the gaussian kernel dictates the strength of the filtering operation.

2.3.4 Image Resizing and Thresholding

Fundus images available in various datasets come in varying resolution and sizes. Another important part in preparing the fundus images for machine learning algorithms is to extract the retinal area and resize the image to a standard size. Thresholding is one of the most effective methods to separate foreground and background pixels. Using thresholding, the retinal structure can be extracted from varying size fundus images and resized to a standard size for further processing. Otsu's thresholding (Otsu, 1979) finds the optimum threshold value that separates the foreground and background pixels in an image. This is achieved by minimising the intra-class variance or maximising the inter-class variance between the foreground and background intensity distributions. The process of Otsu's thresholding can be summarised in the following steps:

- Compute and normalise the histogram of pixel intensities of the image. This gives the Probability Density Function (PDF) of the pixel intensities.
- Calculate the inter-class variance between foreground and background using the Equation 2.7.

$$\sigma_b^2(t) = w_1(t)w_2(t)[\mu_1(t) - \mu_2(t)]^2 \quad \text{Equation 2.7}$$

Where $w_1(t)$ and $w_2(t)$ are the probabilities of the two classes i.e., foreground and background pixels for a given threshold t .

$\mu_1(t)$ and $\mu_2(t)$ are the mean intensities of the two classes i.e., foreground and background pixels for a given threshold t .

- Select the threshold value that maximises the inter-class variance.

Otsu's global thresholding approach maximises the variance between foreground and background pixels. The varying intensity distribution of fundus images, non-uniform illumination, and changing contrast between retinal and non-retinal areas makes Otsu's thresholding a good choice for segmenting retinal area. Furthermore, the automated and adaptive selection of threshold value using the Otsu's method makes it ideal for pre-processing large datasets.

2.3.5 Data Augmentation

Data augmentation is the process of artificially expanding the training dataset for machine learning algorithms by applying different transformation techniques on the input. This process is particularly useful for unbalanced datasets where the data in the training samples for each class do not match. For image data augmentation, various transformation techniques are used.

- **Rotation:** The process adjusts the image by a specific angle which helps in providing different perspectives of the same image.
- **Flipping:** This creates mirror images of the input horizontally or vertically. This helps in generating models that are insensitive to orientation changes of the input.
- **Scaling:** Scaling is another technique used to modify images to concentrate on a specific area of the image. This helps in mimicking the draw distance of the camera used to capture fundus images.
- **Translation:** Translation allows the images to move along the horizontal or vertical axis that helps simulate the eye placement alterations.

- **Brightness and Contrast Adjustments:** Light condition variations can be simulated in data augmentation techniques by varying the brightness and contrast of the input images. This helps in model to handle images taken under varying light conditions.
- **Noise Injection:** Another method of augmentation is to introduce noise that simulates flaws in image capture.
- **Colour Jittering:** Various colour tones during image capture can be emulated by colour jittering during the augmentation process.

Most of the research work done in image enhancements for fundus images focuses on three fundamental approaches i.e., histogram-based enhancement, colour space transformations, and filters-based approaches (Zhou et al., 2018a). Gupta & Tiwari (2019) introduced a novel preprocessing methodology targeting fundus images for diabetic retinopathy detection. Their technique involves an enhancement process utilising adaptive gamma correction specifically applied to the luminosity channel in the L^*a^*b colour space. This enhancement is achieved by calculating the weights derived from the cumulative distribution of the histogram of input image pixels. To further refine the contrast enhancement within the L^*a^*b space, they incorporated a quantile-based histogram approach. Jawad et al. (2022) employed CLAHE on the normalised luminosity channel within the L^*a^*b colour space after segmenting the retinal region. Afterwards, the luminosity channel undergoes integration with the remaining two chromatic colour channels inherent to the L^*a^*b colour space. Zhou et al. (2018) enhanced the luminosity and contrast of fundus images by the utilisation of a luminance gain matrix. This matrix is derived from gamma correction applied to the value channel within the HSV (Hue, Saturation, and Value) colour space. Subsequently, this gain matrix is employed to enhance the individual RGB channels accordingly. Afterwards, contrast enhancement is performed on the luminosity

channel within the LAB colour space utilising CLAHE. Singh et al. (2019) addressed the challenge of non-uniform illumination in fundus images by proposing two radiance-based histogram equalisation methods, namely RIHE-RRVE (Radiance Indicator-based Histogram Equalisation for Recursive Retinal Vessel Enhancement) and RIHE-RVE (Radiance Indicator-based Histogram Equalisation for Retinal Vessel Enhancement), designed for enhancing retinal vessel visibility under bad illumination. One of these methods employs a recursive algorithm while the other is non-recursive in nature. A key aspect of these approaches involves estimating a tuneable parameter to partition the histogram into sub-bands and determine the radiance value. Subsequently, if the radiance value falls below a certain threshold, the histogram equalisation technique is applied accordingly. The performance of these methods was evaluated across various datasets, including DRIVE, STARE, and CHASE, utilising metrics such as entropy, PSNR (Peak Signal-to-Noise Ratio), Euclidean measure, and visual quality inspection to gauge performance. The work of Wang et al. (2021) propose the pre-processing of fundus images by a multi-layer decomposition comprising of base, detail, and noise layers. Afterwards, a visual adaptation model is proposed to address non-uniform illumination in fundus images. This model incorporates several key steps: utilising a luminance map extracted from the base layer for illumination correction, employing weighted fusion techniques to enhance details within the detail layer, and applying denoising methods to the noise layer. To summarise, most of the enhancements for retinal fundus images mainly focus on working with the image contrast, colour space transformation and filters to remove noise from the fundus images.

2.4 Features Extraction

Features are distinct class dependent patterns in the data that are extracted through various techniques to help the machine learning algorithm classify data into different classes.

In the past, the fundus images were pre-processed, and feature were extracted using various image processing methodologies. With the availability of powerful hardware, convolutional neural networks, which is a branch of machine learning modalities, has become widely popular for feature extraction. In this section, we discuss the research work that is reported in literature for extracting features from the fundus image of the eye.

2.4.1 Traditional Feature Extraction Methods

Conventional approaches of extracting important features from the input fundus image employ different methods to acquire local and global features of the fundus images. Local feature extraction methods extract minor patches within an image and provide information about edges and minor details. These local features can be combined to provide a global perspective of the image that helps in different classification tasks. Local Binary Pattern (LBP) and Local Ternary Pattern (LTP) (Chetoui et al., 2018b) are used in literature to extract texture-based features that are fed to ML classifiers for separating the output classes. LBP observes the neighbourhood around a pixel value and applies thresholding to the surrounding pixel values based on the central pixel. The binary code of the patch is used as the image descriptor (Tan & Triggs, 2010). LTP improves on the LBP by introducing three valued codes by adding a fixed threshold constant to the centre pixel. Local Energy-based Shape Histogram (LESH) is another technique used to extract local features from fundus images (Wajid & Hussain, 2015). The method finds areas of an image having maximum local frequency components in terms of phase and magnitude and take the histogram of that region. Image phase is obtained by convolution of image with Gabor filter. This gives magnitude and phase of every pixel for given scale and magnitude. The feature vectors generated can be used as input for image classifiers. A revised version of LBP called Neighbour Mean – Local Binary Pattern (NM-LBP) and Global Mean – Local Binary Pattern

(GM-LBP) is used by Abdillah et al. (2017). NM-LBP uses the mean of neighbourhood pixels as the thresholding values whereas the GM-LBP uses the global mean of the image to extract features. Fusion of textural features from fundus images is another technique used by Ramasamy et al., (2021). The approach involves extracting and combining textural features using techniques such as Gray-Level Co-occurrence Matrices (GLCM), Run-Length Matrices, and the Ridgelet Transform. The proposed method achieves 98.87% sensitivity, 95.24% specificity, and 97.05% accuracy on the DIARETDB1 dataset, and 90.9% sensitivity, 91.0% specificity, and 91.0% accuracy on the KAGGLE dataset. Most of the works in traditional feature extraction methods achieve reliable performance on small datasets but is not as good compared to modern CNN based approaches.

2.4.2 CNN based Feature Extraction

A Convolutional Neural Network is a type of deep learning network that is mainly used for processing image data. A fully connected neural network cannot capture the spatial features of an input image. Therefore, in CNN, a convolution operation is performed on the image using various filters (each filter captures a particular image feature, e.g., edge, smoothness, brightness, etc.) to create a feature map. The last layer in CNN contains a reduced feature set that is fully connected to an artificial neural network for classification. There are different methods available for extracting the feature vectors from existing CNN models. Some of these include updating the architecture, retraining the model, or freezing some layers of the model to utilise the pre-trained weights. Gangwar & Ravi (2021) use Inception and Residual Networks (ResNet) models of CNN to extract features from the input images and build a hybrid model to classify images on the Messidor-1 and Asia Pacific Tele-Ophthalmology Society (APTOS) dataset. A test accuracy of 72.33% and 82.18% is reported on the Messidor-1 and APTOS dataset respectively. Farag et al. (2022) propose a

deep-learning-based model that uses the DenseNet encoder and convolutional attention module block for DR severity detection. The encoder is used to extract the features from the input fundus images from the APTOS dataset and the attention block is used for refining the features. The authors achieved a binary classification accuracy of 97.00% and multiclass classification accuracy of 82.00%. Another study by Vives-Boix and Ruiz-Fernández (2021) introduces artificial synaptic meta plasticity into the initial learning stages of different CNNs for enhancing the feature extraction by the CNN models. They achieved an average accuracy of 94.00% on binary classification of APTOS dataset. Image concatenation of left and right fundus images of the eyes is used by Gour & Khanna (2021) to extract feature vectors with the help of different CNN models. These features are then passed on to a neural network classifier for categorising the fundus images. VGG16 model of the CNN architectures provide the best performance from all the tests that are reported with a maximum accuracy of 89.06%. Another work by Li et al. (2017) uses feature extraction with the help of VGG16 and GoogleNet models. The output of features extracted from these models is used to train a SVM classifier to predict the DR classes. The authors of (Padmanayana & B.K, 2022) developed a web interface with a CNN trained model that extracts the features from the fundus images from the APTOS dataset for binary classification. They achieved a maximum classification accuracy, sensitivity, and specificity of 94.60%, 86.00%, and 97.00% respectively.

2.4.3 Transfer Learning based Feature Extraction of Fundus Images

Transfer Learning (TL) (Olivas et al., 2009), is another field of deep learning in which learned features from one application model are transferred to a different domain (Shin et al., 2016). With the availability of powerful hardware having high processing power and large image datasets, different CNN models like VGG16, AlexNet, GoogleNet, ResNet

etc., have become widely popular in the feature extraction and classification of images. These models are usually trained on approximately one million images and can classify images into 1000 different categories. However, datasets of fundus images of the eye are not available in this substantial quantity. This is where transfer learning provides its usefulness by utilising the pre-trained models of VGG16, AlexNet, GoogleNet, and ResNet etc. These pre-trained networks are fine-tuned according to the target domain by updating weights of some specific layers or all layers of the network. The general workflow of transfer learning is shown in Figure 2.4.

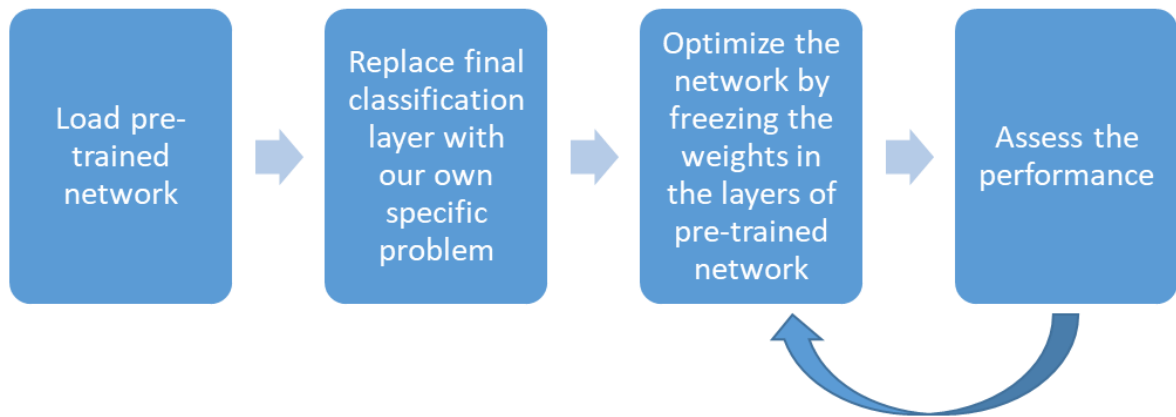


Figure 2.4: General Workflow in Transfer Learning

Gangwar & Ravi (2021) use TL by adding CNN layers on top of ResNet and Inception-based models for multiclass classification of fundus images from the APTOS blindness detection dataset. The images are pre-processed using resizing, blurring, and bounding box operations while data augmentation is performed to balance the data. The authors report a test accuracy of 82.18% on the APTOS dataset. Gour & Khanna (2021) proposed a multiclass classification approach for various eye-related diseases. The method utilises CNN architectures and TL to classify fundus images into different categories of ocular diseases. The Ocular Disease Intelligent Recognition dataset (ODIR), provided by

Peking University, contains class labels for eight categories of ocular diseases: Normal, Diabetic, Glaucoma, Cataract, Hypertension, Myopia, AMD, and other diseases. Two models were proposed by the authors using TL. The first model utilises right and left fundus images of the eye to establish a parallel architecture, whose feature vectors are combined prior to applying the pooling layer at the end. The second architecture employs a concatenated image of the right and left eye of the input for classification. The results demonstrate superior performance with the concatenated image input of the second model when employing TL on the VGG16 CNN architecture. Zhang et al. (2022) designed a source free transfer learning model for the binary classification of DR images. The model achieved a 91.20% accuracy on the APTOS dataset. The authors in (Saini & Susan, 2022a) applied transfer learning on three datasets (Kaggle DRD, IDRiD and DDR) to extract features and perform multi-class classification of DR images. They achieved a maximum accuracy of 79.10% with the DenseNet121 model on image resolutions of 4288×2848 from the IDRiD dataset.

2.5 Fundus Image Classification

Image classification is the core part of detecting DR from fundus images of the eye. The features extracted from the fundus images are passed on to different machine learning classifiers. Classification is a type of supervised machine learning that is used to guess the accurate label for input data. First the model is trained by training set of data. The trained model is afterwards tested using the test data to make predictions. Various ML algorithms are used in the literature to classify the fundus images.

2.5.1 Fundus Image Classification using Machine Learning

Traditional approaches of image classification involve extracting feature vectors from input data and passing the information to ML classifiers.

Support Vector Machines is a classification approach that is based on finding a hyperplane (decision boundary) that maximises the margin between the closest data points from different classes, known as support vectors (Haque et al., 2019). Suppose you have a dataset with input vectors X and corresponding labels y , where X is of size $m \times n$ (m samples, n features), and y is a vector of size m . The goal is to find a hyperplane defined by a weight vector w and a bias term b such that the hyperplane effectively separates the data into two classes and the margin between the two classes is maximised. The decision function for a linear SVM is given by:

$$f(x) = \text{sign}(w \cdot x + b) \quad \text{Equation 2.8}$$

where, w is the weight vector, x is the input vector, and b is the bias term. The optimisation problem tries to maximise the distance between the classes by minimising the weight vector subject to following constraints:

$$\min \left(\frac{1}{2} \|w\|^2 \right) \quad \text{Equation 2.9}$$

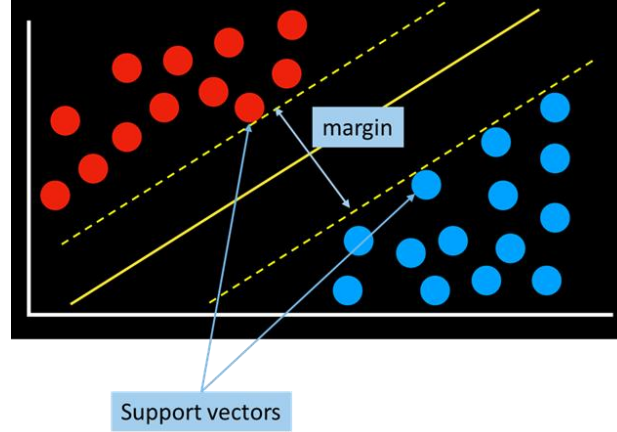


Figure 2.5: Working of Support Vector Machines (Lu & Malan, 2023)

Multiple parameters within the Support Vector Machine (SVM) framework can be adjusted to optimise model training. The regularization parameter (C) controls the balance between maximising the margin and minimising misclassification errors, with higher values penalizing misclassifications more heavily but potentially leading to overfitting. The choice of kernel, such as Linear, Radial Basis Function (RBF), Polynomial, or others like Sigmoid or Laplacian, defines the similarity measure between data points and shapes the decision boundary accordingly. The gamma parameter for RBF kernel influences the localization of decision boundaries, while the degree parameter for Polynomial kernel determines the complexity of relationships captured, with higher degrees risking overfitting. Additionally, class weights can be assigned to address imbalanced datasets, where higher weights are allocated to underrepresented classes to improve learning effectiveness.

Random Forests (RF) is an ensemble learning algorithm that combines the predictive power of multiple decision trees to achieve robust and accurate results (Latif et al., 2020). Given training data consisting of N data points, each with d features: $(x_1, y_1), \dots, (x_N, y_N)$, where $x_i \in \mathcal{R}^d$ and $y_i \in Y$. For each of the M trees in the forest, it randomly samples m features ($m < d$) without replacement. Then for each internal node it randomly selects a

subset of the m features. The best split among the selected features is then chosen based on an impurity measure (such as Gini index for classification). It then splits the data based on the chosen split point. The steps are repeated until a stopping criterion is met.

The Multilayer Perceptron (MLP) represents a fundamental artificial neural networks-based architecture (Harun et al., 2019). It is characterized by a layered structure that consists of an input layer, one or more hidden layers, and an output layer. Each layer fully connects to the subsequent layer. MLPs leverage nonlinear activation functions in hidden layers that enables them to model complex relationships between inputs and outputs that are not linearly separable. Common activation functions include ReLU, sigmoid, and tanh. MLP models are trained using the backpropagation algorithm, which propagates the error signal backwards through the network, adjusting the weights and biases of connections to minimise the loss function.

Naïve Bayes (NB) is yet another traditional classifier that is based on the probabilistic statistics model of the Bayes theorem (John & Langley, 2013). The assumption that strong independence exists between the features of the images gives this classifier the name of naïve. In the original Bayes classifier, the conditional probability of whether data belongs to a particular class is calculated through the conditional and unconditional probabilities of the same data belonging to each class within the dataset. The complexity of NB is finding the class within the data that has the same number of attributes with strong dependence.

Sreng et al. (2018) use different ML algorithms (kNN, SVM, and RF) in their work to classify the input fundus images. The features are extracted using variations of LBP from local dataset consisting of 1200 funds images. SVM algorithm outperforms other ML

algorithms with highest classification accuracy of 97.00%. Colour transformation and vessel segmentation techniques are used by Carrera et al. (2017) for feature extraction. The SVM and decision tree classifier is used for comparing the classification performance on the Messidor dataset. The SVM achieves an accuracy of 92.40%. Similar classification accuracy of 92.40% is observed by Suryawanshi & Setpal (2017) using SVM on features extracted from fundus images using the Gray Level Co-occurrence Matrix (GLCM). Abdilllah et al. (2017) in their research use the SVM and k-Nearest Neighbour (kNN) classifiers on DirectDB dataset for DR classification. Variations of LBP are used to extract features that are passed on to the classifiers for training. The results show a 90.00% AUC for binary classification and 68.00% for four class classification.

2.5.2 Fundus Image Classification using Deep Learning

Deep learning methodologies are based on the fundamental principles of the Artificial Neural Network (ANN). The structure of these networks is inspired by the collective working of neurons inside a human (Rosenblatt, 1958). The fundamental element in an ANN is the perceptron whose output we can calculate using Equations 2.10 and 2.11.

$$y = f\left(w_0 + \sum X^T * W\right) \quad \text{Equation 2.10}$$

$$y = f(z) \quad \text{Equation 2.11}$$

$$\text{where } z = w_0 + \sum X^T * W$$

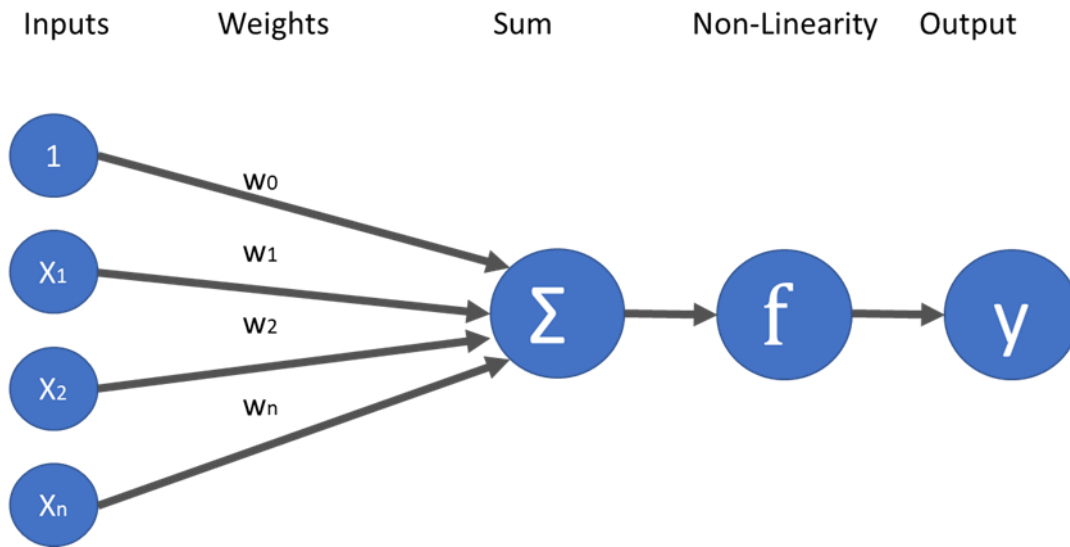


Figure 2.6: Working of a Perceptron in an Artificial Neural Network

In the equations above, X is the input, W is the weight matrix, w_0 is the bias, and f is the non-linear function. An ANN consists of multiple neurons whose weights are trained to predict the output from the given input. The working of a perceptron is illustrated in the Figure 2.6. Another important factor required for the accurate training of the network is the large number of training inputs that can help capture the features of the input. In a multilayer ANN, several layers each containing a collection of perceptron's are used. The layers of perceptron's are called the hidden layers. A densely connected neural network with one hidden layer is shown in Figure 2.7. A deep neural network consists of multiple hidden layers and needs abundant input data to accurately train the deep network to learn the features of the input.

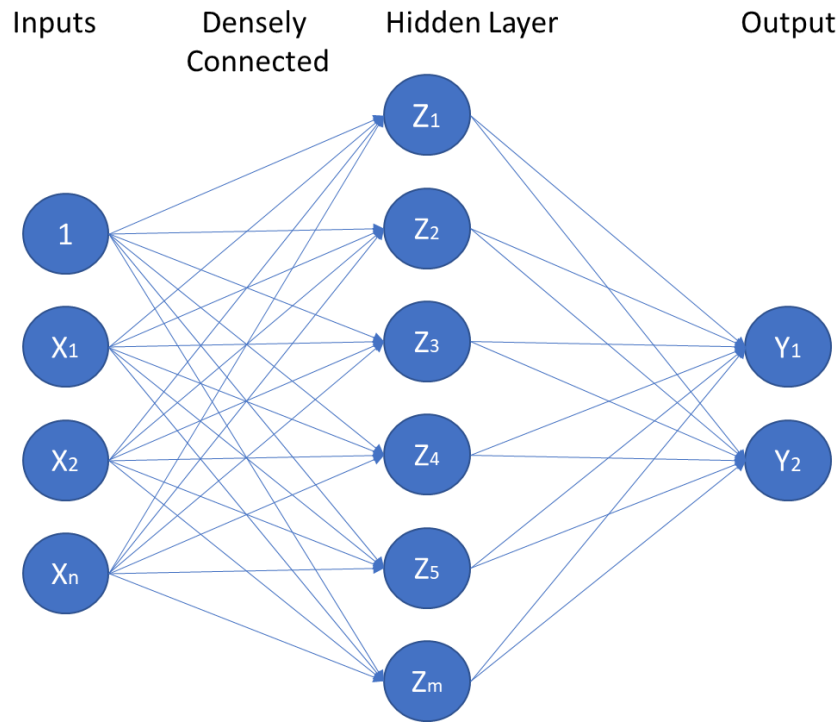


Figure 2.7: Neural Network with Single Hidden Layer

A Convolutional Neural Network (CNN) is a type of deep learning network that is mainly used for processing image data. A fully connected ANN cannot capture the spatial features of an input image. Therefore, in CNN, a convolution operation is performed on the image using various filters (each filter captures a particular image feature, e.g., edge, smoothness, brightness, etc.) to create a feature map. Non-linearity is introduced in the activation layer using a Rectified Linear Unit (ReLU) operator (Nair & Hinton, 2010). The hidden layers of an ANN are replaced with convolution layers. The convolution layers capture diverse image features. Pooling is performed to reduce the dimensions of the input image. In the last stage, the image with the reduced feature set is flattened, and a fully connected layer is used to predict the output classes. The workflow in a CNN is described in Figure 2.8.

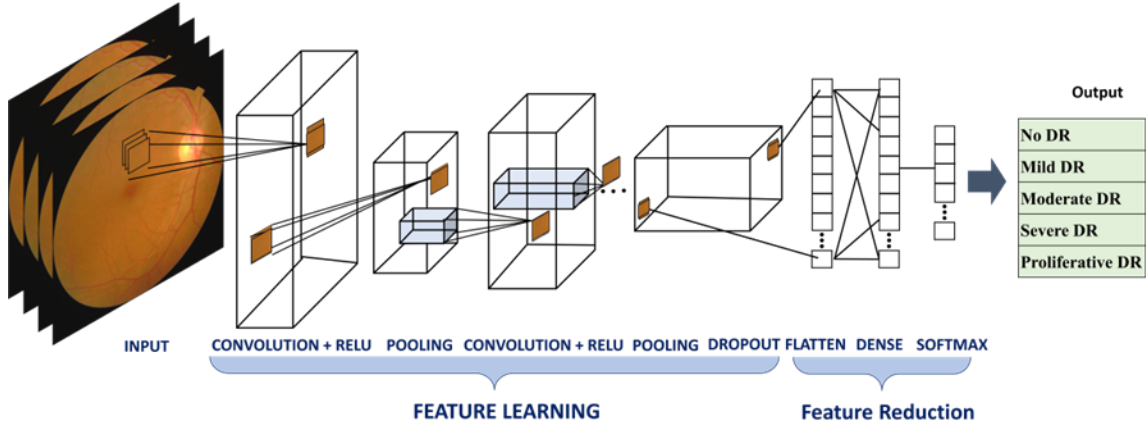


Figure 2.8: Workflow in a Convolutional Neural Network

Alyoubi et al. (2021) in their work present a custom CNN512 and YOLOv3 deep learning model using the DDR and APTOS datasets and worked on fixed sized images of resolution of 512×512 . With this approach which also involves data augmentation, they achieve a classification accuracy of 89.00%. Some novel approaches have harnessed the optimisation characteristics of bio-inspired optimisation algorithms. One such paper by Elwin et al. (2022), uses Autoregressive-Henry Gas Sailfish Optimisation (Ar-HGSO) on the augmented DDR and IDRiD datasets, but resizing the images to a resolution of 256×256 for training their models. They achieve a multiclass classification accuracy of 91.40% for the Indian Diabetic Retinopathy Image Dataset (IDRiD). Yet another recent approach uses the Visual Transformer and Residual attention models for classifying DR by achieving an AUC metric value of 90.00% (Gu et al., 2023). They have used the image resolution of 512×512 from the DDR and IDRiD datasets, but also perform a comparison with 5 other pre-trained CNN models. A coarse-to-fine CNN architecture is proposed by Wu et al. (2020) that first uses a coarse network to perform binary classification of the input data into No DR and DR affected images. The architecture introduces attention gate modules into the CNN architecture that reduces background information and enhances the lesion features. The Fine Network later classifies the remaining four stages of DR, i.e., mild, moderate, severe, and

proliferative DR from the DR classified images of the coarse network. The datasets used in the paper are the EyePACS and the IDRiD. The IDRiD dataset contains images acquired from Kowa VX-10 α digital fundus camera from an Eye Clinic in India). The model achieved a maximum accuracy of 83.10% on the EyePACS dataset and 56.19% on the IDRiD dataset. Rakhlin (2017) uses deep CNN for the binary classification of the retinal fundus images. The images are grouped into two classes i.e., no DR and referable DR. The referable DR groups the images from stages 1–4 of the ICDR severity scale. A CNN with nineteen layers is trained for extracting image features and classification. The proposed CNN is a modified version of the VGGNet CNN architecture.

2.5.3 High Resolution Fundus Image Classification using CNN

CNN (Convolutional Neural Network) models are extensively used for image and video processing tasks, such as object detection, image classification, and segmentation. In certain application domains (such as medical imaging and remote sensing) CNN models can provide better accuracy with higher resolution input images (Thambawita et al., 2021). Medical imaging, for instance, relies heavily on precise identification of subtle anomalies. High-resolution scans in fields like radiology or pathology expose minute details invisible at lower resolutions. CNNs trained on such high-resolution images can discern nuanced tissue textures, variations in density, and other critical features crucial for accurate diagnoses. This can be seen in Figure 2.9 that the level of details perceptible in the fundus image increase with higher resolution.

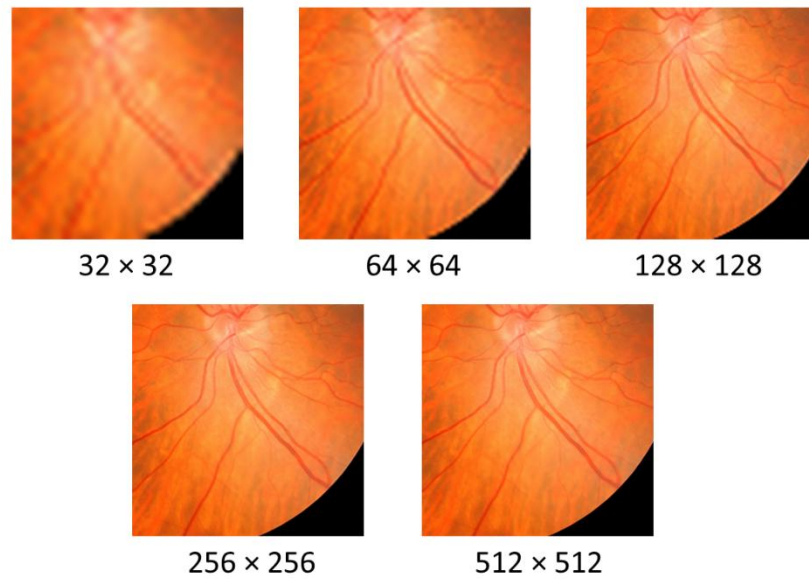


Figure 2.9: Level of Details in Fundus Images at Varying Resolution

The lower resolution images can capture general object shapes; however, higher resolutions offer a wealth of local details, often crucial for accurate classification. It is known that CNN models tend to rely mostly on local features rather than global features for classification purposes (Baker et al., 2020). Local features provide several advantages over global features. Firstly, local features capture more detailed and distinctive information about specific regions of an image. Global features, on the other hand, treat the entire image as a single entity, and do not differentiate between different regions of the image that accurately. This can result in loss of important information about the image that could be critical for accurate classification. Secondly, local features are more robust to variations in the image, such as changes in lighting, rotation, and scale. This is because local features are computed based on small patches of the image and are therefore less affected by changes in the overall appearance of the image. Global features, on the other hand, may not be as robust to such variations, as they consider the entire image. Lastly, local features can be combined in a spatially aware manner to capture the overall structure and layout of the image. This can

be especially useful for tasks such as object recognition and scene understanding, where the spatial arrangement of objects in the image is important. Overall, local features are a powerful tool for image classification because they capture fine-grained, distinctive information about specific regions of an image, while also being more robust to variations and allowing for spatially aware feature combinations. In general, higher resolution fundus images can provide better details and features that can help in distinguishing DR classes having inter-class similarities. On the other hand, the increase in input image dimensions for CNN based models substantially increases the number of trainable parameters and thus requires a significant amount of memory to store and process data. This results in a higher computational cost (increased time complexity) and require larger memory (increased space complexity). There have been several attempts at mitigating time complexity aspect of the model training using data and model parallelism (Ben-Nun & Hoefler, 2019). A critical constraint in working with deep learning models having a large number of parameters and with higher resolution input images is the limited GPU memory. A popular approach for reducing space complexity is to reduce model complexity with a reduced number of parameters such as MobileNet, SqueezeNet, and EfficientNet. However, these memory efficient models can still result in larger memory requirements for processing higher resolution images. Saini and Susan (2022) applied 16 pre-trained CNN models on three datasets (Kaggle DRD, IDRiD and DDR) to perform multiclass classification of DR images. They achieved a maximum accuracy of 79.10% with the DenseNet121 model on image resolutions of 4288 x 2848 from the IDRiD dataset. Zago et al. (2020) have used VGG16 and ImageNet models on 7 different datasets (DDR, IDRiD, Kaggle, Messidor, Messidor2, DIARETDB0 and DIARETDB1) by leveraging the patch-based method to achieve a ROC value of 91.20%. The original image sizes were 512 x 512, and the patch size was empirically

chosen as 65 pixels. Similarly, Li et al. (2019) used eight different pre-trained models (namely, VGG16, ResNet18, GoogleNet, DenseNet-121, Inception, SSD, YOLO and RCNN) to work on the DDR dataset with image resolutions of 512 x 512. The maximum classification accuracy of 76.59% was reported for the ResNet18 model.

To summarise, modern approaches for DR detection using CNN follow two fundamental rules. If the input data is large, a custom CNN can be trained to detect and identify various stages of DR. This takes a long processing time to train the model, extract features and classify the image. However, for smaller datasets that do not have enough information to fully train the CNN, transfer learning is utilised. Transfer learning can speed up the training process and provide ample features for problems with smaller datasets. Some authors have also combined various stages of DR into either two or three class classification problems due to inter-class similarities that make the model easier to train. Furthermore, study shows that higher resolution fundus images provide better quality features that are able to extract useful information for multiclass DR classification but suffer from slow training time and higher memory requirements in the GPU for training the CNN models.

2.6 Convolutional Neural Networks Models

This section describes various CNN models that are used in literature for image classification tasks. CNN models mainly rely on balanced and large amounts of data to develop a stable model. Most of these CNN models are trained on ImageNet dataset (Deng et al., 2009) that classifies the input images into 1000 different categories. The general workflow of CNN is given in Figure 2.10. The steps involved in classification using CNN are given below:

- **Input Image:** The images usually pass through a pre-processing phase before being input to the CNN models. This can include various image enhancement techniques to improve the quality of input images.
- **Convolution Layers:** Unlike fully connected ANNs that treat each pixel independently, CNNs employ convolution. This involves sliding small matrices called filters (or kernels) over the input image. Each filter is designed to detect specific features such as edges, brightness, contrast etc. The filter performs element-wise multiplication with the corresponding portion of the image, and the results are summed to produce a single output value. This process is repeated as the filter slides across the entire image, creating a feature map (also called an activation map) as shown in Figure 2.11. Each convolutional layer consists of multiple filters, each generating its own feature map. These layers capture diverse image features at different levels of abstraction. Early layers might detect simple edges and corners, while deeper layers learn more complex features like shapes and objects.
- **Activation Function:** The main purpose of activation functions is to introduce non-linearity in the learning process. This non-linearity allows the network to learn more intricate relationships in the data.
- **Pooling:** Pooling is mainly used in CNN to reduce the dimensionality of the feature maps and decrease computational complexity. Max pooling is a common choice, selecting the maximum value in each block. This reduces the number of parameters and computations in the network, making it more efficient and less prone to overfitting.

- **Flatten:** After several convolutional and pooling layers, the feature maps are flattened into a one-dimensional vector. This vector represents the learned high-level features of the input image. These features can also be extracted out from a CNN and used as input to other ML classifiers.
- **Fully Connected Layer:** Finally, the flattened vector is fed into one or more fully connected layers. These fully connected layers perform the final classification by learning relationships between the high-level features and the output classes. A softmax function is typically used in the final layer to produce a probability distribution over the possible classes.

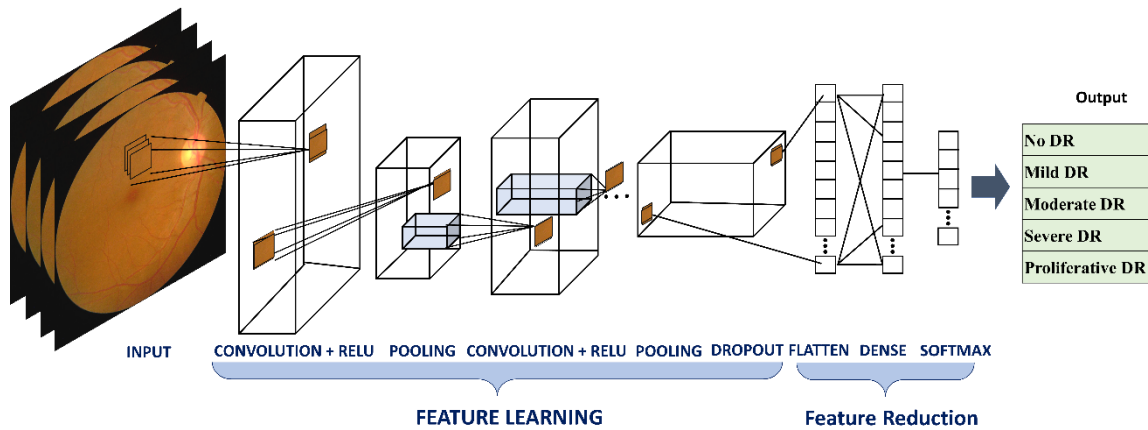


Figure 2.10: Working of CNN Models

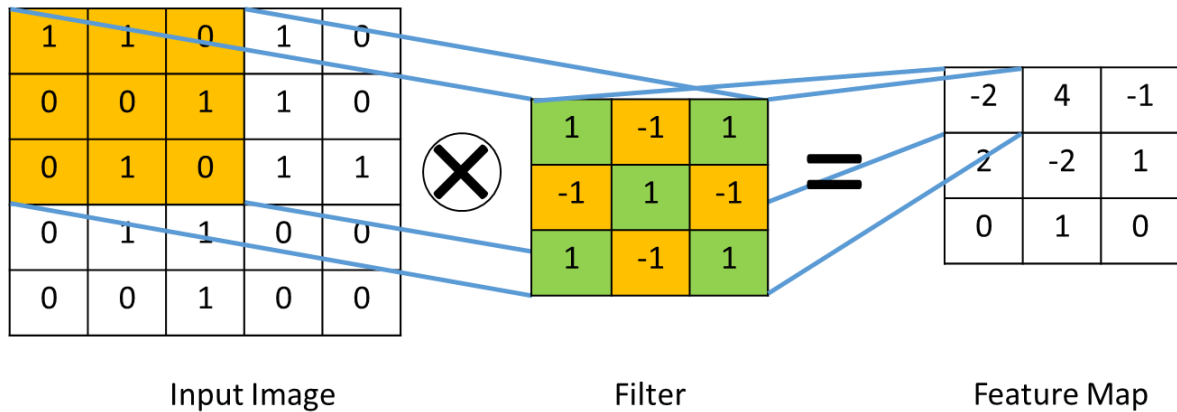


Figure 2.11: Convolution Layer in CNN

2.6.1 AlexNet

AlexNet is one of the very first CNN models that became popular during the ImageNet 2012 classification challenge. The network was able to outperform many traditional ML based approaches and paved the way for CNN being utilised for image classification tasks. Its architecture comprises of 60 million parameters and 650,000 neurons. The structure entails a total of eight layers, incorporating five convolutional layers alongside three fully connected layers. The fully connected and softmax layers are used to scale down the features and categorise the output into 1000 image classes. Each convolutional and fully connected layer undergoes ReLU activation. The input size of the image expected by the

network is $227 \times 227 \times 3$, with the initial convolutional layer applying filtering to the input image using 96 kernels with filter sized $11 \times 11 \times 3$ and stride of 4 pixels. Max pooling with filter size 3×3 and stride 2 is used to scale down the feature map size. The second convolutional layer uses 256 kernels with filter size $5 \times 5 \times 48$. No pooling or normalization layers are present between the third, fourth, and fifth convolutional layers. The fully connected layer at the end, consisting of 4096 neurons, is used to flatten the features extracted from the input. AlexNet garnered acclaim for winning the imageNet2012 classification competition, where it was able to classify 1.2 million input images into 1000 distinct output classes. An illustration of AlexNet's core architecture (Krizhevsky et al., 2012) is provided in Figure 2.12.

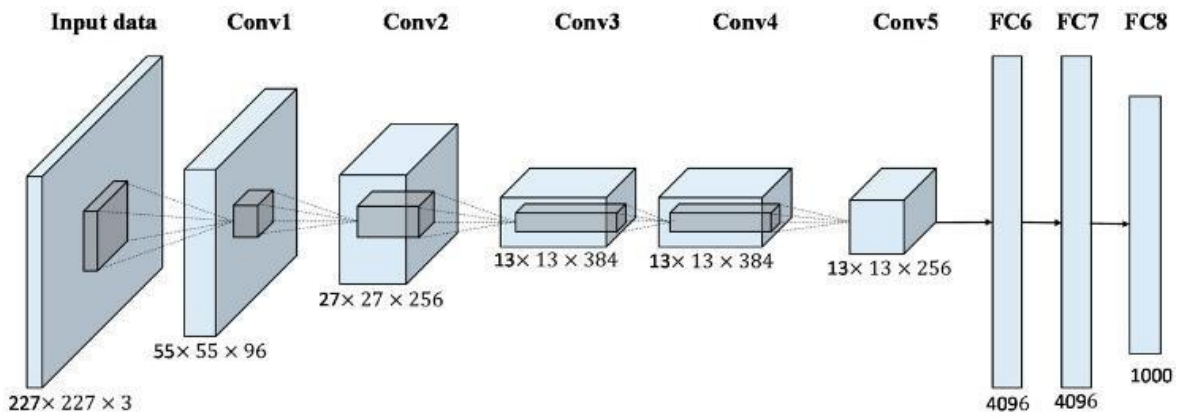


Figure 2.12: AlexNet Architecture for Image Classification (Butt et al., 2019)

2.6.2 GoogleNet

GoogleNet (Szegedy et al., 2014) is a CNN that is 22 layers deep and efficiently uses the computational resources using repeated inception modules. The size of input image expected by the network is $224 \times 224 \times 3$. The inception modules in GoogleNet enhance the width and depth of the network that helps capture the features at varying scales. Each inception module contains different-sized convolutional layers to capture various local and spatial features of the input. The inception module for the GoogleNet model is shown in

Figure 2.13. The 1×1 convolutional layers reduce the dimensions of the input and extract the local cross-channel features. The 3×3 and 5×5 convolutional layers help in capturing the spatial features of the input. The pooling layer is included in the inception module to reduce the dimensions of the input. The 22 layered architecture of GoogleNet that uses the inception modules is shown in Figure 2.14.

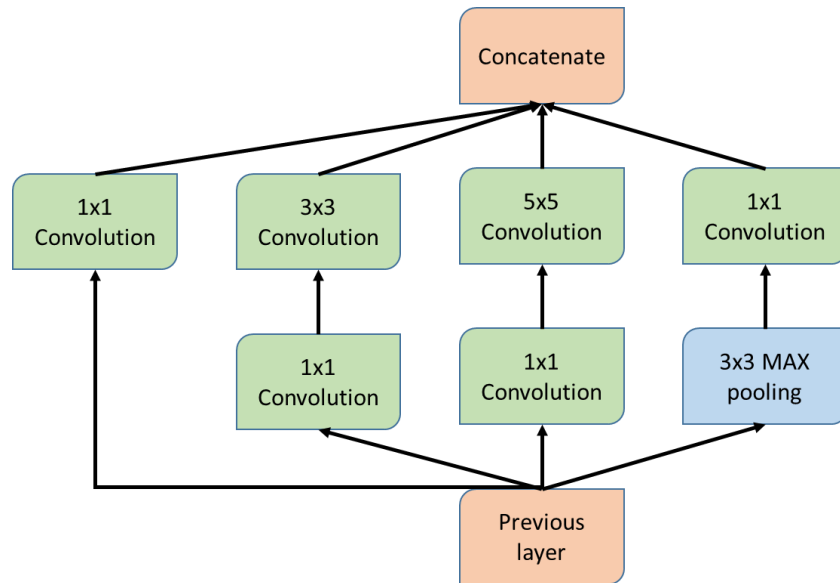


Figure 2.13: Inception Module of GoogleNet CNN (Butt et al., 2022)

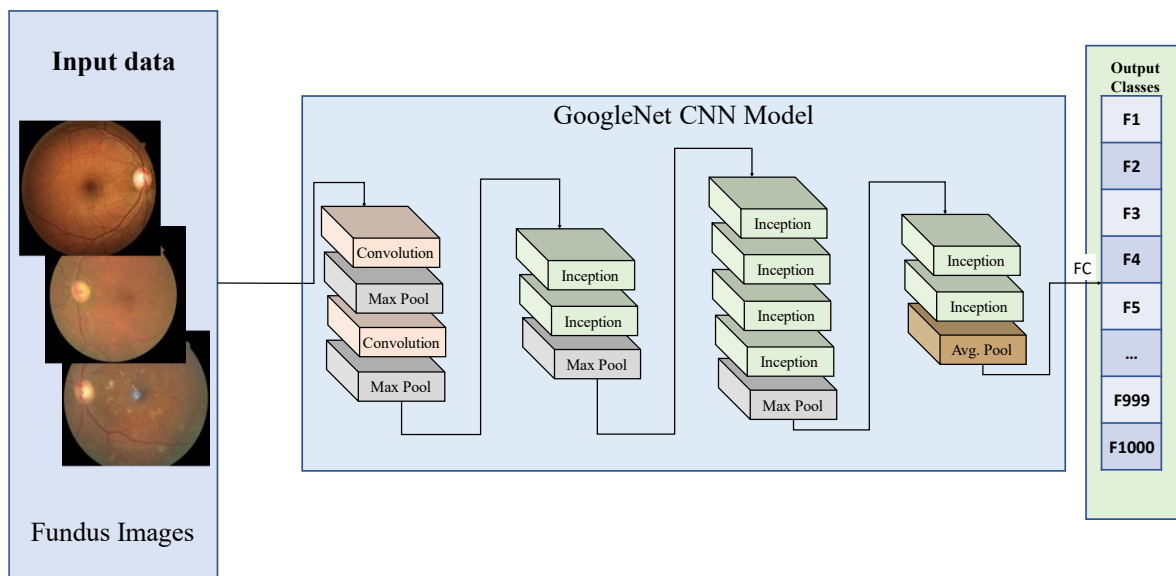


Figure 2.14: Architecture of GoogleNet utilising the Inception modules (Butt et al., 2022)

2.6.3 VGG16

The VGG16 model (Simonyan & Zisserman, 2014) won the ImageNet 2014 classification challenge. The authors proposed the use of small filter size of 3×3 and increased the depth of the network for input images of size $224 \times 224 \times 3$ to extract intricate features. The network does not use batch normalization in its layers which increases the time for training the model. The proposed architecture of VGG16 is represented in Figure 2.15.

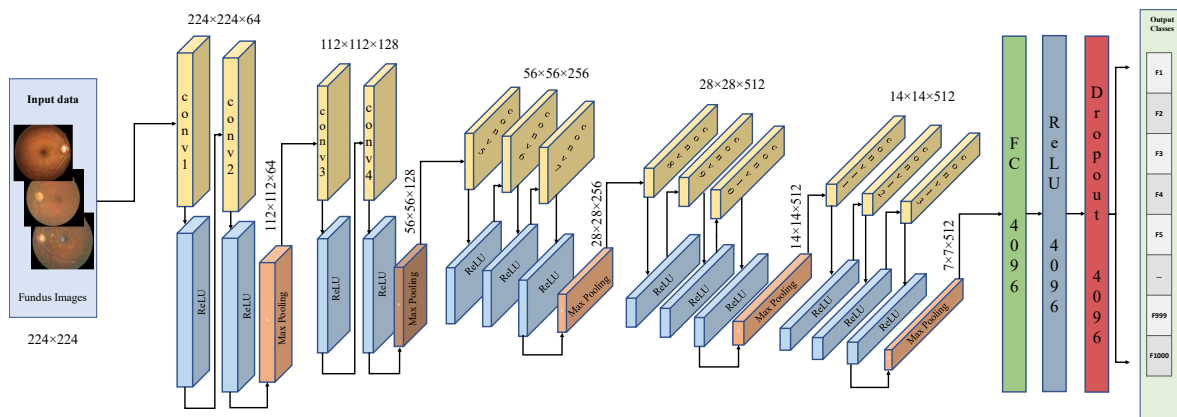


Figure 2.15: VGG 16 Architecture for Image Classification

2.6.4 ResNet

The Residual Neural Network (ResNet) (He et al., 2016) model won the ImageNet 2015 classification competition. The main breakthrough in ResNet CNN model is that it eliminates specific layers in the network using skip connections (Xu et al., 2023). These skip connections are also called residual or shortcut connections. The skip connections help solve the problem of vanishing gradients in CNN and reduce the training time. This is done by allowing the gradients to flow more efficiently during backpropagation. Non-linear activation functions are used between the skipped layers. Batch normalization is also applied between the shortcut connections. A weight matrix is used that calculates the weights of the

jump connections. After learning the features of the input, flattening is applied in the later stages of the network.

Figure 2.16 shows the basic building block of a ResNet. Multiple instances of the residual block are used throughout the network. In a CNN, the mapping from $x \rightarrow f(x)$ is learned. In the fundamental block of the residual network, the mapping is carried out by a feed-forward neural network that contains shortcut connections called jump or skip connections, i.e., $x \rightarrow f(x) + g(x)$. The function $g(x)$ is an identity connection if both the output and input dimensions match, otherwise, zero padding is applied.

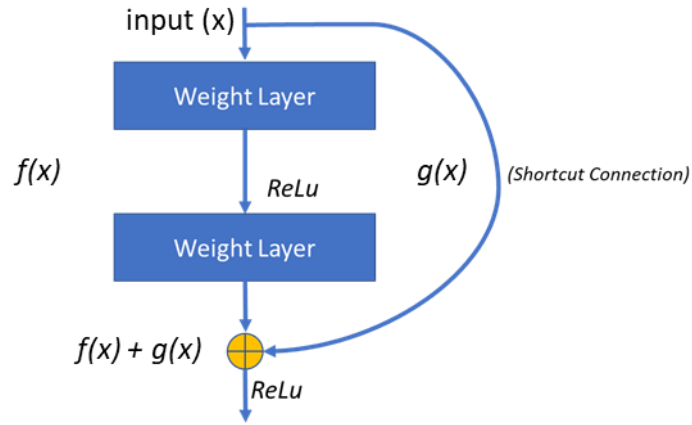


Figure 2.16: Skip Connections in ResNet (Butt et al., 2022)

The resulting residual block for the stacked layers in the network with the same dimensions can be given by Equation 2.12.

$$y = f(x, \{W_i\}) + x \quad \text{Equation 2.12}$$

The function $f(x, \{W_i\})$ represents the convolution layer mapping, which is learned during the training. There are different variations of ResNet models that were presented in (He et al., 2016) i.e., ResNet-18, ResNet-50, ResNet-201. The network accepts input images of size $224 \times 224 \times 3$. It uses 3×3 filters with a stride of 1, average pooling layer that

contains 1×1 filter, and one fully connected layer at the end. This is followed by a final SoftMax layer for classification. The ResNet-18 model, for example, contains a total of 17 convolutional layers with one fully connected layer at the end as shown in Figure 2.17.

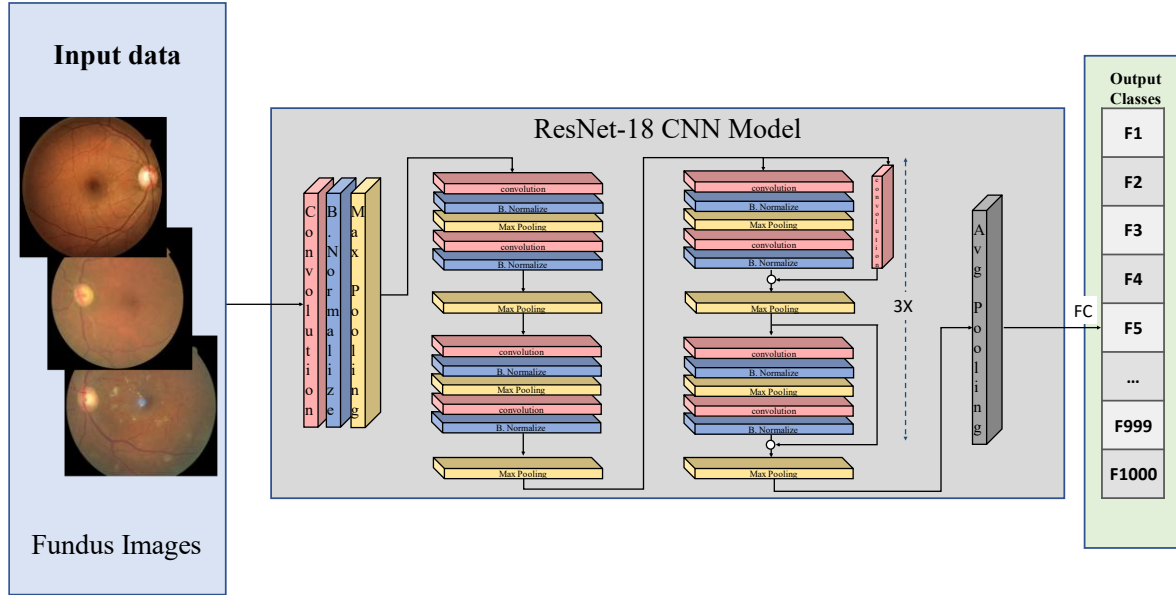


Figure 2.17: Architecture of ResNet-18 CNN Model (Butt et al., 2022)

2.6.5 EfficientNet

EfficientNet, is a CNN model that has garnered attention for its significant performance in balancing model efficiency and accuracy across various tasks (Tan & Le, 2020). EfficientNet introduces a novel scaling method that uniformly scales dimensions in network width, depth, and resolution, resulting in models that are both smaller and more accurate compared to conventional architectures. By systematically varying the network's depth, width, and resolution, EfficientNet achieves state-of-the-art performance while requiring fewer computational resources, making it particularly appealing for resource-constrained environments. The EfficientNet models, starting from B0 to B7, demonstrate superior performance on several benchmarks. The models are developed using a compound scaling method, which considers the relationship between the model's depth, width, and

resolution. This method is based on the observation that the three dimensions are not independent but have an intricate relationship that impacts the model's performance. For DR classification from fundus images, EfficientNet has shown promising results. Lazuardi et al. (2020) used EfficientNet-B4 and EfficientNet-B5 models for automatic detection of DR from retinal images, achieving quadratic weight kappa and accuracy scores of 0.7922 / 83.87% and 0.7931 / 83.89% respectively. Chetoui and Akhloufi (2020) also used EfficientNet for classifying DR images into five stages and localizing the affected lesions on the retina surface.

2.7 Evaluation Metrics

Different performance metrics like specificity, sensitivity, precision, and accuracy are used to compare the results with other methodologies of DR detection. These metrics are defined below.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad \text{Equation 2.13}$$

$$\text{Sensitivity / Recall} = \frac{TP}{TP + FN} \quad \text{Equation 2.14}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad \text{Equation 2.15}$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad \text{Equation 2.16}$$

$$\text{F - Measure} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad \text{Equation 2.17}$$

Where TP = True Positive (Number of positive images classified correctly), TN = True Negative (Number of negative images correctly classified), FP = False Positive (Number of negative images wrongly classified as positive), FN = False Negative (Number

of positive images wrongly classified as negative). **Accuracy** determines what fraction of total predictions that were correctly classified. **Sensitivity** / **Recall** determines what fraction of actual positive labels were predicted correctly as positive by the classifier. It is also called the True Positive Rate (TPR). **Specificity** (SP) determines what fraction of actual negative labels were predicted correctly as negative by the classifier. It is also called the True Negative Rate (TNR). **Precision** determines what fraction of predictions classified as positive are actually positive.

Confusion matrix also known as error matrix (Stehman, 1997) is tabular layout of all the measures discussed that helps in visualization of the performance of an algorithm. The confusion matrix for a binary classifier of DR is given in Table 2.1.

Table 2.1: Confusion Matrix for Binary DR classification

Actual Output (Label)	Predicted output (Binary classification)		
		NDR	PDR
	NDR	True Negative (Number of images correctly identified as having DR)	False Positive (Number of Images wrongly classified as having no DR)
	PDR	False Negative (Number of Images wrongly classified as DR)	True Positive (Number of images correctly identified as having no DR)

For multiclass DR classification, the confusion matrix is represented in Figure 2.18.

		Predicted Class				
		No DR (0)	Mild DR (1)	Moderate DR (2)	Severe DR (3)	PDR (4)
Actual Class	No DR (0)	TP for No DR				
	Mild DR (1)		TP for Mild DR			
	Moderate DR (2)			TP for MDR		
	Severe DR (3)				TP for Severe DR	
	PDR (4)					TP for PDR

Figure 2.18: Confusion Matrix for Multiclass Classification

The diagonal entries represent the true positive predictions of each class. The average accuracy can be calculated according to the following equation.

$$\text{Accuracy} = \frac{\text{Sum of Diagonal Entries (Total TP)}}{\text{Sum of all entries in the confusion matrix}} \quad \text{Equation 2.18}$$

		Predicted Class				
		No DR (0)	Mild DR (1)	Moderate DR (2)	Severe DR (3)	PDR (4)
Actual Class	No DR (0)	1	2	3	4	5
	Mild DR (1)	6	7	8	9	10
	Moderate DR (2)	11	12	13	14	15
	Severe DR (3)	16	17	18	19	20
	PDR (4)	21	22	23	24	25

Figure 2.19: Performance Metrics for Moderate DR (2) Class

From Figure 2.19, the per-class performance metrics can be calculated based on the following equations.

$$TP_{\text{Moderate DR (2)}} = \text{Cell}_{13} \text{ (Green)} \quad \text{Equation 2.19}$$

$$FP_{\text{Moderate DR (2)}} = \text{Cell}_{03} + \text{Cell}_{08} + \text{Cell}_{18} + \text{Cell}_{23} \text{ (Red)} \quad \text{Equation 2.20}$$

$$FN_{\text{Moderate DR (2)}} = \text{Cell}_{11} + \text{Cell}_{12} + \text{Cell}_{14} + \text{Cell}_{15} \text{ (Orange)} \quad \text{Equation 2.21}$$

$$TN_{\text{Moderate DR (2)}} = \sum \text{All remaining cells Cell (Blue)} \quad \text{Equation 2.22}$$

$$\text{Precision}_{\text{Moderate DR(2)}} = \frac{TP_{\text{Moderate DR (2)}}}{TP_{\text{Moderate DR (2)}} + FP_{\text{Moderate DR (2)}}} \quad \text{Equation 2.23}$$

$$\text{Recall}_{\text{Moderate DR(2)}} = \frac{TP_{\text{Moderate DR (2)}}}{TP_{\text{Moderate DR (2)}} + FN_{\text{Moderate DR (2)}}} \quad \text{Equation 2.24}$$

2.7.1 Macro and Weighted Averages

During evaluation of performance for different models, there are different ways to represent the average score of precision, recall, and f-measure. The macro average scores

are the regular method where the individual class scores are summed and divided by the total number of classes. In weighted average scores, the number of instances of each class are considered during the calculation of the average. If a DR class has low number of instances during the model testing, that class will have less impact on the weighted average of any performance evaluation parameter. The number of instances of each class represented as the score of that class can be used to calculate the weighted average. Equation 2.25 represents the example of calculating the weighted average precision for fundus images.

$$\text{Precision}_{\text{weighted}} = \sum_{i=0}^4 \text{Precision}_i * \left(\frac{\text{Score}_i}{\text{Total Instances}} \right) \quad \text{Equation 2.25}$$

Where i represents the classes 0-4.

$$\text{Precision}_{\text{Macro}} = \frac{\sum_{i=0}^4 \text{Precision}_i}{5} \quad \text{Equation 2.26}$$

In a balanced dataset having equal number of observations in each class, usually the macro and weighted averages yield same results. However, in unbalanced dataset, it is valuable to use the weighted averages to avoid a class of lower number of instances affect the overall result.

2.8 Summary

In this chapter, a review of recent research work in fundus image pre-processing, feature extraction, and classification is presented. While significant progress has been made in DR classification using deep learning, several research gaps remain, particularly in multiclass classification scenarios, hindering the development of robust and clinically applicable systems. This research addresses three key limitations identified in the current literature. Pre-processing plays a vital role in enhancing the quality of fundus images for

improved DR classification. While most of the research work uses CLAHE to enhance the quality of degraded fundus images, there is very limited work done on the use of individual channel variance and standard deviation of fundus images for enhancing the degraded images. Secondly, CNN based approaches have provided good performance but there is limited work done for combining features from multiple CNN and using traditional ML models for multiclass image classification. Lastly, using high-resolution images for multiclass DR classification offers better features which helps in improving the multiclass classification. However, processing high-resolution images is challenging due to high computational power and memory requirements. The work presented in this thesis will address this issue by acquiring high-resolution image features without putting constraint on the GPU memory.

CHAPTER 3

MATERIALS AND METHODS

3.1 Overview

In this chapter, we cover the methodology that is used for DR classification of fundus images. The overall workflow is presented in Figure 3.1. First, we cover the experimental environment in Section 3.2. Various datasets that are available for fundus images are covered in Section 3.3. Next, the proposed pre-processing framework for fundus images is presented in Section 3.4. Afterwards, in Section 3.5, the classification methodologies are proposed that extract useful features from the fundus images using deep learning methods. The classification of the extracted features using different ML classifiers is presented afterwards in Chapter 4.

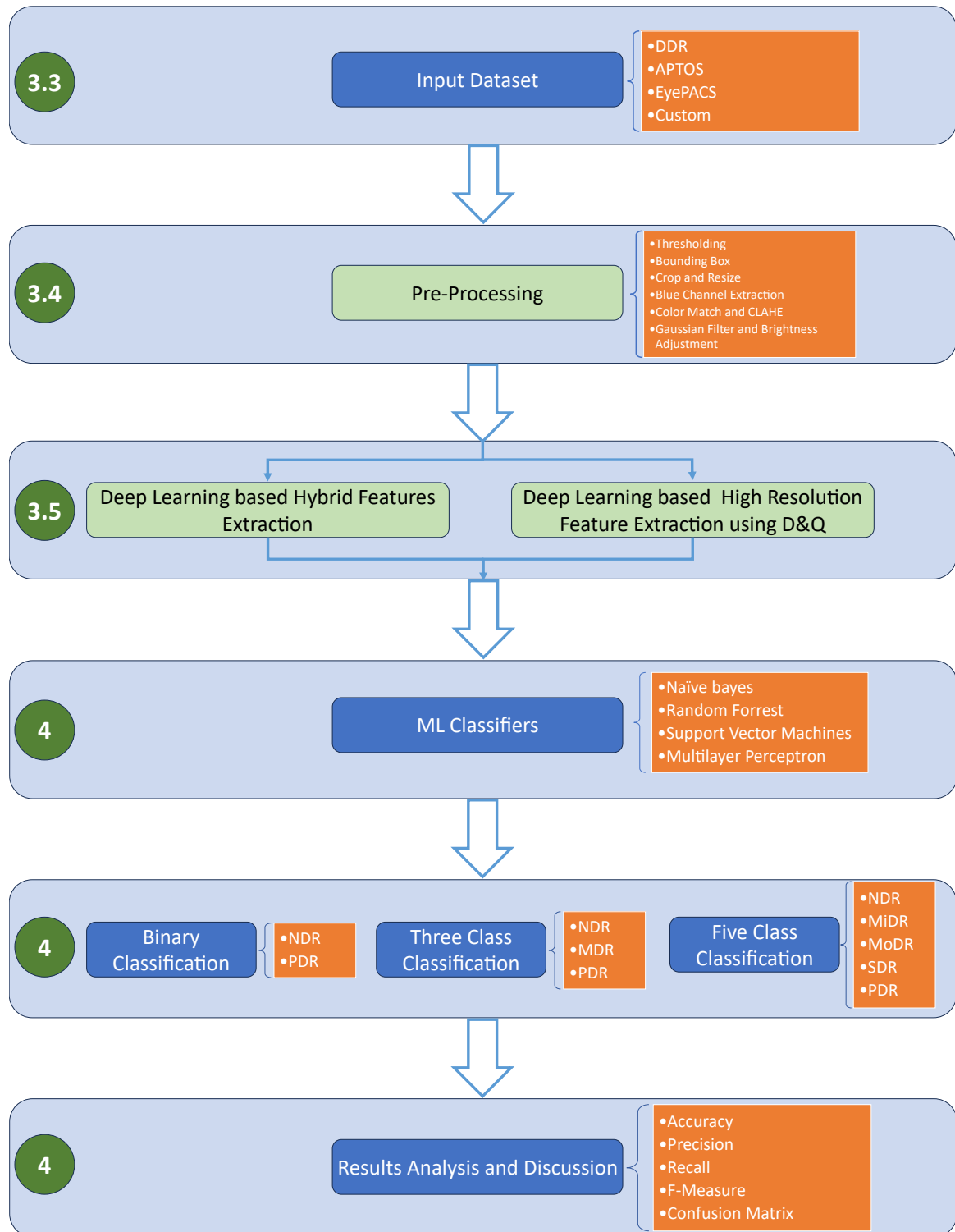


Figure 3.1: Workflow Diagram of DR Classification from Fundus Images

3.2 Experimental Environment

The hardware used in this work consists of an AMD Ryzen 2700x processor with 32 Gigabytes of memory. The Graphics Processing Unit (GPU) installed in the system is an NVIDIA GeForce RTX 2080 with 8 GB graphics memory. MATLAB was used for running various image processing and ML algorithms. Different classifiers are used in MATLAB for binary and multiclass classification of input images.

3.3 Experimental Datasets

There are various datasets that are available for fundus image classification on the internet.

3.3.1 DDR Dataset

The DDR dataset (Li et al., 2019b) consists of 13,673 fundus images. These images are collected from 9598 patients from 147 hospitals in China covering 23 provinces. The images are divided into 6 classes. Five of the classes follow the ICDR severity scale i.e., No Dr, Mild DR, Moderate DR, Severe DR and Proliferative DR. The sixth additional class contains images which are of poor quality and are marked ungradable. Some samples of the DDR dataset are shown in the Figure 3.2.

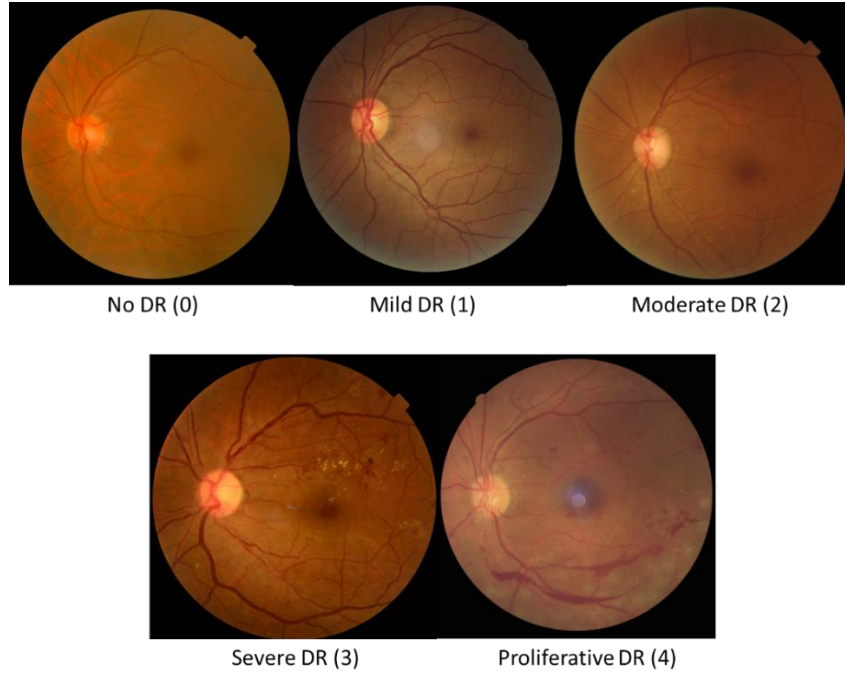


Figure 3.2: Different Stages of DR in DDR dataset (Li et al., 2019b)

The data suffers from imbalance as indicated in the histogram plot of the dataset where No DR (50.00%) and Moderate DR (35.00%) classes cover 85.00% of the gradable images. The rest of the classes i.e., Mild DR, Severe DR, Proliferative DR cover 5.03%, 1.88%, and 7.29% of the gradable images.

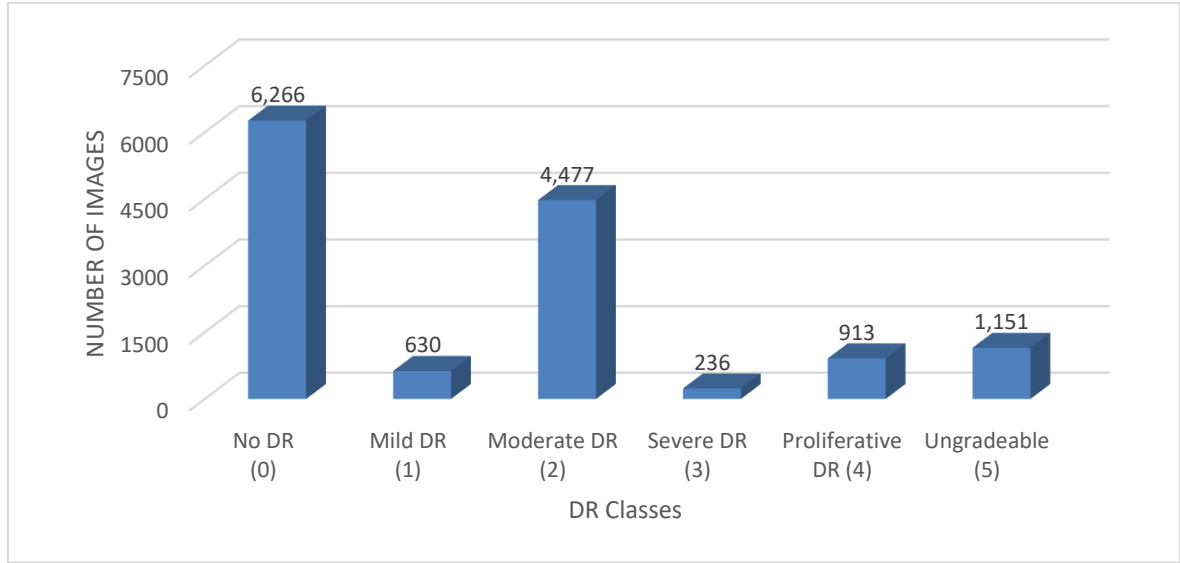


Figure 3.3: Data Distribution in each Class of DDR Dataset

3.3.2 APTOS Dataset

The APTOS data is available on the Kaggle website (Karthik Maggie, 2019). There are 3662 fundus images present in the dataset that were collected from The Aravind Eye Hospital in India. The labels for the images use the ICDR severity scale for five stages of DR classification i.e., 0 (No DR), 1 (Mild DR), 2 (Moderate DR), 3 (Severe DR), and 4 (Proliferative DR). The distribution of the data for the various stages of DR is shown in Figure 3.4.

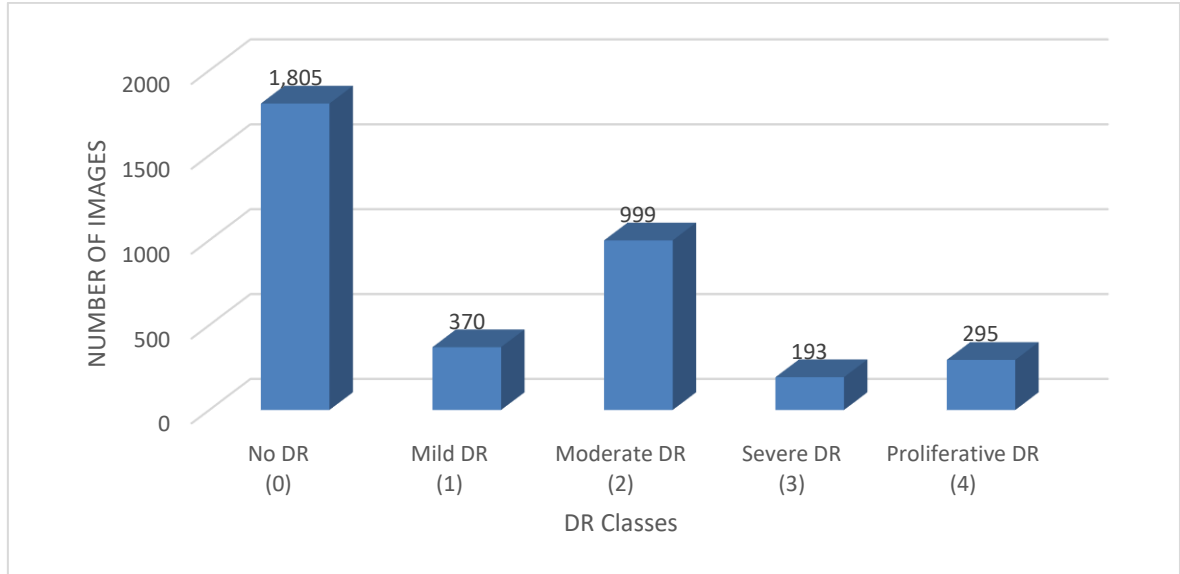


Figure 3.4: Fundus Image Distribution of Different Classes in the APTOS Dataset

A sample of images suffering from various stages of DR according to the ICDR severity scale in the APTOS dataset is shown in Figure 3.5. The healthy images with no DR contain no Micro Aneurysms (MAs) or Haemorrhages (HEs). Images labelled in stage 1 contain a few lesions. In stage 2, images contain some MAs, Exudates (EXs), and at least one Haemorrhage (HE). Stage 3 contains MAs from 5 to 15 and HEs less than 5. In the last group with images labelled as stage 4, MAs above 15 and HEs above 5 are present.

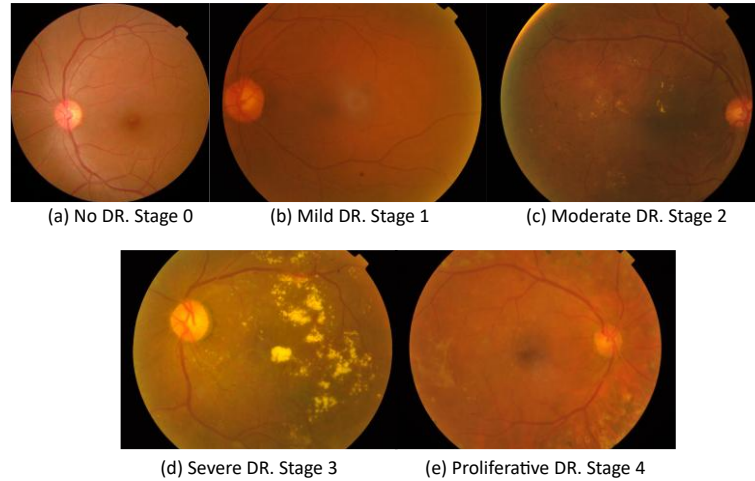


Figure 3.5: Various Stages of DR According to the ICDR Severity Scale in the APTOS Dataset (Karthik Maggie, 2019)

3.3.3 EyePACS Dataset

The EyePACS Dataset which is available on the Kaggle website (Emma Dugas Jared, 2015) contain 88,702 images of 44,351 patients. Each patient has the left and right eye images in the dataset. The label for the images uses the ICDR severity scale for five stages of DR classification i.e., 0 (No DR), 1 (Mild DR), 2 (Moderate DR), 3 (Severe DR), 4 (Proliferative DR). The distribution of the data for the various stages of DR is shown in Figure 3.6.

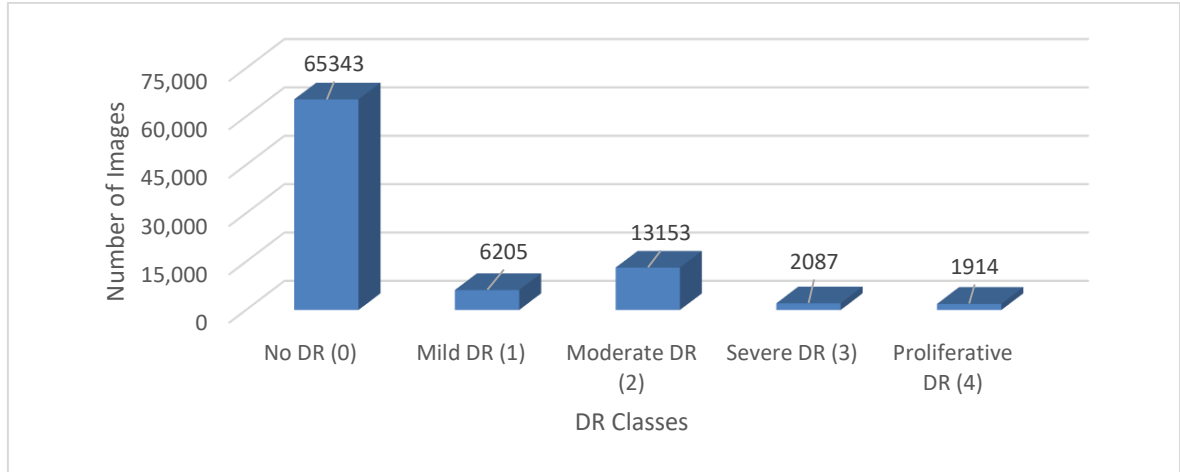


Figure 3.6: Dataset Distribution (image count) in EyePCS Dataset

This dataset although contain vast number of samples for DR but suffers from major imbalance and lacks quality in the fundus images. The images in the dataset contain various artefacts including noise, focus issues, haze, colour imbalance, and exposure. This issue is illustrated in Figure 3.7. Besides the issues some images in the dataset are completely black and are unusable.

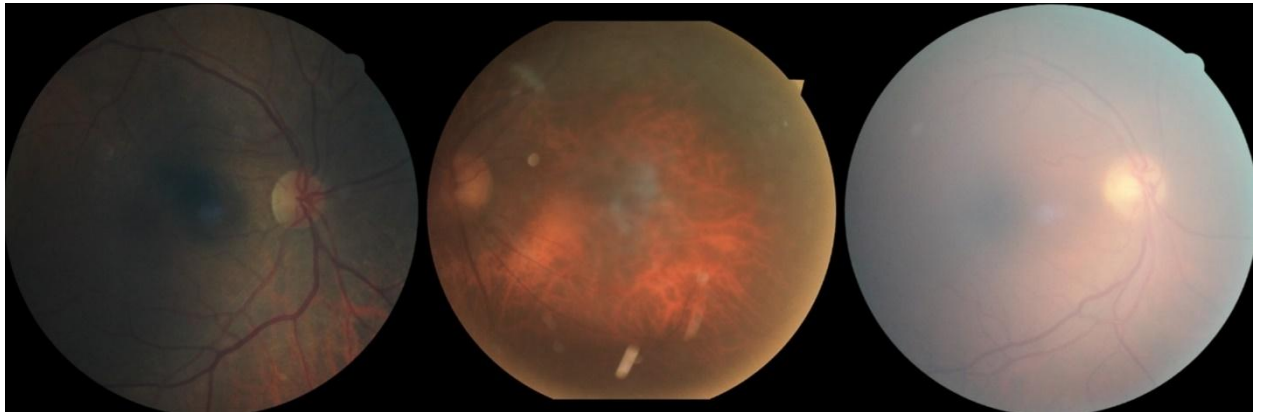


Figure 3.7: Sample Images Suffering from Focus and Exposure Issues (Emma Dugas Jared, 2015)

In addition to these datasets, various other datasets with a smaller number of images are used in the literature. A summary of various all these datasets is presented in Table 3.1.

Table 3.1: Summary of Different DR Image Processing Datasets

Dataset	Year	Number of Images	Labels
DDR	2019	13673	0-5 (5 Stages) (1 ungradable)
APTOS	2019	3662	0-4 (5 stages)
Kaggle EyePACS	2016	88,702	0-4 (5 stages)
Messidor-2	2014	1,748	0-4 (5 Stages)
IDRiD	2018	516	0-4 (5 stages)
DIARETDB0	2006	130	0-1 (2 stage)
DIARETDB1	2009	89	0-1 (2 stage)
E-Ophtha	2014	463	0-1 (2 stage)

3.3.4 Proposed Custom Dataset

As seen from Table 3.1, there are various image datasets available for DR image classification. However, most of these datasets suffer from class imbalance and quality issues. Most of these datasets contain ample fundus images in the No DR stage of DR detection. The latter stages of DR can be combined when doing binary classification for fundus images. However, for multiclass DR image classification, data augmentation can be utilised to create a balanced dataset for training the ML algorithms, but it has its limitation on the variety of data produced. In this work, we create a combined dataset for multiclass DR classification using images from DDR, APTOS, and EyePACS datasets. Since PDR class (Stage 4) has the least number of images, combining the images from all three datasets provide 2516 images for this class. We set this limit as the base for other classes and create a custom dataset containing 12,580 fundus images in total with each class containing 2,516 images. This dataset is used in later stages of the results section to check the performance of our proposed frameworks for multiclass DR classification. The histogram plot of the custom dataset is presented in Figure 3.8.

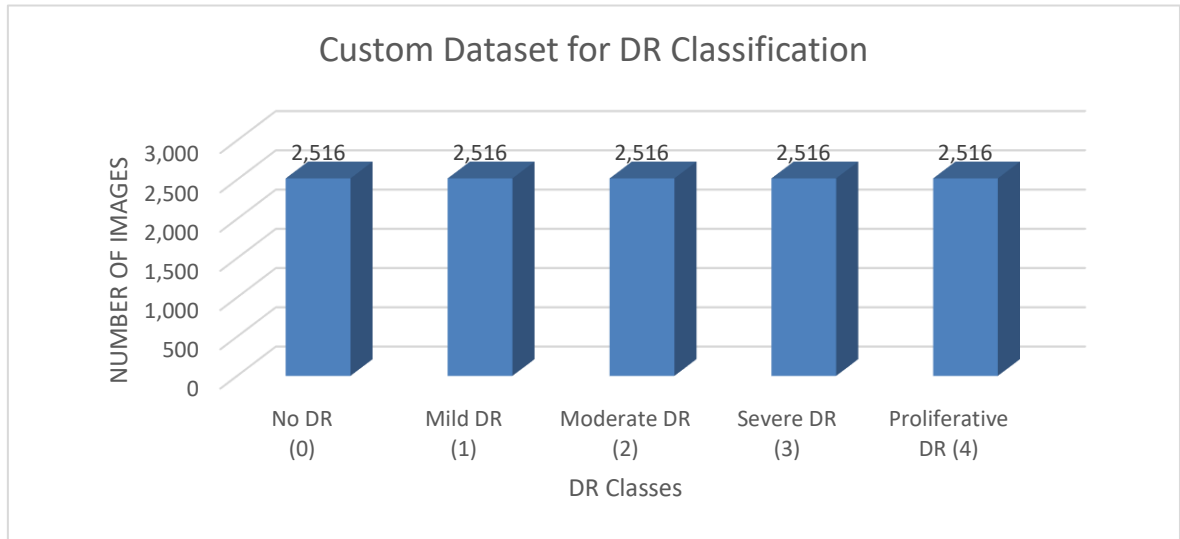


Figure 3.8: Custom Dataset for DR Image Classification

3.4 Proposed Framework for Pre-Processing fundus images

The fundus images in different datasets suffer from varying degradations that includes noise, artefacts, illumination, and colour variation. In our work, we use the blue channel from the fundus images to help automatically identify and enhance fundus images suffering from degradation. The blue channel helps in identifying the images suffering from degradation since those images have high blue channel variance compared to standard fundus images (C & R, 2023). The detailed workflow of the pre-processing framework is presented in Figure 3.9. The steps involved in the enhancement procedure are explained below:

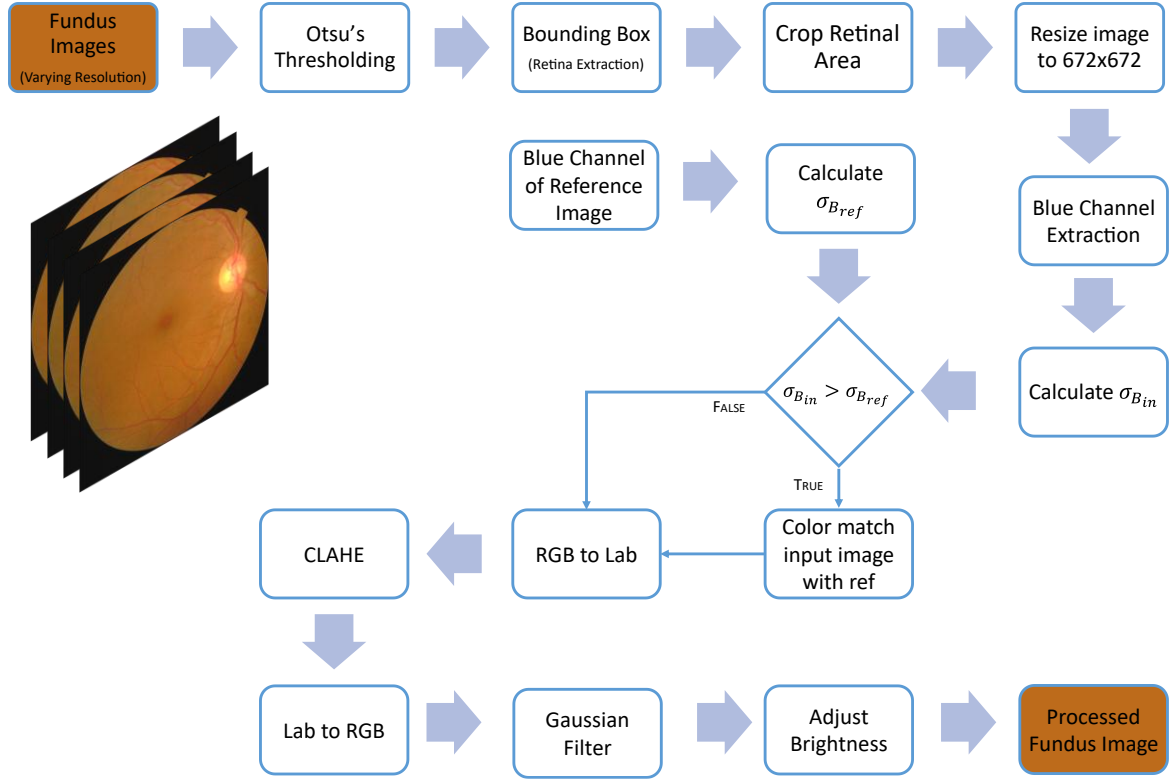


Figure 3.9: Proposed Pre-Processing Framework for Enhancing Fundus Images

- One of the main works in pre-processing is to extract the retinal area from input images of varying sizes. To accomplish this, the images are converted to grayscale level and passed to Otsu's thresholding function. The resulting threshold value is used to convert the grayscale image is into binary image. A sample of this conversion on an input fundus image is shown in Figure 3.10

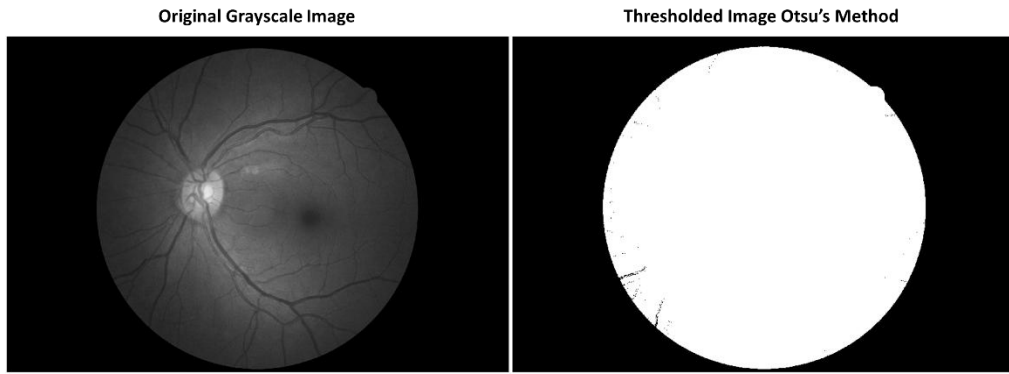


Figure 3.10: Effect of applying Otsu's Thresholding on Fundus Image

- In the next stage, we find the largest connected area in the binary image. This **bounding box** is used to extract the retinal area from the original RGB image. A sample of extracting the retinal structure after this operation is shown in Figure 3.11.



Figure 3.11: Cropped Retinal Structure from Input Fundus Image

- This cropped retinal structure from different images of varying dimensions is resized to $672 \times 672 \times 3$. This helps in providing a high-resolution base image that can be further enhanced using the remaining stages of pre-processing and used as input for divide and conquer feature extraction framework that is presented in this work.

- The standard deviation of the blue channel of fundus images that suffer from degradation is usually higher than the standard fundus images captured by the ophthalmologists. Figure 3.12 shows the standard deviation of a reference and degraded fundus image.



Figure 3.12: Standard Deviation and Mean of Degraded and Reference Fundus Image

- The standard deviation and mean of the input image are compared in the next phase with the reference image. If the value is greater than the reference image, a colour matching is performed that scales the R component of the input image based on the ratio of standard deviation of input and reference images. The colour matched degraded image based on reference image is shown in Figure 3.13.

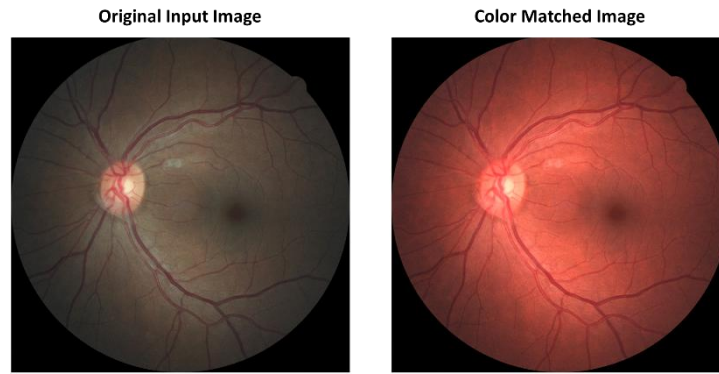


Figure 3.13: Colour Matching Input Fundus Image with Reference Image

- Next, CLAHE is applied on the image that enhances the contrast of lesions inside the image using the transformation of RGB to L^*a^*b colour space. The comparison is shown in Figure 3.14.

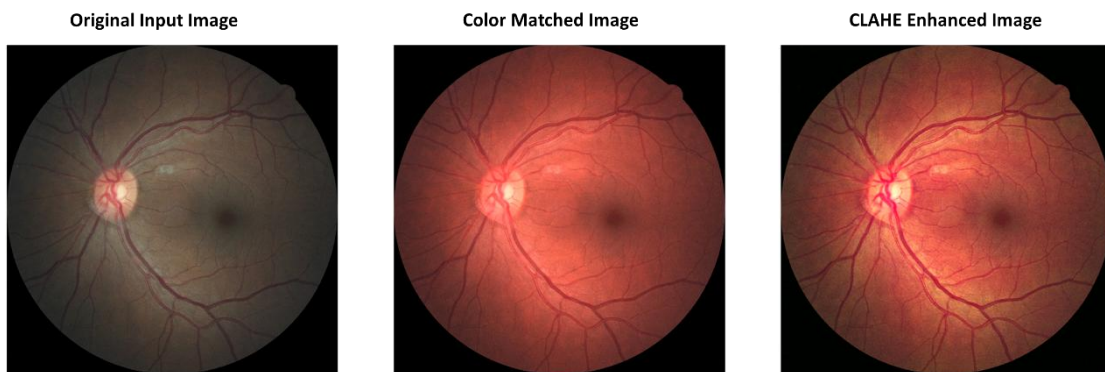


Figure 3.14: CLAHE on Fundus Image using RGB to L^*a^*b Colour Transformation

- In the last stage, the image passes through a gaussian filter and brightness of the image is automatically adjusted to generate the final result of pre-processed image.

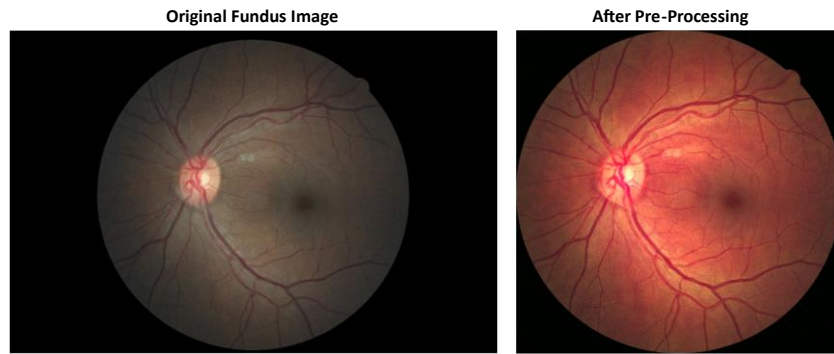


Figure 3.15: Final Result of Pre-Processed Fundus Image

Some sample results of pre-processing on multiple degraded images from the DDR dataset are shown in Figure 3.16.

Table 3.2: Results of Proposed framework for Pre-Processing Fundus Images

Original Image	Color Matched	Processed Image

3.5 Proposed Framework for DR Image Classification

In this work, there are two proposed frameworks for DR image classification. In first proposed algorithm, hybrid features from best performing CNN on pre-processed fundus images are extracted with help of transfer learning and classification is performed with the help of SVM, RF and other ML classifiers. In the second proposed algorithm, a divide and conquer approach is presented that helps in extracting the high-resolution image features and at the same time reduce the GPU memory usage. Classification in the second algorithm is also implemented through different ML classifiers. The details of both these algorithms are given next.

3.5.1 Hybrid CNN Features based DR Image Classification

In this section, a hybrid approach using transfer learning is presented. In pre-processing, the algorithm proposed in the previous section is utilised and images are resized afterwards to match the input of two best performing CNN models used for feature extraction. Afterwards, the initial layers of both CNN architectures are frozen and pre-processed input images which are resized to match input CNN dimensions are passed through to get the activation values. At the end of each architecture, the SoftMax layer is removed, and 1000 features are extracted from the fully connected layer. These features are concatenated for each processed image to create feature set for all processed images F . The feature set is afterwards divided into training F_T and validation set F_V which are passed to ML classifiers for training and testing. The pseudo-code of the proposed hybrid feature extraction method is presented in Figure 3.16. As an example, the process of extracting hybrid features based on GoogleNet (Szegedy et al., 2015) and ResNet-18 (He et al., 2016) architectures are presented in Figure 3.17. The pre-processed images are resized to match the input of the GoogleNet and ResNet-18 architectures i.e., $224 \times 224 \times 3$. Each of these

models applies the convolution, normalization, and pooling layers on the input fundus images. GoogleNet uses the inception modules to reduce the computational resources and capture the spatial and local features, while the ResNet-18 model uses skip connections to avoid degradation and reduce the training error. Next, the feature vectors from both the models are merged to form a hybrid feature vector that contains 2000 features. This feature vector is passed to different classifiers and results are compared with different methods for DR classification. For binary classification, the fundus images are grouped into two categories, i.e., No DR (NDR) which represents stage 0 of the ICDR severity scale, and PDR images that combines the images from stages 1-4. Multiclass classification is done in two ways. First, the images are grouped into three classes i.e., no DR (NDR) representing the images from stage 0 of the ICDR severity scale, MDR representing the images from stages 1 and 2 (mild and moderate), and PDR which characterizes the stages 3 and 4 (severe and proliferative) of the ICDR severity scale. Lastly, using the balanced custom dataset, five stage multiclass DR classification is performed with NDR representing the stage 0 (No DR) of ICDR severity scale, MiDR representing stage 1 (Mild DR), MoDR representing stage 2 (Moderate DR), SDR representing stage 3 (Severe DR) and PDR representing stage 4 (Proliferative DR).

Algorithm: Hybrid CNN Features based DR Image Classification

Input: Fundus images M with labels L

Algorithm:

```

    read  $M$ , read  $L$ 
     $P \leftarrow \text{Pre-process}(M)$ 
     $R \leftarrow \text{Resize the processed images } P \text{ based on the input size of CNN models.}$ 
     $F \leftarrow \text{Hybrid feature vector from } R \text{ images}$ 
     $F_T, F_V \leftarrow \text{Divide } F \text{ into training } F_T \text{ and validation } F_V$ 
     $n \leftarrow \text{number of features to extract from each CNN model}$ 
     $f \leftarrow \text{TransferLearning}(M, I_{p,x,y}) \text{ (Extract } n \text{ Features)}$ 
     $\text{Nets} \leftarrow \text{CNN models with frozen initial layers and classification layers removed}$ 
    for each image  $r$  in  $R$ 
        for  $n$  in  $\text{Nets}$ 
             $f \leftarrow \text{TransferLearning}(n, r) \text{ (Extract } n \text{ Features)}$ 
             $F_r \leftarrow F_r \cup f$ 
        end for
    end for
     $\text{ML\_Model} = \text{Train}(\text{ML classifier}, F_T, L)$ 
     $\text{DR\_class} = \text{Classify}(\text{ML\_Model}, F_V)$ 

```

Figure 3.16: Proposed Algorithm for Hybrid Feature Extraction and Classification of Fundus Images

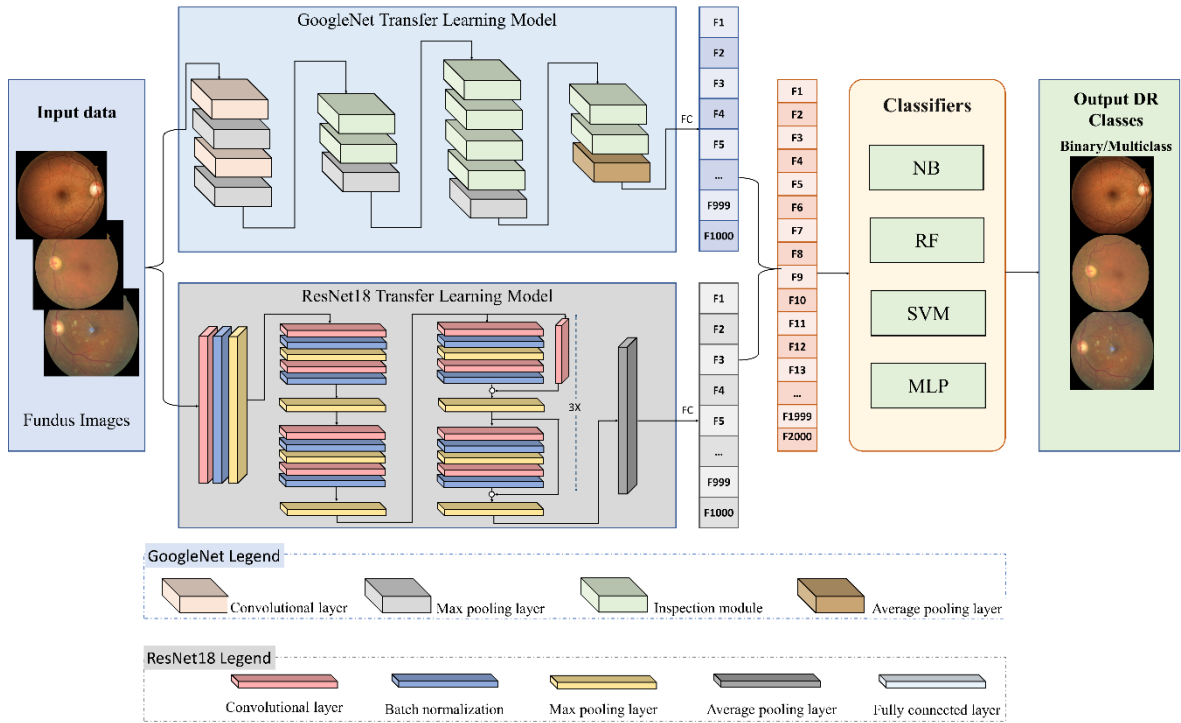


Figure 3.17: The Proposed Method for Hybrid Feature Extraction and Classification of Fundus Images

3.5.2 High-Resolution Memory Efficient Divide and Conquer approach DR image classification

In this section, a new framework to deal with high-resolution fundus images and minimise the GPU memory usage is presented. Increasing the input image resolution that is fed to a CNN is one of the methods to improve the quality of features that are extracted from the input data. The inter-class similarities between different classes of DR can be better differentiated by the CNN model if the features are extracted from a high-resolution input image. However, increasing the input size of the image to a CNN causes slow training time and high GPU memory utilisation. Most of the graphics processing units are unable to handle inputs with high pixel density. This work proposes a divide and conquer approach for handling high-resolution images while keeping the GPU memory utilisation within bounds. The main idea about the proposed algorithm is presented in the Figure 3.18.

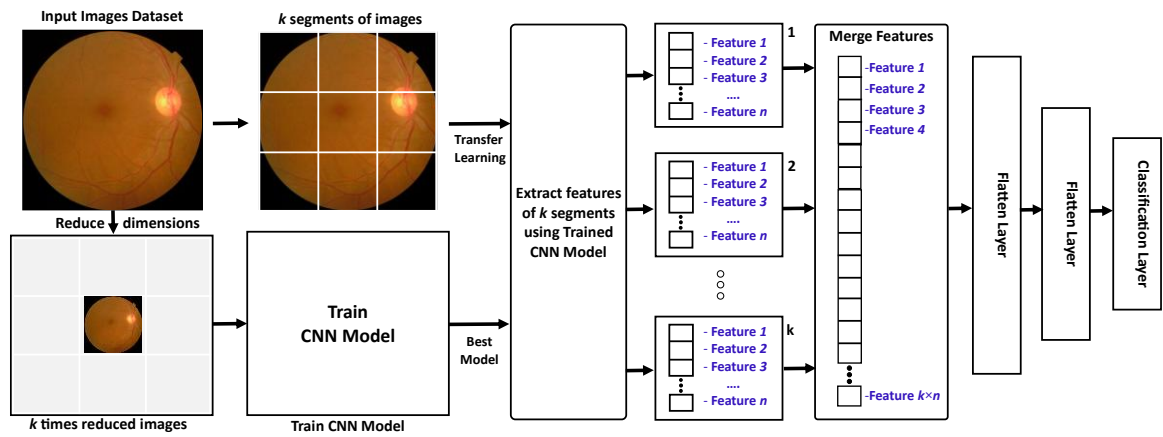


Figure 3.18: Proposed Divide and Conquer Approach for DR Image Classification

Algorithm: Divide and Conquer approach for extracting high resolution features from input images

Input: Fundus images M with labels L and scaling factor k (number of image patches)

Algorithm:

```

read  $M$ , read  $L$ 
 $P \leftarrow \text{Pre-process } (M)$ 
 $R \leftarrow \text{Resize the processed images } P \text{ based on CNN model.}$ 
 $T, V \leftarrow \text{Divide } R \text{ into training } T \text{ and validation } V$ 
 $f_c \leftarrow \text{number of neurons in fully connected layer of original CNN}$ 
 $n \leftarrow \text{number of features to extract for each patch i.e., } \left\lfloor \frac{f_c}{k} \right\rfloor$ 
 $\text{Net} \leftarrow \text{modify CNN model with added fn layer to extract}$ 
 $M \leftarrow \text{Train modified model (Net) with } (T, V)$ 
 $k_w, k_h \leftarrow \sqrt{k}$  (Dimension of patches based on scaling factor  $k$ )
 $P_w, P_h \leftarrow \text{Height } h \text{ and width } w \text{ of pre-processed images } P$ 
for each image  $p$  in  $P$ 
    for  $x$  in  $\text{range}(k_w)$ 
        for  $y$  in  $\text{range}(k_h)$ 
             $s \leftarrow \frac{P_w}{k_w} * \frac{P_h}{k_h}$  %size of each patch
             $I_{p,x,y} \leftarrow \text{Extract patch } s \text{ based on } (x,y) \text{ from image } p$ 
        end for
    end for
for each  $p$  in  $I$ 
    for  $x$  in  $\text{range}(k_w)$ 
        for  $y$  in  $\text{range}(k_h)$ 
             $f \leftarrow \text{TransferLearning}(M, I_{p,x,y})$  (Extract  $n$  Features)
             $F_p \leftarrow F_p \hat{\cup} f$ 
        end for
    end for
end for
 $\text{ML\_Model} = \text{Train}(\text{classifier}, F, L)$ 
 $\text{DR\_class} = \text{classify}(\text{ML\_Model}, F_p)$ 

```

Figure 3.19: Divide and Conquer Approach for Extracting High Resolution Features from Input Images

The steps involved in implementing this approach are presented in Figure 3.19. The algorithm first reads and pre-processes the input images using the proposed methodology. In the next stage, the pre-processed images are resized by a scaling factor k that is defined in Equation 3.1.

$$k = k_w \times k_h \quad \text{Equation 3.1}$$

Where w and h are scaling ratios

For example, if scaling factor is nine, k_w is set to three and k_h is set to three. This means that pre-processed images will be reduced in width and height by a factor of three. The resized images are used to train a modified CNN model that extracts n features. The number of features to extract are calculated by dividing the number of neurons in the FC connected layer of the CNN divided by the scaling factor k and taking the floor of resulting value.

$$n = \left\lfloor \frac{f_c}{k} \right\rfloor \quad \text{Equation 3.2}$$

As an example, if the scaling factor is nine and there are 1000 neurons in the FC layer of the CNN, then the value of n will be $n = \text{floor}(1000/9)$ i.e., 111. Based on the number of features n , the CNN model is modified by adding fully connected layer with n neurons before the classification output of the CNN. This fully connected layer helps in extracting n features from the image patches. Next, the high-resolution pre-processed images are divided into k patches according to the Equation 3.3.

$$I \leftarrow \bigcup_{p=1}^P I_p \quad \text{Equation 3.3}$$

$$I_p \leftarrow \bigcup_{x,y=1}^{\sqrt{k}} I_{p,x,y} \quad \text{Equation 3.4}$$

Where $I_{p,x,y}$ is the patch extracted from the high-resolution pre-processed image. The deep learning model trained in the previous step is then used to extract n features from each of the image patch using TL. Afterwards, the resulting features are combined to create the feature vector F_p for each pre-processed image having $n \times k$ features. Equation 3.5

describes the feature extracted for pre-processed images where f represents the features extracted from each patch.

$$F_p \leftarrow F_p \cup f \quad \text{Equation 3.5}$$

These collective features that contain high-resolution features from different patches of the image are used for classification by different ML classifiers. A summarized workflow of the steps in divide and conquer approach is also represented in Figure 3.20.

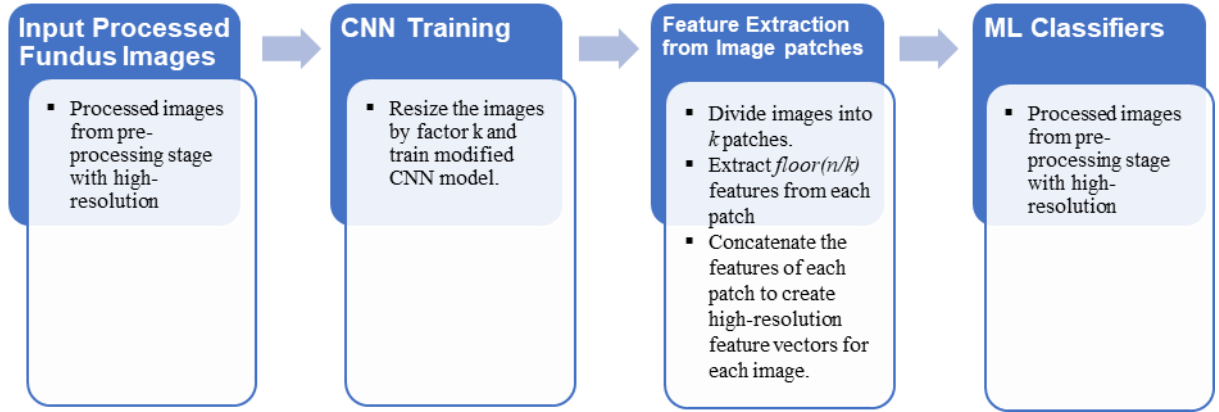


Figure 3.20: Flowchart of proposed Memory Efficient Divide and Conquer Framework

In this work, various ML classifiers such as Naïve Bayes, Random Forrest, Support Vector Machines and Multi-Layer Perceptron are used to check the performance of this divide and conquer framework. Our proposed methodology outperforms most of the recent approaches in the literature for DR image classification.

3.6 Summary

In this chapter, detailed information regarding the proposed methodologies for fundus image pre-processing, feature extraction and classification are presented. The details of the datasets used for training and testing the proposed methods are also listed. In the next chapter, the results of the proposed frameworks will be discussed.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Overview

In this chapter, the results acquired by applying the various proposed methods for DR image classification are presented. In the beginning, the performance comparison of using the proposed image enhancement technique is presented. Next, the results of running binary classification using standard CNN models is discussed. Afterwards, the results of various features extraction methods and corresponding ML algorithms is analysed. The proposed methods provide ample performance boost for different metrics. The CNN models are tested by running the training process with batch size and epochs of 32. The low value of batch size helps in keeping the GPU memory within the limits. The learning rate is set to 0.0001. The split between training and testing data is set to 80-20% during the tests performed. For the proposed Divide and conquer approach, the number of features n extracted from each patch is calculated from Equation 3.2 with number of patches k for the fundus image is set to 9 and f_c set to 1000. A summary of the parameters used in all the experiments is given in Table 4.1. For ResNet CNN models the 50 Layer ResNet-50 CNN is used in all the experiments.

Table 4.1: Parameters used for Experimentation

Parameter	Value
Batch size	32
Epochs	32
Learning Rate	0.0001
n (Features for each patch)	111
k (Number of patches)	9
$k_w = k_h$	3

4.2 DR Classification using proposed Pre-Processing Framework

We first discuss the results of DR classification using the proposed pre-processing method for fundus images. In binary classification, the data no longer suffers from the imbalance issue after combining stage 1-4 images into a single class, i.e., PDR.

Table 4.2 shows the classification results of different CNN models (ResNet, AlexNet, VGG16, GoogleNet, EfficientNet) without using the proposed pre-processing framework. Table 4.2 shows the results when the proposed pre-processing methodology is applied to the input fundus images from the DDR dataset.

Table 4.2: Classification Results of Different CNN Models without Proposed Pre-Processing

CNN Model	Metrics	NDR	PDR	Average
AlexNet	Accuracy	91.70%	72.10%	81.91%
	Precision	76.70%	89.66%	83.18%
	Recall	91.70%	72.10%	81.91%
	F-Measure	83.53%	79.93%	81.73%
GoogleNet	Accuracy	93.93%	72.90%	83.43%
	Precision	77.64%	92.31%	84.97%
	Recall	93.93%	72.90%	83.43%
	F-Measure	85.01%	81.46%	83.24%
ResNet	Accuracy	91.38%	75.86%	83.63%
	Precision	79.13%	89.78%	84.45%
	Recall	91.38%	75.86%	83.63%
	F-Measure	75.86%	91.38%	83.53%

Table 4.2: continued

EfficientNet	Accuracy	90.50%	77.30%	83.91%
	Precision	79.97%	89.04%	84.50%
	Recall	90.50%	77.30%	83.91%
	F-Measure	84.91%	82.76%	83.83%
VGG16	Accuracy	92.74%	76.82%	84.78%
	Precision	80.03%	91.35%	85.68%
	Recall	92.74%	76.82%	84.78%
	F-Measure	85.91%	83.46%	84.69%

The results in Table 4.3 show that our proposed pre-processing methodology provides a good performance improvement for DR classification in all the CNN models tested. The average comparison for CNN models with and without proposed pre-processing methodology is given in Table 4.4.

Table 4.3: Classification Results of Different CNN Models with Proposed Pre-processing

CNN Model	Metrics	NDR	PDR	Average
AlexNet	Accuracy	90.66%	74.74%	82.71%
	Precision	78.24%	88.88%	83.55%
	Recall	90.66%	74.74%	82.71%
	F-Measure	83.99%	81.20%	82.60%
GoogleNet	Accuracy	89.78%	79.54%	84.66%
	Precision	81.46%	88.60%	85.03%
	Recall	89.78%	79.54%	84.66%
	F-Measure	85.42%	83.82%	84.66%
ResNet	Accuracy	88.91%	85.29%	87.10%
	Precision	85.82%	88.47%	87.15%
	Recall	88.91%	85.29%	87.10%
	F-Measure	87.34%	86.85%	87.10%
EfficientNet	Accuracy	88.59%	85.13%	86.86%
	Precision	85.65%	88.16%	86.90%
	Recall	88.59%	85.13%	86.86%
	F-Measure	87.09%	86.62%	86.86%

Table 4.3: continued

VGG16	Accuracy	87.95%	86.49%	87.22%
	Precision	86.70%	87.75%	87.23%
	Recall	87.95%	86.49%	87.22%
	F-Measure	87.32%	87.12%	87.22%

Table 4.4: Average Comparison between Different CNN Models With and Without Proposed Pre-processing

CNN Model	Metrics	Average without pre-processing	Average with pre-processing
AlexNet	Accuracy	81.91%	82.71%
	Precision	83.18%	83.55%
	Recall	81.91%	82.71%
	F-Measure	81.73%	82.60%
GoogleNet	Accuracy	83.43%	84.66%
	Precision	84.97%	85.03%
	Recall	83.43%	84.66%
	F-Measure	83.24%	84.66%

Table 4.4: continued

ResNet	Accuracy	83.63%	87.10%
	Precision	84.45%	87.15%
	Recall	83.63%	87.10%
	F-Measure	83.53%	87.10%
EfficientNet	Accuracy	83.91%	86.86%
	Precision	84.50%	86.90%
	Recall	83.91%	86.86%
	F-Measure	83.83%	86.86%
VGG16	Accuracy	84.78%	87.22%
	Precision	85.68%	87.23%
	Recall	84.78%	87.22%
	F-Measure	84.69%	87.22%

Our proposed pre-processing framework provides a good performance uplift for the fundus images. The results of pre-processing framework using various CNN models are presented in a graph format in Figure 4.1. It is worth noting that the ResNet and VGG16 models perform best among the different CNN models that are used on pre-processed fundus images for classification with a maximum accuracy of 87.10% and 87.22% respectively. An ablation study for color matching step in the pre-processing framework is also presented in Figure 4.2. The findings indicate that color matching plays a modest yet significant role in

enhancing performance through the pre-processing framework. For the rest of the results, the pre-processed fundus images are used as input to the proposed methodologies.

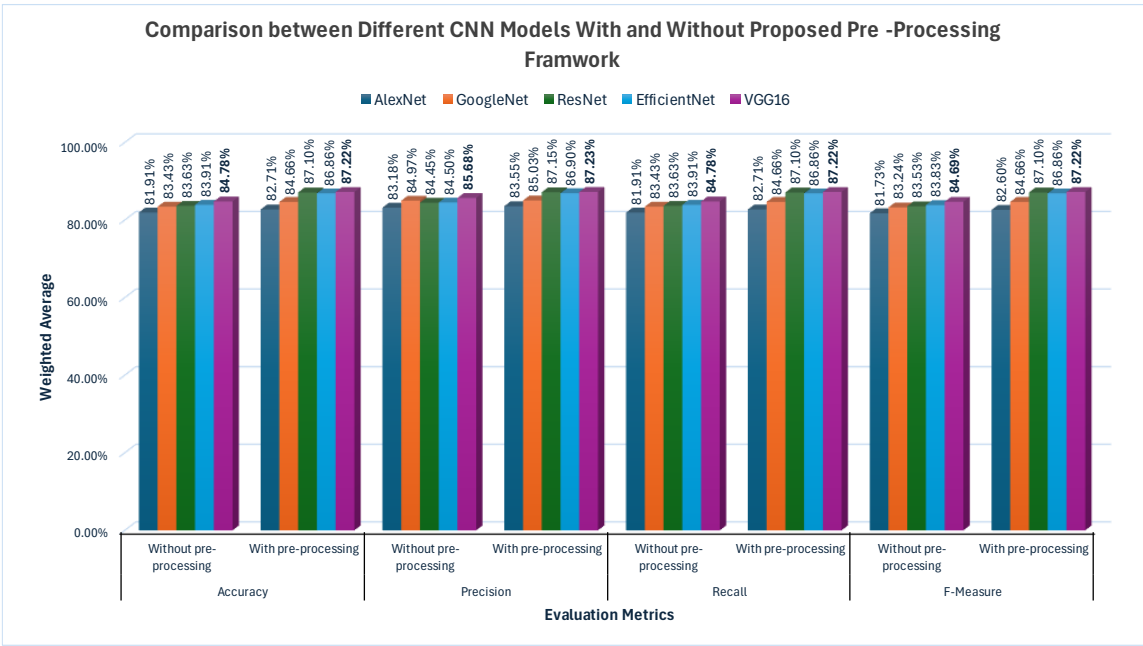


Figure 4.1: Comparison between different CNN Models With and Without Proposed Pre-Processing Framework.

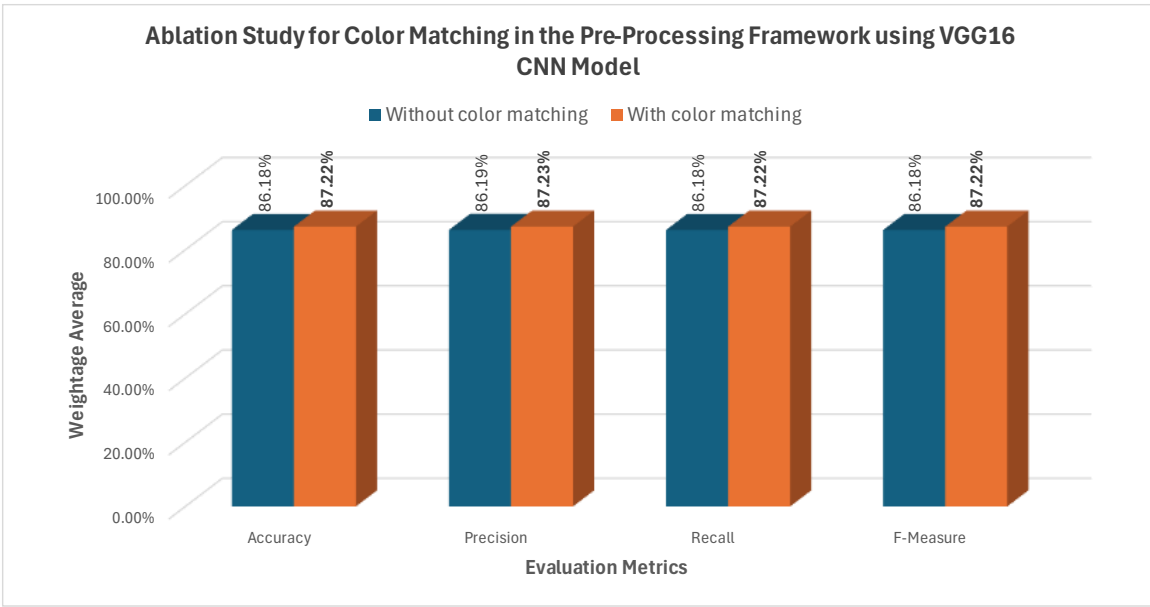


Figure 4.2: Ablation Study for Color Matching in the Pre-Processing Framework using VGG16 CNN Model.

4.3 Binary DR Classification using Pre-Processed Fundus Images

We first discuss the results of binary DR classification using the fundus images. In binary classification, stage 0 images in the DDR dataset represent the no DR image class i.e., NDR. Stage 1-4 images in the DDR dataset are combined into a single class, i.e., PDR. The data no longer suffers from imbalance issues since NDR and PDR class now contain approximately equal number of images.

4.3.1 Binary DR classification using CNN Features extraction and Transfer Learning

In this section the results acquired using the transfer learning approach of features extraction are compared between different ML algorithms. The input to these experiments is the pre-processed images acquired from the previous section. These images are resized to appropriate size depending on the CNN model used to extract the features. Table 4.5 shows the experimental results of applying binary classification on the feature vector extracted from the GoogleNet model. This feature vector is passed on to four classifiers, i.e., NB, RF, SVM, and MLP. Results show that the SVM classifier provides the highest-class accuracy of 84.73%. SVM also achieves the highest average values for precision, recall, and f-measure.

Table 4.5: Experimental Results of Different ML Classifiers for Binary Classification using Features Extracted from the GoogleNet Model

Classifier	Metrics	NDR	PDR	Average
Naïve Bayes	Accuracy	87.77%	76.26%	82.19%
	Precision	78.71%	86.18%	82.33%
	Recall	87.77%	76.26%	82.19%
	F-Measure	82.99%	80.92%	81.99%
Random Forrest	Accuracy	86.90%	82.01%	84.53%
	Precision	83.72%	85.47%	84.57%
	Recall	86.90%	82.01%	84.53%
	F-Measure	85.28%	83.71%	84.52%
Support Vector Machines	Accuracy	87.98%	81.27%	84.73%
	Precision	83.33%	86.40%	84.82%
	Recall	87.98%	81.27%	84.73%
	F-Measure	85.60%	83.76%	84.71%
Multilayer Perceptron	Accuracy	81.86%	85.56%	83.65%
	Precision	85.78%	81.59%	83.75%
	Recall	81.86%	85.56%	83.65%
	F-Measure	83.78%	83.53%	83.66%

The results of extracting the features using ResNet CNN model and passing on to different classifiers are represented in Table 4.6. The results provide better performance compared to GoogleNet CNN model. SVM classifier still performs better compared to other ML classifiers used for classification. The same trend is observed in Table 4.7 and Table 4.8 where EfficientNet and VGG16 CNN model are used to extract the features respectively.

Table 4.6: Experimental Results of Different ML Classifiers for Binary Classification using Features Extracted from the ResNet Model

Classifier	Metrics	NDR	PDR	Average
Naïve Bayes	Accuracy	83.18%	78.30%	80.82%
	Precision	80.31%	81.39%	80.84%
	Recall	83.18%	78.30%	80.82%
	F-Measure	81.72%	79.81%	80.80%
Random Forrest	Accuracy	87.29%	83.66%	85.53%
	Precision	85.05%	86.08%	85.55%
	Recall	87.29%	83.66%	85.53%
	F-Measure	86.15%	84.85%	85.52%
Support Vector Machines	Accuracy	88.84%	88.45%	88.65%
	Precision	89.11%	88.16%	88.65%
	Recall	88.84%	88.45%	88.65%
	F-Measure	88.98%	88.30%	88.65%

Table 4.6: continued

Multilayer Perceptron	Accuracy	88.29%	87.95%	88.13%
	Precision	88.64%	87.59%	88.13%
	Recall	88.29%	87.95%	88.13%
	F-Measure	88.47%	87.77%	88.13%

Table 4.7: Experimental Results of Different ML Classifiers for Binary Classification using Features Extracted from the EfficientNet Model

Classifier	Metrics	NDR	PDR	Average
Naïve Bayes	Accuracy	84.42%	80.12%	82.33%
	Precision	81.88%	82.85%	82.35%
	Recall	84.42%	80.12%	82.33%
	F-Measure	83.13%	81.46%	82.32%
Random Forrest	Accuracy	85.89%	83.83%	84.89%
	Precision	84.97%	84.81%	84.89%
	Recall	85.89%	83.83%	84.89%
	F-Measure	85.43%	84.32%	84.89%

Table 4.7: continued

Support Vector Machines	Accuracy	91.24%	87.54%	89.45%
	Precision	88.63%	90.37%	89.47%
	Recall	91.24%	87.54%	89.45%
	F-Measure	89.92%	88.94%	89.44%
Multilayer Perceptron	Accuracy	90.78%	86.96%	88.93%
	Precision	88.11%	89.86%	88.96%
	Recall	90.78%	86.96%	88.93%
	F-Measure	89.42%	88.39%	88.92%

Table 4.8: Experimental Results of Different ML Classifiers for Binary Classification using Features Extracted from the VGG16 Model

Classifier	Metrics	NDR	PDR	Average
Naïve Bayes	Accuracy	93.02%	83.50%	88.41%
	Precision	85.71%	91.83%	88.68%
	Recall	93.02%	83.50%	88.41%
	F-Measure	89.22%	87.47%	88.37%

Table 4.8: continued

Random Forrest	Accuracy	91.47%	89.77%	90.65%
	Precision	90.49%	90.82%	90.65%
	Recall	91.47%	89.77%	90.65%
	F-Measure	90.98%	90.29%	90.65%
Support Vector Machines	Accuracy	91.55%	90.35%	90.97%
	Precision	90.99%	90.95%	90.97%
	Recall	91.55%	90.35%	90.97%
	F-Measure	91.27%	90.65%	90.97%
Multilayer Perceptron	Accuracy	90.85%	89.77%	90.33%
	Precision	90.43%	90.22%	90.33%
	Recall	90.85%	89.77%	90.33%
	F-Measure	90.64%	89.99%	90.33%

The experimental results of applying binary classification on the feature vectors from different CNN models show that the SVM classifier provides the highest performance metrics values. The average accuracy for SVM using the features extracted from VGG16 model i.e., 90.97% is the highest compared to other classifiers and CNN models that are tested. SVM classifier using VGG16 model also achieves the highest average values for precision, recall, and f-measure at 90.97%, 90.97%, 90.97% respectively. The lowest accuracy is achieved by the NB classifier in all the models. The results also show that the

features extracted from VGG16 model provides better classification accuracy for DR class compared to other CNN models. EfficientNet and ResNet models provide the second-best accuracy of 89.45%, and 88.65% respectively. The results for VGG16 model are also presented in a graph format in Figure 4.3. The confusion matrix for VGG16 model for different classifiers is represented in Figure 4.4. The confusion matrix shows that although SVM classifiers gives best performance overall, but the NB classifier has better performance in identifying the patients with No DR. However, for practical purposes, the importance of correctly identifying the patients affected by DR is more important and SVM outperforms all the other classifiers in this case.

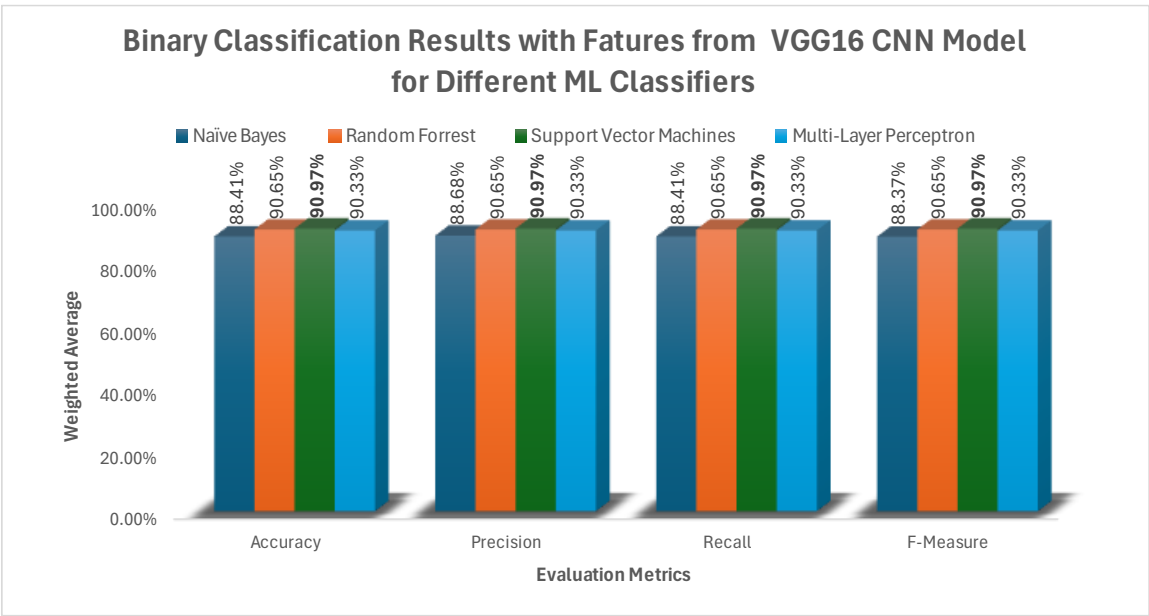


Figure 4.3: Binary Classification Results with Features from VGG16 CNN Model for Different ML Classifiers.

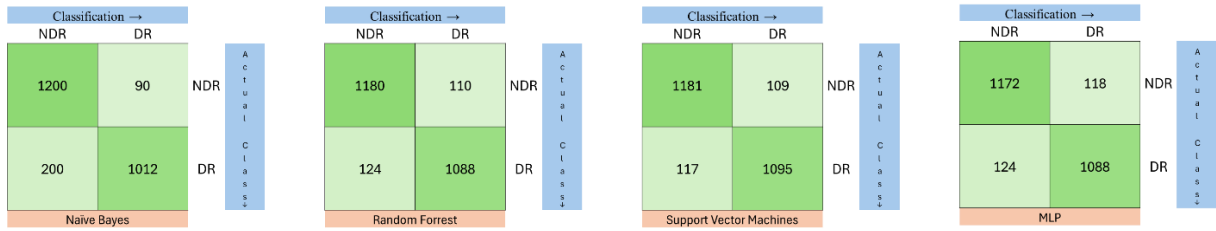


Figure 4.4: Confusion Matrix for Binary Classification using Features Extracted from VGG16 for Different Classifiers

4.3.2 Binary DR classification using Hybrid Features extraction and Transfer Learning

The result from previous section indicates that the VGG16 model can classify the images in the DR class better compared to other CNN models. The ResNet and EfficientNet models' performance is also close to each other but less compared to VGG16. In the hybrid approach, the feature vectors acquired from VGG16 are concatenated first with EfficientNet features and then with ResNet. This creates two hybrid features vector i.e., VGG16 + EfficientNet and VGG16 + ResNet. This results in providing improvement in performance metrics over the individual CNN models.

In Table 4.9, the results of using the first hybrid features vector i.e., VGG16 + EfficientNet, with 2000 features are presented. This model achieves the highest average accuracy of 92.69%. This provides approximately a 2% performance boost compared to using simple feature vectors from the CNN Models. SVM is again able to outperform other ML classifiers and achieve the highest average percentage in all the metrics, i.e., precision, recall, and f-measure. The classifier with the lowest performance is again the NB classifier with a weighted accuracy of 88.49%. However, the NB classifier gives second best performance accuracy in the NDR class for DR classification.

Table 4.9: Experimental Results of Different Classifiers for Binary Classification using Hybrid Features Extracted from VGG16 and EfficientNet

Classifier	Metrics	NDR	PDR	Average
Naïve Bayes	Accuracy	93.26%	83.42%	88.49%
	Precision	85.68%	92.08%	88.78%
	Recall	93.26%	83.42%	88.49%
	F-Measure	89.31%	87.53%	88.45%
Random Forrest	Accuracy	91.78%	90.18%	91.01%
	Precision	90.87%	91.16%	91.01%
	Recall	91.78%	90.18%	91.01%
	F-Measure	91.32%	90.67%	91.01%
Support Vector Machines	Accuracy	93.88%	91.42%	92.69%
	Precision	92.09%	93.34%	92.70%
	Recall	93.88%	91.42%	92.69%
	F-Measure	92.98%	92.37%	92.68%

Table 4.9: continued

Multilayer Perceptron	Accuracy	93.10%	90.43%	91.81%
	Precision	91.19%	92.49%	91.82%
	Recall	93.10%	90.43%	91.81%
	F-Measure	92.14%	91.45%	91.80%

Table 4.10: Experimental Results of Different Classifiers for Binary Classification using Hybrid Features Extracted from VGG16 and ResNet

Classifier	Metrics	NDR	PDR	Average
Naïve Bayes	Accuracy	92.87%	83.33%	88.25%
	Precision	85.57%	91.65%	88.52%
	Recall	92.87%	83.33%	88.25%
	F-Measure	89.07%	87.29%	88.21%
Random Forrest	Accuracy	92.02%	89.44%	90.77%
	Precision	90.27%	91.32%	90.78%
	Recall	92.02%	89.44%	90.77%
	F-Measure	91.13%	90.37%	90.76%

Table 4.10: continued

Support Vector Machines	Accuracy	93.10%	91.58%	92.36%
	Precision	92.17%	92.57%	92.37%
	Recall	93.10%	91.58%	92.36%
	F-Measure	92.63%	92.07%	92.36%
Multilayer Perceptron	Accuracy	90.85%	89.77%	90.33%
	Precision	91.19%	92.49%	90.33%
	Recall	93.10%	90.43%	90.33%
	F-Measure	92.14%	91.45%	90.33%

The results of using the second hybrid feature vector are presented in Table 4.10. Both the results of hybrid features vector classification indicate that combining features from different CNN classifiers can further improve the performance of the DR image classifier. The results of hybrid features from VGG16 and ResNet models are also presented in a graph format in Figure 4.5. The hybrid features from different CNN models combine to further enhance the binary classification compared to using single CNN models. The confusion matrix for binary classification using the hybrid features vector from VGG16 and EfficientNet are depicted in Figure 4.6. The confusion matrix for the classifiers indicates that SVM performs best for average as well as per class evaluation metrics since it classifies maximum number of correct predictions in each DR class. The SVM confusion matrix also contains the least number of false negative and false positive predictions compared to the other ML classifiers.

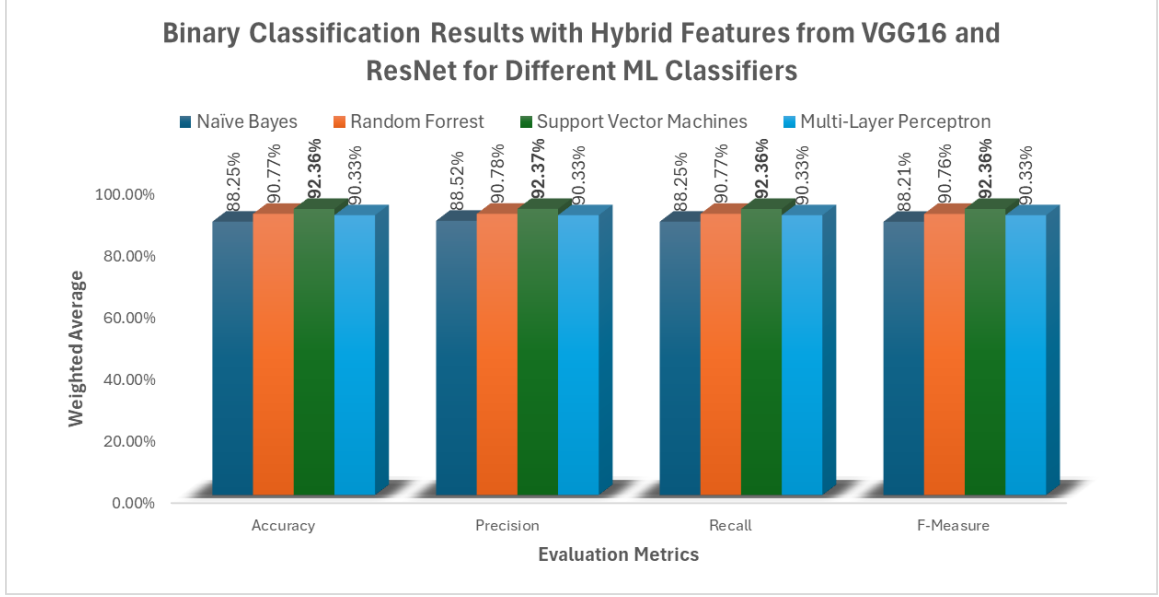


Figure 4.5: Binary Classification Results with Hybrid Features from VGG16 and ResNet for Different ML Classifiers

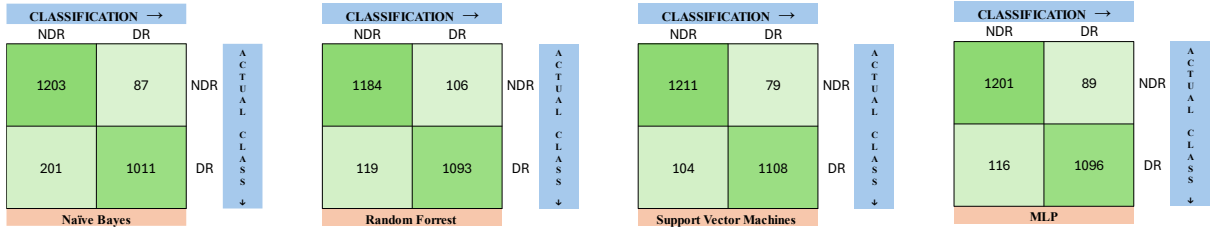


Figure 4.6: Confusion Matrix for Binary Classification using Hybrid Features Extracted from VGG16 + EfficientNet Models for Different Classifiers

4.3.3 Binary DR Classification using localized Divide and Conquer approach

In this section, the results of using the memory efficient localized divide and conquer approach are discussed. The input images from the DDR dataset are passed through our proposed pre-processing framework. This generates images of size $672 \times 672 \times 3$. To extract patches from the high-resolution image, the scaling factor k is set to nine. Figure 4.3 shows the result of extracting the image patches from the PDR fundus image. Each of the patch has a dimension of $224 \times 224 \times 3$ that fits the input size of different CNN models that are used to extract features from the patches. These features are combined to create a feature vector that is passed on to train the ML classifiers. Table 4.11 shows the results of using GoogleNet

to train the resized fundus images and extract features from the patches. The GoogleNet CNN is modified by adding additional fully connected layer of size n . Afterwards, the CNN is retrained using the training images of the DDR dataset. Next, we freeze the last layer of GoogleNet using TL and extract features for each patch. These features are combined and fed to the different classifiers. SVM performs best between the different ML classifiers with an accuracy, precision, recall, and f-measure of **95.92%, 96.11%, 95.92%, 96.11%**.

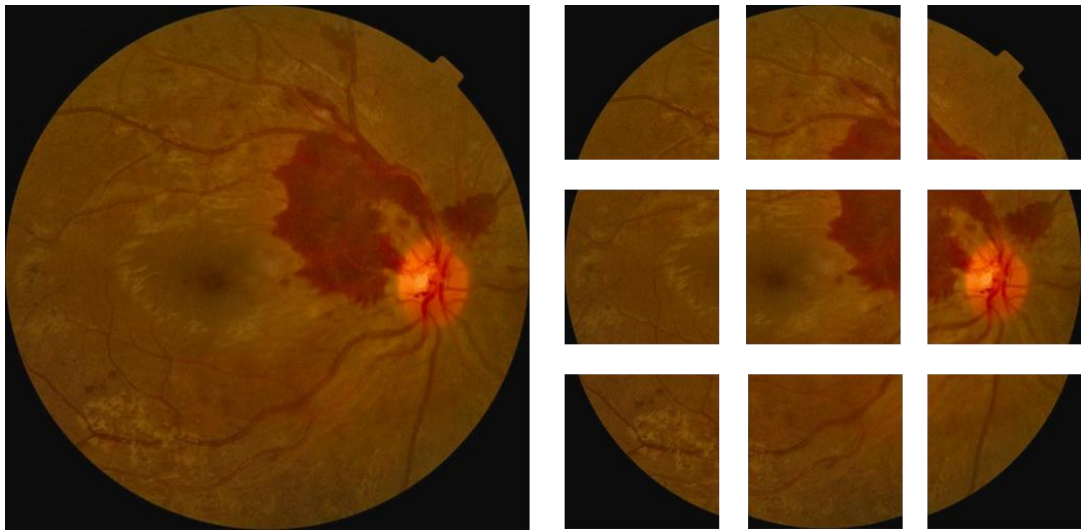


Figure 4.7: Results from Binary DR Classification using Divide and Conquer Approach, Left: PDR Processed Image, Right: k Patches Extracted from the Image

Table 4.11: Experimental Results of ML Classifiers for Binary Classification using Proposed Divide and Conquer Framework on the GoogleNet Model

Classifier	Metrics	Average without Divide and Conquer	Average with Divide and Conquer
Naïve Bayes	Accuracy	82.19%	92.29%
	Precision	82.33%	92.55%
	Recall	82.19%	92.29%
	F-Measure	81.99%	92.51%

Table 4.11: continued

Random Forrest	Accuracy	84.53%	94.44%
	Precision	84.57%	94.98%
	Recall	84.53%	94.44%
	F-Measure	84.52%	94.77%
Support Vector Machines	Accuracy	84.73%	95.92%
	Precision	84.82%	96.11%
	Recall	84.73%	95.92%
	F-Measure	84.71%	96.11%
Multilayer Perceptron	Accuracy	83.65%	95.12%
	Precision	83.75%	95.15%
	Recall	83.65%	95.12%
	F-Measure	83.66%	95.19%

Next, we discuss the results of using this approach on the ResNet CNN model. The network is retrained using the DDR dataset. Extra layers are added to the network to extract 111 features from each patch of the image. These patches are combined to create a feature vector of 999 features for each image. Comparison between the proposed method is provided in Table 4.12.

Table 4.12: Experimental Results of ML Classifiers for Binary Classification using Proposed Divide and Conquer Framework on the ResNet Model

Classifier	Metrics	Average without Divide and Conquer	Average with Divide and Conquer
Naïve Bayes	Accuracy	80.82%	95.88%
	Precision	80.84%	95.88%
	Recall	80.82%	95.88%
	F-Measure	80.80%	95.87%
Random Forrest	Accuracy	85.53%	96.88%
	Precision	85.55%	96.92%
	Recall	85.53%	96.88%
	F-Measure	85.52%	96.96%
Support Vector Machines	Accuracy	88.65%	97.60%
	Precision	88.65%	97.72%
	Recall	88.65%	97.60%
	F-Measure	88.65%	97.75%
Multilayer Perceptron	Accuracy	88.13%	96.48%
	Precision	88.13%	96.63%
	Recall	88.13%	96.48%
	F-Measure	88.13%	96.65%

Afterwards, the test results of extracting features from EfficientNet and VGG16 are presented in Table 4.13 and 4.14 respectively. Both the architectures are retrained with extra layers added at the end to extract features from the patches. The 1000 features from original pre-processed images are also extracted. Both the feature vectors are used to train the different ML classifiers and results show that SVM performs best among all the classifiers tested.

Table 4.13: Experimental Results of ML Classifiers for Binary Classification using Proposed Divide and Conquer Framework on the EfficientNet Model

Classifier	Metrics	Average without Divide and Conquer	Average with Divide and Conquer
Naïve Bayes	Accuracy	82.33%	93.45%
	Precision	82.35%	93.51%
	Recall	82.33%	93.45%
	F-Measure	82.32%	93.54%
Random Forrest	Accuracy	84.89%	95.08%
	Precision	84.89%	95.32%
	Recall	84.89%	95.08%
	F-Measure	84.89%	95.30%

Table 4.13: continued

Support Vector Machines	Accuracy	89.45%	95.80%
	Precision	89.47%	96.14%
	Recall	89.45%	95.80%
	F-Measure	89.44%	96.06%
Multilayer Perceptron	Accuracy	88.93%	95.16%
	Precision	88.96%	95.43%
	Recall	88.93%	95.16%
	F-Measure	88.92%	95.39%

Table 4.14: Experimental Results of ML Classifiers for Binary Classification using Proposed Divide and Conquer Framework on the VGG16 Model

Classifier	Metrics	Average without Divide and Conquer	Average with Divide and Conquer
Naïve Bayes	Accuracy	82.33%	91.49%
	Precision	82.36%	91.49%
	Recall	82.27%	91.49%
	F-Measure	82.29%	91.46%

Table 4.14: continued

Random Forrest	Accuracy	84.89%	94.80%
	Precision	84.89%	94.97%
	Recall	84.86%	94.80%
	F-Measure	84.87%	94.98%
Support Vector Machines	Accuracy	89.45%	95.72%
	Precision	89.50%	95.93%
	Recall	89.39%	95.72%
	F-Measure	89.43%	95.92%
Multilayer Perceptron	Accuracy	88.93%	95.48%
	Precision	88.98%	95.70%
	Recall	88.87%	95.48%
	F-Measure	88.90%	95.69%

The divide and conquer approach can extract more meaningful features from the fundus images of the eye which results in high performance gains. The results show that when DDR dataset is used to train and extract features from the different models, the performance of ML classifiers improves as compared to simple approach. From all the CNN models tested, the ResNet model can extract the most relevant features from the individual patches and achieves a maximum performance using SVM classifier. The graphical representation of results acquired using features from the ResNet model are shown in Figure

4.8. The chart shows that all the ML algorithms get a significant performance uplift using the proposed divide and conquer approach. The confusion matrix for the ResNet Model with different classifiers is presented in Figure 4.11. The matrix shows that the SVM trained model achieves maximum performance specifically in detection of the patients affected by DR with 100% accuracy. The features extracted from ResNet Model achieves a maximum accuracy, precision, recall and f-score of 97.60%, 97.72%, 97.60% and 97.75% respectively on the DDR dataset. The confusion matrix also shows that the SVM performs best in identifying the DR class images since there are no images from the PDR class that are wrongly classified. However, the SVM model provides low performance in predicting the NDR class since it contains 60 miss predictions for this class.

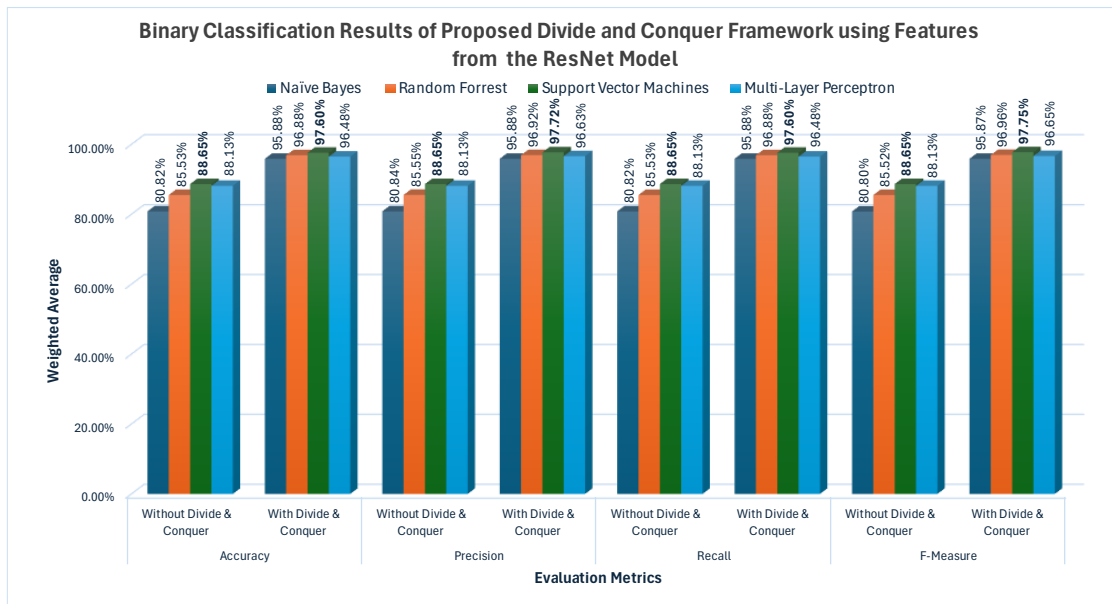


Figure 4.8: Binary Classification Results of Proposed Divide and Conquer Framework using Features from the ResNet Model.

The sample results of correctly classified images in binary classification using divide and conquer approach are shown in Figure 4.9. The incorrectly classified images are also represented in Figure 4.10. The images show that incorrect classification in usually occurs due to small similarities between class 0 and 1 in the ICDR severity scale.

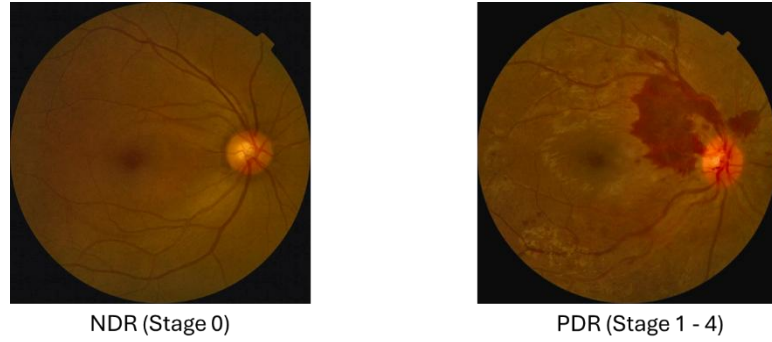


Figure 4.9: Sample Fundus Images from Binary DR Detection Correctly Classified as NDR, and PDR

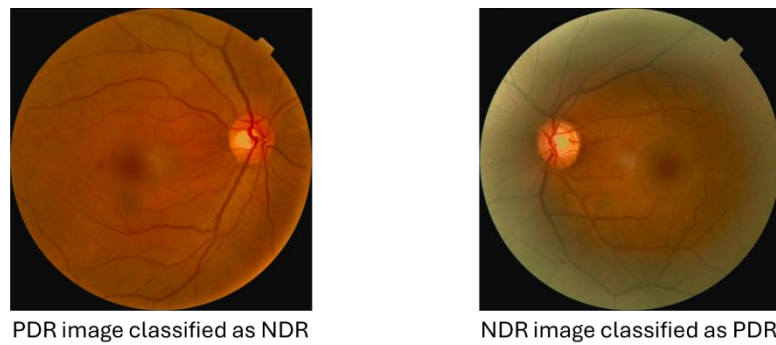


Figure 4.10: Sample Fundus Images from Binary DR Detection Incorrectly Classified as NDR, and PDR

The Table 4.15 shows that best results comparison of our proposed hybrid classification, divide and conquer approach, and other recent works. To further reduce any bias of DDR dataset during the validation phase, we tested the proposed divide and conquer approach by using APTOS dataset to train the best performing CNN model. The ResNet CNN model was retrained using APTOS dataset, but the feature extraction and validation was done on DDR dataset. The results show that the performance reduces by a small percentage but still outperforms all the recent works and the hybrid approach. To further reduce any bias, we also tested divide and conquer approach using the DDR trained ResNet to extract features from APTOS dataset. The results still outperform most of the recent works for binary DR classification.

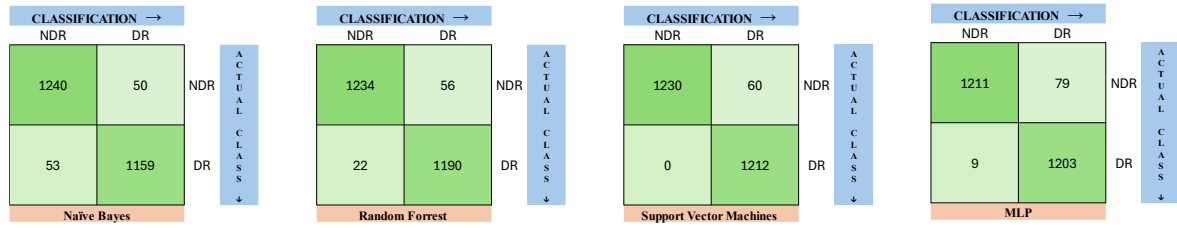


Figure 4.11: Confusion Matrix for Binary Classification using Proposed Divide and Conquer Approach with Features Extracted from ResNet Model

Table 4.15: Comparison of Binary Classification Results of Proposed Frameworks with Recent Works

Method ↓	Dataset	Metrics			
		Accuracy	Precision	Recall	F-Measure
Best Results of Divide and Conquer approach (Training on DDR)	DDR	97.60%	97.72%	97.60%	97.75%
Best Results of Divide and Conquer approach (Training on APTOS)	DDR	96.08%	96.21%	96.19%	96.08%
Best Results of Divide and Conquer approach (Training on DDR)	APTOS	97.39%	97.40%	97.40%	97.40%
Best Results of Hybrid Features approach (DDR Dataset)	DDR	92.69%	92.72%	92.65%	92.67%

Table 4.15: continued

(Padmanayana & K, 2022)	APTOS	94.60%	-	86.00%	-
(Wang et al., 2022)	DDR	88.89%	-	-	-
(Elwin et al., 2022b)	DDR	90.87%	-	92.54%	-
(Vives-Boix & Ruiz-Fernández, 2021)	APTOS	95.56%	96.07%	94.46%	94.24%
(Zhang et al., 2022)	APTOS	91.17%	90.42%	91.71%	90.88%
(Farag et al., 2022)	APTOS	97.00%	-	97.00%	94.55%
(Adriman et al., 2021)	APTOS	96.35%	-	-	-
(Mohanty et al., 2023)	APTOS	97.30%	-	-	-
(El-Ateif & Idri, 2022)	APTOS	90.70%	-	92.80%	-
(Sanjana et al., 2021)	APTOS	86.10%	84.84%	85.43%	85.01%
(Ramasamy et al., 2021b)	APTOS	91.00%	-	90.90%	90.90%

4.4 Multiclass DR Classification using Pre-Processed Fundus Images

For multiclass classification, we first reduce the number of classes from five to three, i.e., NDR (stage 0), MDR (stage 1-2), and PDR (stage 3-4). The class imbalance and interclass similarities between stage 1-2 and stage 3-4 makes it difficult to distinguish between the two classes. The input to these experiments is the pre-processed fundus images from the DDR dataset that are resized to appropriate scale depending on the CNN model used to extract the features. Besides pre-processing, the input fundus images are also passed through data augmentation to create a balanced dataset during the training process of various CNN algorithms. The batch size is set to 32 which allows efficient usage of the GPU's memory resources and 32 epochs are run for each of the CNN algorithm.

4.4.1 Multiclass DR classification using CNN Features extraction and Transfer Learning

In this section, the CNNs are used to extract features from the fundus images that are categorised into three stages. In Table 4.16, the results of multiclass classification using the features extracted from the GoogleNet model are presented. For multiclass DR classification, since the data is unbalanced, we take the weighted average instead of macro average when evaluating different performance metrics. The weighted accuracy of the SVM model is lowered compared to binary classification because of inter-class similarities. Here, Random Forrest and SVM ML classifier provide comparable performance with a weighted accuracy of 76.76% and 76.24% respectively.

Table 4.16: Experimental Results of Different ML Classifiers for Multiclass Classification using Features Extracted from the GoogleNet Model

Classifier	Metrics	NDR	MDR	PDR	Weighted Average
Naïve Bayes	Accuracy	86.76%	50.54%	80.59%	71.41%
	Precision	76.80%	73.30%	48.60%	72.70%
	Recall	86.80%	50.50%	80.60%	71.40%
	F-Measure	81.50%	59.80%	60.60%	70.70%
Random Forrest	Accuracy	87.00%	69.44%	54.43%	76.76%
	Precision	79.20%	72.90%	79.10%	76.60%
	Recall	87.00%	69.40%	54.40%	76.80%
	F-Measure	82.90%	71.10%	64.50%	76.40%
Support Vector Machines	Accuracy	88.84%	66.60%	51.48%	76.24%
	Precision	78.00%	73.10%	78.70%	76.10%
	Recall	88.80%	66.60%	51.50%	76.20%
	F-Measure	83.10%	69.70%	62.20%	75.70%
Multilayer Perceptron	Accuracy	88.60%	66.31%	51.05%	75.96%
	Precision	77.86%	72.72%	78.06%	75.78%
	Recall	88.60%	66.31%	51.05%	75.96%
	F-Measure	82.88%	69.36%	61.73%	75.37%

Experimental results of different classifiers for multiclass classification using features extracted from the ResNet Model are presented in Table 4.17. Here, SVM model outperforms the other classifiers and achieves better classification accuracy compared to GoogleNet model in all individual class metrics. The maximum individual class accuracy is achieved by NDR class in the SVM classifier with a value of 91.57%. For MDR class, SVM classifier also acquires the best classification accuracy of 82.96%. The features extracted from the ResNet model provide better results for all the classes of DR compared to other CNN models.

Table 4.17: Experimental Results of Different ML Classifiers for Multiclass Classification using Features Extracted from the ResNet Model

Classifier	Metrics	NDR	MDR	PDR	Weighted Average
Naïve Bayes	Accuracy	78.65%	69.64%	84.81%	75.56%
	Precision	84.90%	71.40%	56.80%	76.70%
	Recall	78.70%	69.60%	84.80%	75.60%
	F-Measure	81.70%	70.50%	68.00%	75.80%
Random Forrest	Accuracy	89.17%	77.77%	44.73%	80.31%
	Precision	83.00%	75.30%	95.50%	81.00%
	Recall	89.20%	77.80%	44.70%	80.30%
	F-Measure	86.00%	76.50%	60.90%	79.70%

Table 4.17: continued

Support Vector Machines	Accuracy	91.57%	82.96%	76.79%	86.66%
	Precision	87.20%	84.30%	95.30%	86.80%
	Recall	91.60%	83.00%	76.80%	86.70%
	F-Measure	89.40%	83.60%	85.00%	86.60%
Multilayer Perceptron	Accuracy	90.77%	81.78%	71.01%	85.26%
	Precision	86.27%	82.43%	93.37%	85.38%
	Recall	90.77%	81.78%	71.01%	85.23%
	F-Measure	88.46%	82.10%	80.67%	85.13%

The results of using the features extracted from VGG16 model are presented in Table 4.18. SVM model still outperforms the other classifiers. VGG16 model provides better classification results compared to GoogleNet but is unable to get better results compared to ResNet model. The best overall accuracy in VGG16 is achieved using the SVM classifier with a value of 84.78%. The model performs slightly lower in performance than the ResNet and makes a good choice for use in proposed Hybrid features classification method. The results of EfficientNet are also presented in 4.19. EfficientNet performance dropped further compared to VGG16 model for feature extraction and is not selected for hybrid classification.

Table 4.18: Experimental Results of Different ML Classifiers for Multiclass Classification using Features Extracted from the VGG16 Model

Classifier	Metrics	NDR	MDR	PDR	Weighted Average
Naïve Bayes	Accuracy	86.12%	76.40%	86.50%	82.19%
	Precision	87.00%	79.30%	71.20%	82.40%
	Recall	86.10%	76.40%	86.50%	82.20%
	F-Measure	86.60%	77.80%	78.10%	82.20%
Random Forrest	Accuracy	89.81%	80.31%	73.00%	84.35%
	Precision	85.80%	81.20%	91.10%	84.40%
	Recall	89.80%	80.30%	73.00%	84.30%
	F-Measure	87.80%	80.70%	81.00%	84.30%
Support Vector Machines	Accuracy	90.53%	79.92%	75.53%	84.78%
	Precision	85.70%	82.30%	91.30%	84.80%
	Recall	90.50%	79.90%	75.50%	84.80%
	F-Measure	88.10%	81.10%	82.70%	84.70%
Multilayer Perceptron	Accuracy	89.81%	80.17%	74.26%	84.66%
	Precision	85.88%	81.52%	89.34%	84.42%
	Recall	89.81%	80.17%	74.26%	84.39%
	F-Measure	87.80%	80.84%	81.11%	84.32%

Table 4.19: Experimental Results of Different ML Classifiers for Multiclass Classification using Features Extracted from the EfficientNet Model

Classifier	Metrics	NDR	MDR	PDR	Weighted Average
Naïve Bayes	Accuracy	75.20%	53.57%	75.53%	66.41%
	Precision	77.50%	63.60%	41.10%	68.40%
	Recall	75.20%	53.60%	75.50%	66.40%
	F-Measure	76.30%	58.20%	53.30%	66.70%
Random Forrest	Accuracy	86.04%	70.42%	13.92%	72.84%
	Precision	77.60%	66.10%	97.10%	74.70%
	Recall	86.00%	70.40%	13.90%	72.80%
	F-Measure	81.60%	68.20%	24.40%	70.70%
Support Vector Machines	Accuracy	88.76%	76.10%	59.92%	80.87%
	Precision	83.30%	77.00%	85.00%	80.90%
	Recall	88.80%	76.10%	59.90%	80.90%
	F-Measure	85.90%	76.60%	70.30%	80.60%
Multilayer Perceptron	Accuracy	87.96%	75.32%	58.23%	79.99%
	Precision	82.78%	75.91%	82.63%	79.97%
	Recall	87.96%	75.32%	58.23%	79.99%
	F-Measure	85.29%	75.61%	68.32%	79.74%

The results of features extracted from ResNet CNN model used by different ML classifiers are presented in a graphical format in Figure 4.12. The results show that SVM and MLP perform closely whereas, the naïve bayes classifier performs the worst compared to all the ML classifiers that are tested.

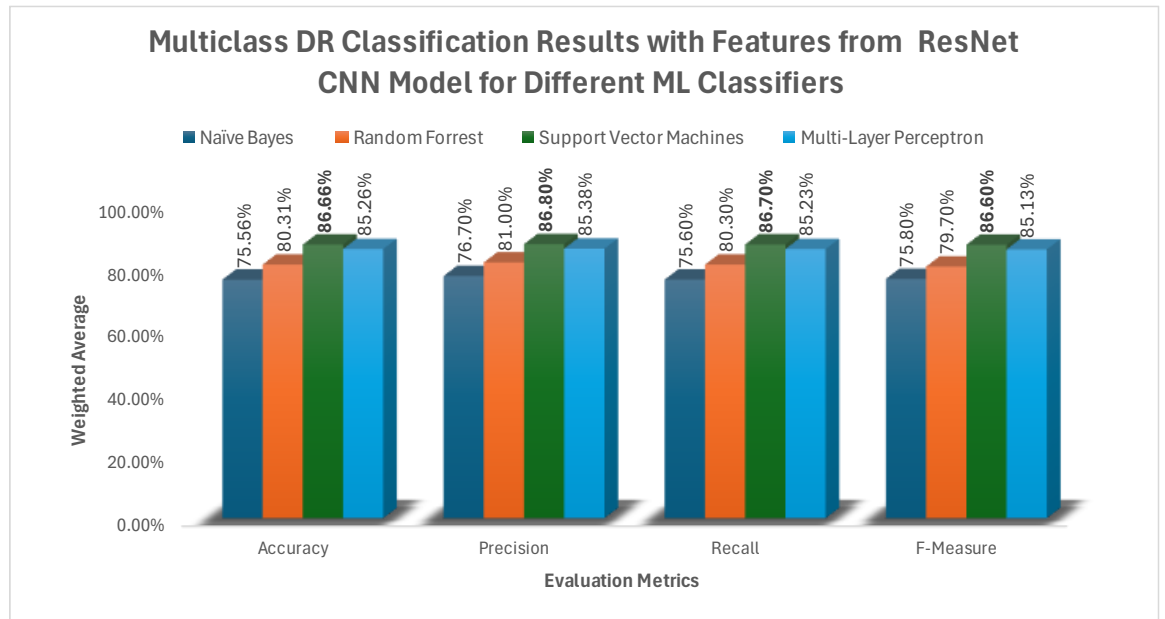


Figure 4.12: Three Class Classification Results with Features from ResNet CNN Model for Different ML Classifiers.

4.4.2 Multiclass DR classification using Hybrid Features extraction and Transfer Learning

Making a hybrid set of features extracted from different models provides significant improvements in the classification. In the previous section, the best individual class performance and overall performance is observed using the features extracted from ResNet CNN model. The VGG16 model provides second best performance for individual feature extraction-based classification. For Hybrid features, we combine the features extracted from these two models according to our proposed framework. The features of EfficientNet are not used in this experiment since its performance is not satisfactory as reported in Table 4.19. The combined features are passed on to different ML classifiers. For performance

measurement, the weighted averages are presented for different classifiers because the data is unbalanced. The SVM classifier outperforms others and provides the highest accuracy of 88.01% using the hybrid features vector extracted from VGG16 and ResNet models. The individual class accuracies are also the highest reported for multiclass SVM classifier with NDR at 92.62% and MDR at 83.64%. However, for PDR class, NB provides the highest performance with an individual class accuracy of 89.87% but it suffers from low precision value of 70.80%. The weighted performance of the remaining parameters is also reported in Table 4.20 with the highest values for SVM classifier having precision at 88.10%, recall at 88.00%, and f-measure also at 88.00%.

Table 4.20: Experimental Results of Different Classifiers for Multiclass Classification using Hybrid Features Extracted from VGG16 and ResNet models

Classifier	Metrics	NDR	MDR	PDR	Weighted Average
Naïve Bayes	Accuracy	85.55%	76.49%	89.87%	82.27%
	Precision	87.40%	79.50%	70.80%	82.60%
	Recall	85.60%	76.50%	89.90%	82.30%
	F-Measure	86.50%	77.90%	79.20%	82.30%
Random Forrest	Accuracy	90.77%	80.71%	70.89%	84.78%
	Precision	85.90%	81.70%	93.30%	84.90%
	Recall	90.80%	80.70%	70.90%	84.80%
	F-Measure	88.30%	81.20%	80.60%	84.70%
Support Vector Machines	Accuracy	92.62%	83.64%	82.70%	88.02%
	Precision	88.00%	86.50%	95.60%	88.10%
	Recall	92.60%	83.60%	82.70%	88.00%
	F-Measure	90.20%	85.10%	88.70%	88.00%
Multilayer Perceptron	Accuracy	91.81%	82.37%	81.86%	87.02%
	Precision	87.26%	85.29%	93.72%	87.07%
	Recall	91.81%	82.37%	81.86%	87.02%
	F-Measure	89.48%	83.81%	87.39%	86.97%

The results are also presented in a graphical format in Figure 4.13. The confusion matrix for multiclass classification using different ML algorithms for the hybrid features classification is depicted in Figure 4.14. The results show that SVM classifier performs best for the NDR and MDR class. Another thing worth noting is that most of the misclassifications occur between NDR and MDR class.

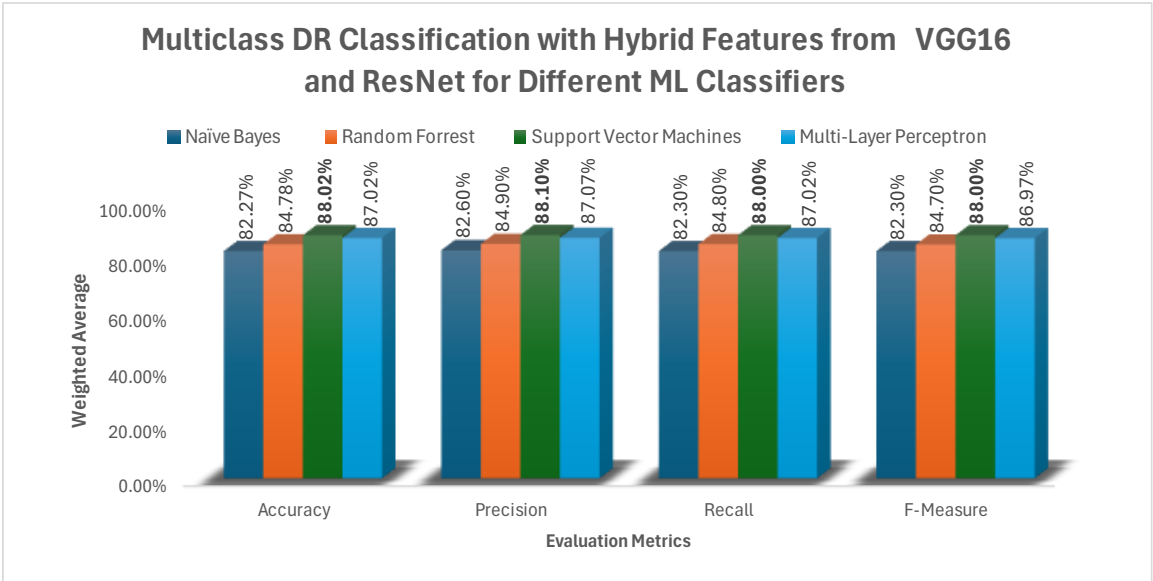


Figure 4.13: Three Class Classification with Hybrid Features from VGG16 and ResNet for Different ML Classifiers.

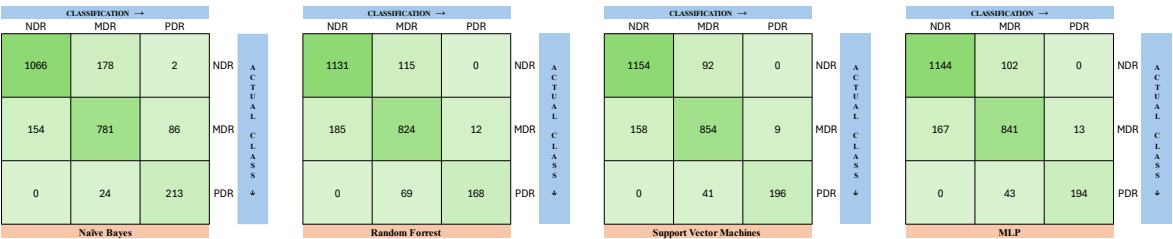


Figure 4.14: Confusion Matrix for Multiclass Classification using Hybrid Features Vector on Different ML Classifiers

4.4.3 Multiclass DR Classification using Localized Divide and Conquer Approach.

In this section, the results of using the memory efficient localized divide and conquer approach for multiclass classification are discussed. The input images from the DDR dataset

are passed through our proposed pre-processing framework. This generates images of size $672 \times 672 \times 3$. To extract patches from the high-resolution image, the scaling factor k is set to nine. Different CNN models are used to extract features from the patches of the image. These features are combined to create a feature vector that is passed on to train the ML classifiers. Table 4.21 shows the results of using GoogleNet for training the resized fundus images and for extracting features from the patches using the proposed framework. The GoogleNet model is modified, and fully connected layer is added at the end to extract 111 features from each patch of the input image. These features are combined and fed to different ML classifiers. For GoogleNet model, SVM, RF and MLP performs very close with an accuracy of 81.07%, 80.91%, and 80.67% respectively.

Table 4.21: Experimental Results of ML Classifiers for Multiclass Classification using Proposed Divide and Conquer Framework on the GoogleNet Model

Classifier	Metrics	Weighted Average without Divide & Conquer	Weighted Average with Divide & Conquer
Naïve Bayes	Accuracy	71.41%	77.52%
	Precision	72.70%	78.65%
	Recall	71.40%	77.52%
	F-Measure	70.70%	76.79%
Random Forrest	Accuracy	76.76%	81.07%
	Precision	76.60%	81.13%
	Recall	76.80%	81.11%
	F-Measure	76.40%	80.72%
Support Vector Machines	Accuracy	76.24%	80.91%
	Precision	76.10%	81.39%
	Recall	76.20%	81.25%
	F-Measure	75.70%	80.92%
Multilayer Perceptron	Accuracy	75.96%	80.67%
	Precision	75.78%	81.12%
	Recall	75.96%	81.01%
	F-Measure	75.37%	80.67%

Next, we discuss the results of using this approach on the ResNet CNN model. The 111 features extracted from each patch of the image are combined to create the feature vector for each image and used to train the ML classifiers. Comparison between the proposed method is provided in Table 4.22.

Table 4.22: Experimental Results of ML Classifiers for Multiclass Classification using Proposed Divide and Conquer Framework on the ResNet Model

Classifier	Metrics	Weighted Average without Divide and Conquer	Weighted Average with Divide and Conquer
Naïve Bayes	Accuracy	75.56%	82.83%
	Precision	76.70%	83.15%
	Recall	75.60%	82.83%
	F-Measure	75.80%	82.85%
Random Forrest	Accuracy	80.31%	85.86%
	Precision	81.00%	86.01%
	Recall	80.30%	85.90%
	F-Measure	79.70%	85.81%
Support Vector Machines	Accuracy	86.66%	89.46%
	Precision	86.80%	89.57%
	Recall	86.70%	89.46%
	F-Measure	86.60%	89.39%

Table 4.22: continued

Multilayer Perceptron	Accuracy	85.26%	88.46%
	Precision	85.38%	88.57%
	Recall	85.23%	88.46%
	F-Measure	85.13%	88.39%

The ResNet with divide and conquer approach achieves the best performance compared to other CNN models. The test results of extracting features from VGG16 are also presented in Table 4.23. The VGG16 CNN model is customised and retrained with extra layers added similar to previous CNN models.

Table 4.23: Experimental Results of ML Classifiers for Multiclass Classification using Proposed Divide and Conquer Framework on the VGG16 Model

Classifier	Metrics	Weighted Average without Divide and Conquer	Weighted Average with Divide and Conquer
Naïve Bayes	Accuracy	82.19%	83.07%
	Precision	82.39%	83.26%
	Recall	82.19%	83.07%
	F-Measure	82.21%	83.09%

Table 4.23: continued

Random Forrest	Accuracy	84.35%	85.62%
	Precision	84.42%	85.70%
	Recall	84.35%	85.62%
	F-Measure	84.27%	85.56%
Support Vector Machines	Accuracy	84.78%	86.54%
	Precision	84.84%	86.41%
	Recall	84.78%	86.36%
	F-Measure	84.70%	86.23%
Multilayer Perceptron	Accuracy	84.66%	86.22%
	Precision	84.43%	86.27%
	Recall	84.41%	86.22%
	F-Measure	84.33%	86.13%

From multiple CNN tested, the features extracted from ResNet model along with SVM model performs the best. The per-class performance of ResNet features and SVM model are represented in the Table 4.24. The results of features extracted from ResNet model are also represented in graphical format in Figure 4.15. The divide and conquer approach show approximately three to four percent performance improvement using the RF, SVM, and MLP classifiers. However, for NB classifier, it shows a large improvement of around seven percent but still NB performance is very low compared to other state of the art ML classifiers tested.

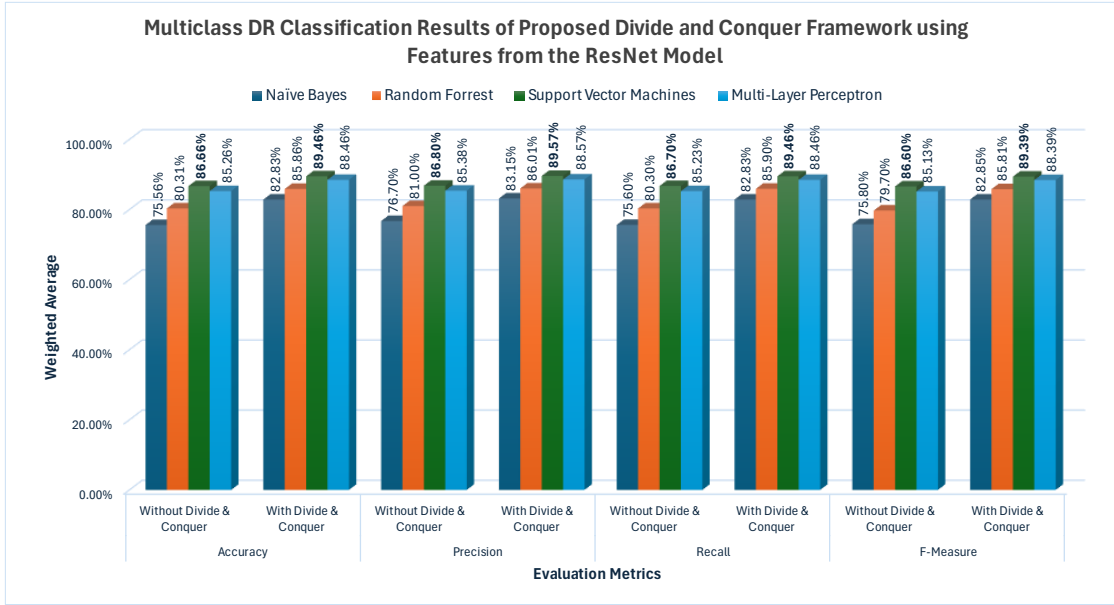


Figure 4.15: Multiclass DR Classification Results of Proposed Divide and Conquer Framework using Features from the ResNet Model.

The confusion matrix of different ML Classifiers using is represented in Figure 4.16. Most of the miss-predictions are in the second DR class i.e. MDR, with some predictions classifying these images as NDR and some classifying them as PDR. This is one of the reason the F-Measure and precision of MDR class is lowest among all the other classes.

Table 4.24: Detailed Experimental Results of SVM Classifier for Multiclass Classification using Proposed Divide and Conquer Framework with Features Extracted from ResNet Model

Classifier	Metrics	NDR	MDR	PDR	Weighted Average
Support Vector Machines	Accuracy	94.62%	84.62%	83.12%	89.46%
	Precision	88.65%	89.07%	96.57%	89.57%
	Recall	94.62%	84.62%	83.12%	89.46%
	F-Measure	91.54%	86.79%	89.34%	89.39%

CLASSIFICATION →															
NDR	MDR	PDR		NDR	MDR	PDR		NDR	MDR	PDR		NDR	MDR	PDR	
1074	161	11	NDR	1141	103	1	NDR	1179	67	0	NDR	1169	77	0	NDR
156	786	79	MDR	179	831	11	MDR	150	864	7	MDR	156	857	8	MDR
2	21	214	PDR	3	56	178	PDR	1	39	197	PDR	2	46	189	PDR
Naïve Bayes				Random Forrest				Support Vector Machines				MLP			

Figure 4.16: Confusion Matrix for Multiclass Classification using Proposed Divide and Conquer Approach with Features Extracted from ResNet Model

The sample of fundus images correctly classified as NDR, MDR, and PDR are shown in Figure 4.17. The incorrectly classified images are also represented in Figure 4.18. The major reason of incorrect classification is the inter-class similarities and badly captured images by the ophthalmologist.



NDR (Stage 0)



MDR (Stage 1+2)



PDR (Stage 3+4)

Figure 4.17: Sample Fundus Images from Multiclass DR Detection Correctly Classified as NDR, MDR and PDR



Figure 4.18: Sample Fundus Images from Multiclass DR Detection Incorrectly Classified as NDR, MDR and PDR

The Table 4.25 shows that best results comparison of our proposed hybrid classification, divide and conquer approach, and other recent works. The results indicate that the proposed divide and conquer framework provides significant performance improvement over the recent classification approaches for DR classification into three stages. To further reduce the bias towards DDR dataset during training for the divide and conquer approach, we tested the divide and conquer approach on APTOS dataset as well with training on DDR dataset. This reduces the performance of the system by a small margin but still gives good performance compared to recent literature. These results indicate that our framework is not biased towards a specific dataset and can perform well for images from different datasets. The multiclass classification by combining stage 1-2 and stage 3-4 is a novel approach that has limited research in the literature. However, as seen from the results, we can get better classification results using three stage classification compared with full five stage DR classification. The patients with MDR can be recommended for minor scans and referral to trained ophthalmologists while patients in the PDR category can be referred with urgent need of attention to the medical practitioners.

Table 4.25: Comparison of Multiclass Classification Results of Proposed Frameworks with Recent Works

Comparison of Multiclass DR Classification	Dataset	Metrics			
Method ↓		Accuracy	Precision	Recall	F- Measure
Best Results of Divide and Conquer approach (Training on DDR)	DDR	89.46%	89.57%	89.46%	89.39%
Best Results of Divide and Conquer approach (Training on DDR)	APTOS	89.19%	89.30%	89.20%	89.20%
Best Results of Hybrid Features approach (DDR Dataset)	DDR	88.02%	88.10%	88.00%	88.00%
(El Houby, 2021)	EyePACS (3-Class)	76.40%	70.23%	69.6%	67.94%
(Lam et al., 2018)	EyePACS (3-Class)	68.80%	-	-	-
(Chowdhury et al., 2020)	EyePACS (3-Class)	60.30%	-	-	-

4.5 Five Stage Multiclass DR Classification using Pre-Processed Fundus Images

In this section, we classify the fundus images into complete five stages specified in the ICDR severity scale. We use the representation NDR for images having no detectable DR (stage 0), MiDR for mild DR images (stage 1), MoDR for fundus images suffering from moderate levels of DR (stage 2), SDR for fundus images having severe levels of DR, and PDR representation is used for the final stage of patients suffering from proliferative DR (stage 4). The inter-class similarities between stage 1-2 and stage 3-4 makes it very difficult to extract useful features for DR image classification. However, using our proposed hybrid technique and divide and conquer approach, we can achieve good improvement in overall performance of the multiclass DR detection framework. For this experimentation, the pre-processed fundus images from three different datasets i.e., DDR, APTOS, and EyePACS are combined to create balanced dataset for training as mentioned in the dataset description in the previous chapter. This is done to equalise the class imbalance between different DR classes during training. Different CNN algorithms are trained with batch size set to 32 which allows efficient usage of the GPU's memory resources and 32 epochs. The trained models are used to extract features that are used for five class DR classification.

4.5.1 Five Stage Multiclass DR classification using CNN Features extraction and Transfer Learning

In this section, the CNNs are used to extract features from the fundus images that are categorised into five stages. In Table 4.26, the results of five stage classification using the features extracted from the GoogleNet model are presented. The weighted accuracy of the SVM model is further reduced compared to binary and three class classification because of interclass dependability between stages 1-2 and 3-4. The results show that SVM outperforms the other models with a highest accuracy, precision, recall and f-score of 70.32%, 70.34%, 70.32%, and 69.89% respectively. PDR class has the lowest individual accuracy of 53.59%.

Table 4.26: Experimental Results of Different ML Classifiers for Five Stage DR Classification using Features Extracted from the GoogleNet Model

Classifier	Metrics	NDR	MiDR	MoDR	SDR	PDR	We. Average
Naïve Bayes	Accuracy	76.58%	52.27%	71.21%	57.96%	42.03%	60.41%
	Precision	80.00%	55.24%	55.04%	53.85%	58.29%	60.84%
	Recall	76.58%	52.27%	71.21%	57.96%	42.03%	60.41%
	F-Measure	78.25%	53.72%	62.09%	55.83%	48.84%	60.12%
Random Forrest	Accuracy	85.87%	56.61%	78.50%	63.91%	53.59%	68.16%
	Precision	81.77%	64.47%	61.97%	62.84%	69.51%	68.35%
	Recall	85.87%	56.61%	78.50%	63.91%	53.59%	68.16%
	F-Measure	83.77%	60.29%	69.26%	63.37%	60.52%	67.79%

Table 4.26: continued

Support Vector Machines	Accuracy	89.78%	59.63%	80.50%	64.54%	54.69%	70.32%
	Precision	82.99%	67.76%	65.27%	63.87%	70.62%	70.34%
	Recall	89.78%	59.63%	80.50%	64.54%	54.69%	70.32%
	F-Measure	86.25%	63.44%	72.09%	64.20%	61.64%	69.89%
Multilayer Perceptron	Accuracy	77.28%	60.37%	36.52%	66.03%	75.15%	63.07%
	Precision	63.85%	47.73%	50.26%	77.56%	79.79%	63.69%
	Recall	77.28%	60.37%	36.52%	66.03%	75.15%	63.07%
	F-Measure	69.92%	53.31%	42.30%	71.33%	77.40%	62.77%

Experimental results of different classifiers for five stage classification of DR images using features extracted from the ResNet Model are presented in Table 4.27. Here, SVM model again outperforms the other classifiers and achieves better classification accuracy compared to GoogleNet model in all individual class metrics. The maximum individual class accuracy is achieved by NDR class in the SVM classifier with a value of 92.94%. For MiDR, MoDR, SDR and PDR classes SVM classifier also acquires the best classification accuracy of 72.31%, 85.77%, 75.00% and 71.12% respectively. SVM classifier achieves a maximum weighted average accuracy, precision, recall and f-score of 79.77%, 79.93%, 79.77% and 79.68% respectively.

Table 4.27: Experimental Results of Different ML Classifiers for Five Stage DR Classification using Features Extracted from the ResNet Model

Classifier	Metrics	NDR	MiDR	MoDR	SDR	PDR	We. Average
Naïve Bayes	Accuracy	77.32%	54.75%	75.38%	60.81%	43.43%	62.72%
	Precision	80.31%	58.50%	57.56%	56.50%	61.24%	63.14%
	Recall	77.32%	54.75%	75.38%	60.81%	43.43%	62.72%
	F-Measure	78.79%	56.56%	65.28%	58.57%	50.82%	62.35%
Random Forrest	Accuracy	89.41%	61.16%	80.58%	66.31%	54.18%	70.79%
	Precision	82.36%	69.98%	65.16%	64.67%	71.20%	70.88%
	Recall	89.41%	61.16%	80.58%	66.31%	54.18%	70.79%
	F-Measure	85.74%	65.27%	72.06%	65.48%	61.54%	70.34%
Support Vector Machines	Accuracy	92.94%	72.31%	85.77%	75.00%	71.12%	79.77%
	Precision	91.41%	79.01%	74.83%	72.69%	80.59%	79.93%
	Recall	92.94%	72.31%	85.77%	75.00%	71.12%	79.77%
	F-Measure	92.17%	75.51%	79.93%	73.83%	75.56%	79.68%
Multilayer Perceptron	Accuracy	92.84%	63.82%	71.37%	62.43%	62.62%	71.09%
	Precision	82.51%	67.86%	66.36%	68.26%	66.32%	70.45%
	Recall	92.84%	63.82%	71.37%	62.43%	62.62%	71.09%
	F-Measure	87.37%	65.78%	68.77%	65.21%	64.42%	70.64%

The results of using the features extracted from VGG16 model are presented in Table 4.28. SVM model still outperforms the other classifiers. VGG16 model provides better classification results compared to GoogleNet and ResNet in case of five stage multiclass DR classification. The best weighted accuracy in VGG16 is achieved using the SVM classifier with a value of 80.84%.

Table 4.28: Experimental Results of Different ML Classifiers for Five Stage DR Classification using Features Extracted from the VGG16 Model

Classifier	Metrics	NDR	MiDR	MoDR	SDR	PDR	We. Average
Naïve Bayes	Accuracy	89.46%	68.32%	75.55%	73.76%	61.63%	74.02%
	Precision	83.18%	71.88%	69.85%	68.83%	75.06%	73.94%
	Recall	89.46%	68.32%	75.55%	73.76%	61.63%	74.02%
	F-Measure	86.21%	70.05%	72.59%	71.21%	67.69%	73.78%
Random Forrest	Accuracy	92.01%	74.59%	85.77%	74.58%	71.31%	79.97%
	Precision	90.66%	76.32%	77.03%	73.79%	81.18%	80.03%
	Recall	92.01%	74.59%	85.77%	74.58%	71.31%	79.97%
	F-Measure	91.33%	75.44%	81.16%	74.18%	75.93%	79.88%

Table 4.28: continued

Support Vector Machines	Accuracy	93.12%	75.41%	86.15%	75.42%	72.51%	80.84%
	Precision	91.26%	77.17%	78.32%	74.48%	81.98%	80.87%
	Recall	93.12%	75.41%	86.15%	75.42%	72.51%	80.84%
	F-Measure	92.18%	76.28%	82.05%	74.95%	76.96%	80.76%
Multilayer Perceptron	Accuracy	93.31%	65.70%	83.08%	76.91%	62.15%	76.59%
	Precision	90.45%	75.71%	70.36%	66.97%	81.04%	77.18%
	Recall	93.31%	65.70%	83.08%	76.91%	62.15%	76.59%
	F-Measure	91.86%	70.35%	76.19%	71.60%	70.35%	76.39%

The graphical representation of results using features extracted from VGG16 model are shown in Figure 4.19. The results indicate that Random Forrest classifier performs closely with the SVM classifier. This can happen due to the ensemble nature of Random Forrest classifier that is able to capture better intrinsic details between the various classes compared to MLP in the five stage DR classification.

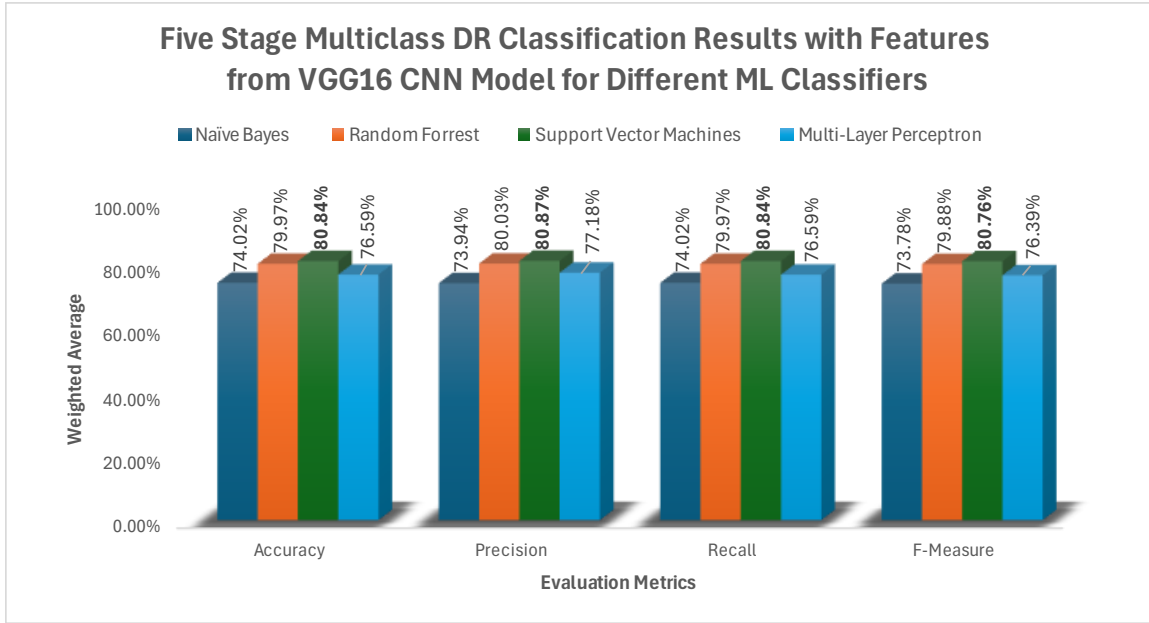


Figure 4.19: Five Stage Multiclass DR Classification Results with Features from VGG16 CNN Model for Different ML Classifiers.

4.5.2 Five Stage Multiclass DR classification using Hybrid Features extraction and Transfer Learning

Making a hybrid set of features extracted from different models provides significant improvements in the classification. In the previous section, the best individual class and overall performance for five stage DR classification is observed using the features extracted from VGG16 CNN model. The ResNet model provides second best performance for individual feature extraction-based classification. For Hybrid features, we combine the features extracted from these two models according to our proposed framework. The combined features are passed on to different ML classifiers. The weighted averages and per-class performance for different classifiers using hybrid features is presented in Table 4.29. The SVM classifier outperforms others and provides the highest accuracy of 83.62% using the hybrid features vector. The individual class accuracies are also the highest reported for SVM classifiers with NDR at 95.17%, MiDR at 80.37%, MoDR at 87.12%, SDR at 78.18%, and PDR at 75.90%. However, for MoDR class, NB provides the highest performance with

an individual class accuracy of 90.58% but with low precision of 72.13%. The weighted performance of the remaining parameters is also reported at the highest values for SVM classifier with precision at 83.76%, recall at 83.62%, and f-measure at 83.59%.

Table 4.29: Experimental Results of Different Classifiers for Five Stage Multiclass DR Classification using Hybrid Features Extracted from VGG16 and ResNet Models

Classifier	Metrics	NDR	MiDR	MoDR	SDR	PDR	We. Average
Naïve Bayes	Accuracy	93.31%	65.70%	90.58%	76.91%	62.15%	78.14%
	Precision	90.61%	79.90%	72.13%	68.49%	81.89%	78.84%
	Recall	93.31%	65.70%	90.58%	76.91%	62.15%	78.14%
	F-Measure	91.94%	72.11%	80.31%	72.46%	70.67%	77.82%
Random Forrest	Accuracy	95.54%	76.65%	87.69%	76.48%	73.11%	82.23%
	Precision	91.79%	79.96%	79.86%	75.84%	82.47%	82.20%
	Recall	95.54%	76.65%	87.69%	76.48%	73.11%	82.23%
	F-Measure	93.62%	78.27%	83.59%	76.16%	77.51%	82.11%
Support Vector Machines	Accuracy	95.17%	80.37%	87.12%	78.18%	75.90%	83.62%
	Precision	94.29%	79.71%	80.04%	77.20%	86.39%	83.76%
	Recall	95.17%	80.37%	87.12%	78.18%	75.90%	83.62%
	F-Measure	94.73%	80.04%	83.43%	77.68%	80.81%	83.59%

Table 4.29: continued

Multilayer Perceptron	Accuracy	94.05%	71.49%	85.96%	78.60%	73.90%	81.12%
	Precision	90.20%	78.64%	77.07%	76.02%	83.00%	81.16%
	Recall	94.05%	71.49%	85.96%	78.60%	73.90%	81.12%
	F-Measure	92.08%	74.89%	81.27%	77.29%	78.19%	80.99%

The classification results of using hybrid features vector are also presented in graph format in Figure 4.20. SVM and RF are the best performing ML classifier. The confusion matrix for different classifiers using the hybrid feature vector from VGG16 and ResNet model is presented in Figure 4.21. The confusion matrix shows that most of the miss-classifications occur between stage three and stage four i.e., SDR and PDR. This happens due to the interclass similarity between the classes. However, utilising hybrid features vector gives a good performance improvement of approximately three percent compared to using standalone CNN models.

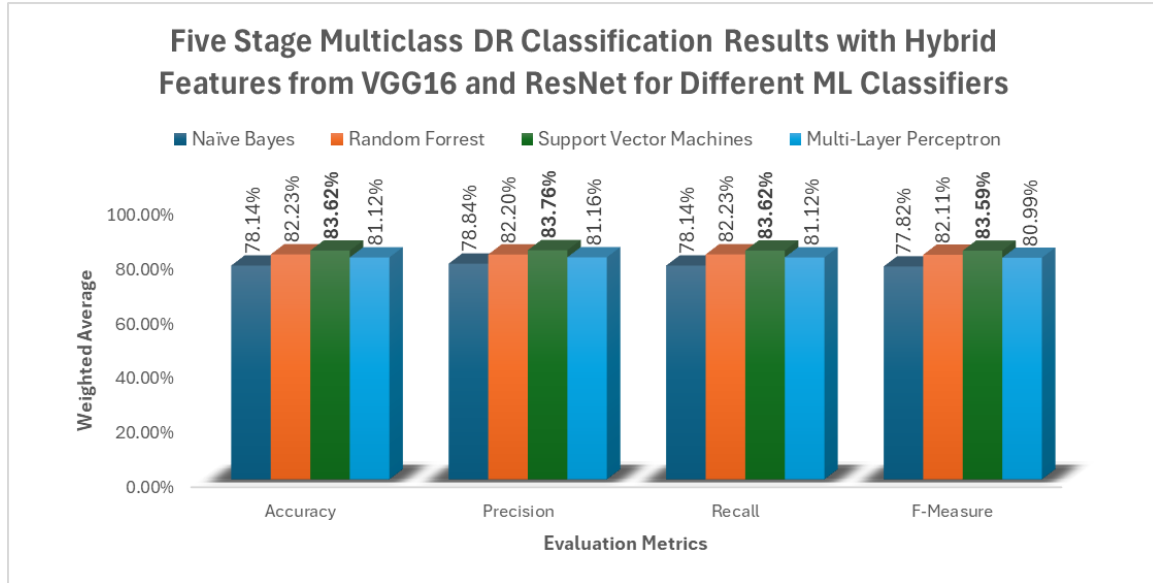


Figure 4.20: Five Stage Multiclass DR Classification Results with Hybrid Features from VGG16 and ResNet for Different ML Classifiers.

CLASSIFICATION →					
NDR	MIDR	MoDR	SDR	PDR	
502	17	19	0	0	NDR
42	318	101	20	3	MIDR
10	15	471	6	18	MoDR
0	33	28	363	48	SDR
0	15	34	141	312	PDR
Naïve Bayes					

CLASSIFICATION →					
NDR	MIDR	MoDR	SDR	PDR	
514	15	9	0	0	NDR
36	371	60	9	8	MIDR
9	30	456	14	11	MoDR
1	34	17	361	59	SDR
0	14	29	92	367	PDR
Random Forrest					

CLASSIFICATION →					
NDR	MIDR	MoDR	SDR	PDR	
512	14	12	0	0	NDR
21	389	59	9	6	MIDR
10	32	453	14	11	MoDR
0	40	20	369	43	SDR
0	13	22	86	381	PDR
Support Vector Machines					

CLASSIFICATION →					
NDR	MIDR	MoDR	SDR	PDR	
506	14	18	0	0	NDR
40	346	71	20	7	MIDR
13	31	447	12	17	MoDR
1	33	15	371	52	SDR
1	16	29	85	371	PDR
MLP					

Figure 4.21: Confusion Matrix for Five Stage Multiclass Classification using Hybrid Features Vector on Different ML Classifiers

4.5.3 Five Stage Multiclass DR Classification using Localized Divide and Conquer Approach.

In this section, the results of using the memory efficient localized divide and conquer approach for five stage multiclass DR classification are discussed. The input images from the DDR dataset are passed through our proposed pre-processing framework. This generates images of size $672 \times 672 \times 3$. To extract patches from the high-resolution image, the scaling factor k is set to nine. Different CNN models are used to extract features from the patches of the image. These features are combined to create a feature vector that is passed on to train the ML classifiers. Table 4.30 shows the results of using GoogleNet for training and extracting features from image patches. The GoogleNet model is modified, and fully connected layer is added at the end to extract 111 features from each patch of the input image. These features are combined and fed to the different classifiers. For GoogleNet model, SVM performs best with an accuracy, precision, recall, f-score of 78.50%, 78.50%, 78.50% and 77.82% respectively.

Table 4.30: Experimental Results of ML Classifiers for Five Stage Multiclass Classification using Proposed Divide and Conquer Framework on the GoogleNet Model

Classifier	Metrics	Weighted Average without Divide and Conquer	Weighted Average with Divide and Conquer
Naïve Bayes	Accuracy	60.41%	62.86%
	Precision	60.84%	63.96%
	Recall	60.41%	62.86%
	F-Measure	60.12%	62.71%

Table 4.30: continued

Random Forrest	Accuracy	68.16%	72.85%
	Precision	68.35%	73.73%
	Recall	68.16%	72.85%
	F-Measure	67.79%	72.40%
Support Vector Machines	Accuracy	70.32%	78.50%
	Precision	70.34%	78.50%
	Recall	70.32%	78.50%
	F-Measure	69.89%	77.82%
Multilayer Perceptron	Accuracy	63.07%	69.63%
	Precision	63.69%	70.32%
	Recall	63.07%	69.63%
	F-Measure	62.77%	69.27%

Next, we discuss the results of using our divide and conquer approach on the ResNet CNN model. The 111 features extracted from each patch of the image are used to train the ML classifiers. Comparison between the proposed method is provided in Table 4.31. ResNet model provides a significant improvement over the regular classification without using the divide and conquer approach. The maximum accuracy, precision, recall and f-score is acquired using the SVM classifier with values of 84.31%, 84.44%, 84.31%, and 84.26% respectively.

Table 4.31: Experimental Results of ML Classifiers for Five Stage Multiclass Classification using Proposed Divide and Conquer Framework on the ResNet Model

Classifier	Metrics	Weighted Average without Divide and Conquer	Weighted Average with Divide and Conquer
Naïve Bayes	Accuracy	62.72%	77.28%
	Precision	63.14%	77.93%
	Recall	62.72%	77.28%
	F-Measure	62.35%	76.97%
Random Forrest	Accuracy	70.79%	80.70%
	Precision	70.88%	80.66%
	Recall	70.79%	80.70%
	F-Measure	70.34%	80.56%
Support Vector Machines	Accuracy	79.77%	84.31%
	Precision	79.93%	84.44%
	Recall	79.77%	84.31%
	F-Measure	79.68%	84.26%
Multilayer Perceptron	Accuracy	71.09%	79.90%
	Precision	70.45%	80.30%
	Recall	71.09%	79.90%
	F-Measure	70.64%	79.64%

The test results of extracting features from VGG16 are also presented in Table 4.31. The VGG16 CNN model is retrained with extra layers added at the end to extract features from the patches. Additional layer is added at the end to extract 111 features for our divide and conquer approach. The results of using the features vector to train the different ML classifiers shows that SVM performs best among all the classifiers tested with a highest accuracy, precision, recall and f-score from all the models tested having values of 85.64%, 85.79%, 85.64%, and 85.58% respectively.

Table 4.32: Experimental Results of ML Classifiers for Five Stage Multiclass Classification using Proposed Divide and Conquer Framework on the VGG16 Model

Classifier	Metrics	Weighted Average without Divide and Conquer	Weighted Average with Divide and Conquer
Naïve Bayes	Accuracy	74.02%	80.14%
	Precision	73.94%	80.06%
	Recall	74.02%	80.14%
	F-Measure	73.78%	79.98%
Random Forrest	Accuracy	79.97%	84.12%
	Precision	80.03%	84.17%
	Recall	79.97%	84.12%
	F-Measure	79.88%	84.06%

Table 4.32: continued

Support Vector Machines	Accuracy	80.84%	85.64%
	Precision	80.87%	85.79%
	Recall	80.84%	85.64%
	F-Measure	80.76%	85.58%
Multilayer Perceptron	Accuracy	76.59%	82.30%
	Precision	77.18%	82.35%
	Recall	76.59%	82.30%
	F-Measure	76.39%	82.19%

From multiple CNN tested, the features extracted from VGG16 model along with SVM model performs the best for five stage DR classification. The ResNet CNN model also performs closely with the VGG16 model. However, both models using our divide and conquer approach provide significant improvements over the regular CNN and feature extraction methods. The per-class performance of features extracted from VGG16 model using the divide and conquer approach along with SVM model are represented in the Table 4.33. The weighted average of different ML classifiers with features extracted from proposed divide and conquer framework is presented in Figure 4.22. It shows an approximately five percentage of performance improvement in all the evaluation metrics tested. In Figure 4.23, the confusion matrix of the ML classifiers using the features extracted from VGG16 CNN are presented. The PDR and the SDR stage show the least performance per-class since in both stages, blood leakage and excessive lesions presence makes it difficult to distinguish the classes. The performance comparison of our proposed techniques with other recent

approaches is presented in Table 4.34. To avoid bias, the divide and conquer approach was also tested on APTOS dataset. The results indicate that our proposed framework provides significant performance improvement in multiclass DR classification compared to recent works.

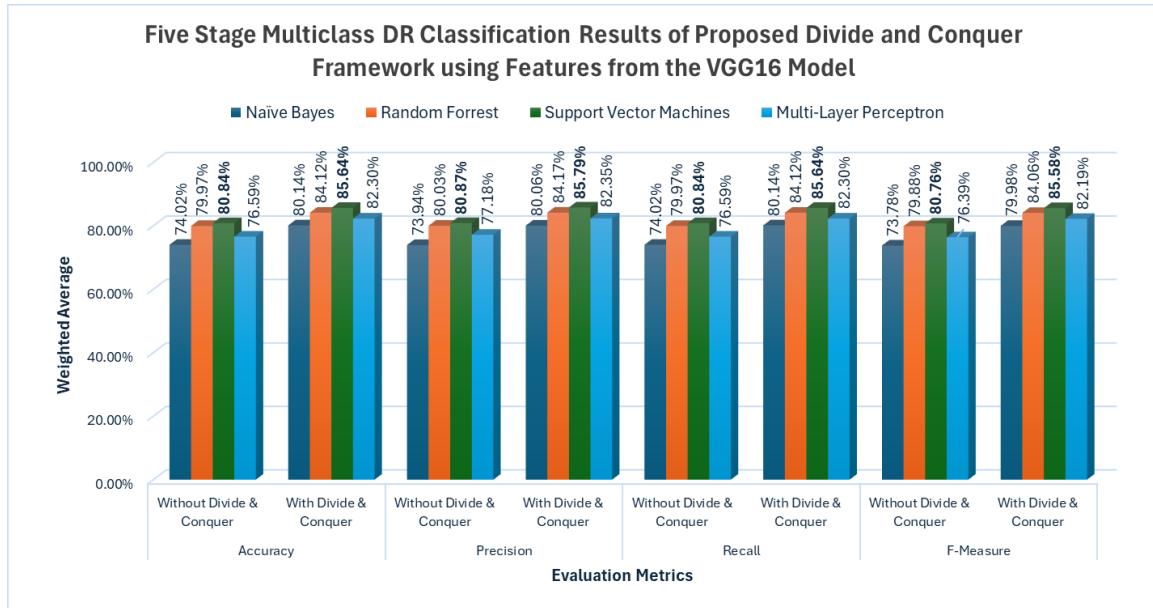


Figure 4.22: Five Stage Multiclass DR Classification Results of Proposed Divide and Conquer Framework using Features from the VGG16 Model.

Table 4.33: Detailed Experimental Results of SVM Classifier for Five Stage Multiclass Classification using Proposed Divide and Conquer framework with Features Extracted from VGG16 Model

Classifier	Metrics	NDR	MiDR	MoDR	SDR	PDR	Wt. Average
SVM	Accuracy	96.25%	83.54%	90.96%	80.30%	75.90%	85.64%
	Precision	95.72%	83.72%	81.69%	79.29%	87.59%	85.79%
	Recall	96.25%	83.54%	90.96%	80.30%	75.90%	85.64%
	F-Measure	95.99%	83.63%	86.08%	79.79%	81.32%	85.58%

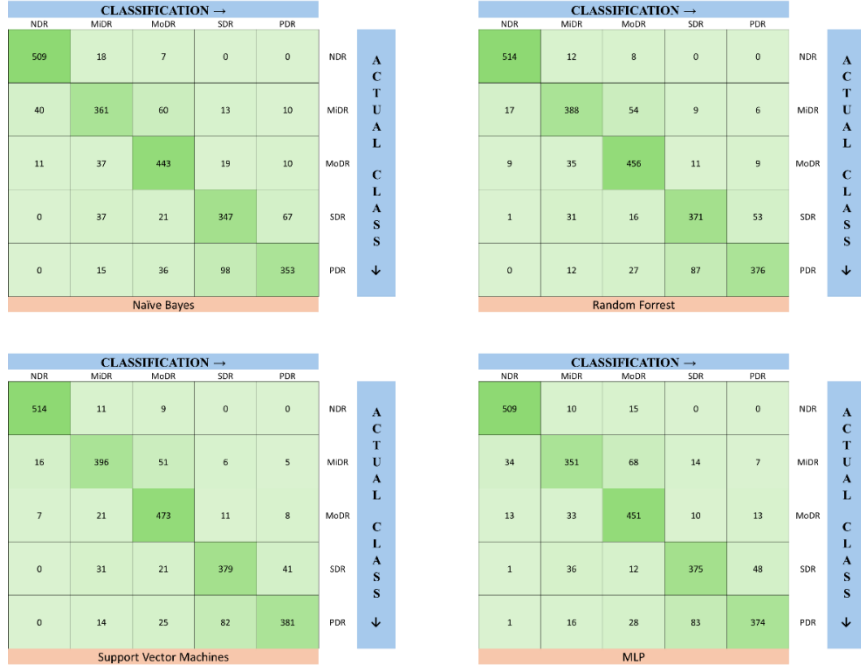


Figure 4.23: Confusion Matrix for Five Stage Multiclass Classification using Proposed Divide and Conquer Approach with Features Extracted from VGG16 Model

The sample of fundus images correctly classified as NDR, MiDR, MoDR, SDR and PDR using the divide and conquer approach are shown in Figure 4.24. The incorrectly classified images are also represented in Figure 4.25. For the five stage DR classification, the main cause of incorrect classification is the inter-class similarities and limited number of images available in the individual classes. This makes it difficult for the CNN models to extract distinguishing features even with high-resolution inputs provided in the divide and conquer approach.

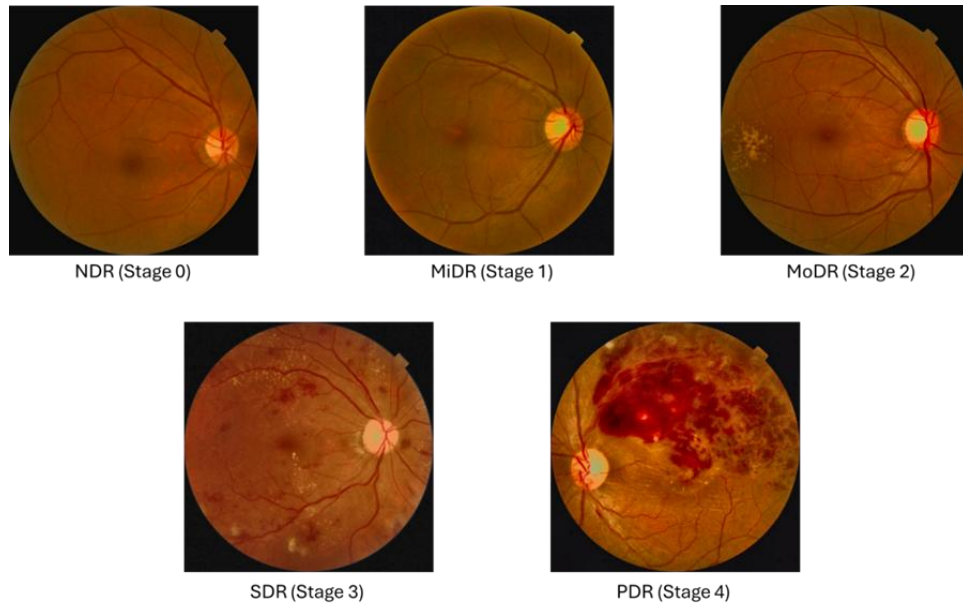


Figure 4.24: Sample Fundus Images from Five Stage DR Detection Correctly Classified as NDR, MiDR, MoDR, SDR, and PDR



Figure 4.25: Sample Fundus Images from Five Stage DR Detection that are Incorrectly Classified

Table 4.34: Comparison of Five Stage Multiclass Classification Results of Proposed Frameworks with Recent Works

Comparison of Multiclass DR Classification	Dataset	Metrics			
		Accuracy	Precision	Recall	F- Measure
Best Results of Divide and Conquer approach (Training on DDR)	DDR	85.64%	85.79%	85.64%	85.58%
Best Results of Divide and Conquer approach (Training on DDR)	APTOS	85.04%	85.34%	85.04%	85.05%
Best Results of Hybrid Features approach (DDR Dataset)	DDR	83.62%	83.76%	83.62%	83.59%
(Gu et al., 2023)	DDR (5 Class)	75.03%	75.39%	75.03%	72.89%
(Saini & Susan, 2022a)	DDR (5-Class)	75.14%	-	-	-

Table 4.34: continued

(Monteiro, 2023)	Mixed	64.50%	-	-	-
(Li et al., 2019a)	DDR (5-Class)	82.84%	-	-	-
(Wang et al., 2022)	DDR (5-Class)	79.04%	-	-	-
(Dekhil et al., 2019)	APTOS (5-Class)	77.00%	-	-	-
(Kassani et al., 2019)	APTOS (5-Class)	83.09%	-	-	-
(Bodapati et al., 2021)	APTOS (5-Class)	82.54%			
(Kobat et al., 2022)	APTOS (5-Class)	84.90%	-	-	72.51%
(Menaouer et al., 2022) Macro Average Results	APTOS (5-Class)	90.60%		95.00%	94.00%
(Menaouer et al., 2022) Weighted Average Results	APTOS (5-Class)	73.22%	69.32%	73.22%	69.92%

4.6 Summary

This chapter has provided results of all the proposed algorithms for DR image classification. The results of pre-processing framework are presented first which shows the improvement in performance on various CNN models. The maximum performance improvement is achieved when using VGG16 model for binary classification with a weighted accuracy of 87.22%. Afterwards, the results of binary classification using the proposed frameworks were presented. The classification system achieves a maximum accuracy of 92.69% on DDR dataset using the proposed hybrid features extraction method with features extracted from VGG16 and EfficientNet models. The Divide and Conquer approach for binary classification on the DDR dataset is tested afterwards. Here, the maximum performance in binary classification achieved was 97.60%, 97.72%, 97.60%, and 97.75% for accuracy, precision, recall and f-score. The comparison with recent research works also shows that our proposed framework can provide better performance for the binary classification of DR from fundus images of the eye. Next, the results of multiclass DR classification were presented. The stages 1-2 and 3-4 were combined into MDR, and PDR class. The results showed that there are also performance gains when using our proposed framework for DR classification using the hybrid features and divide and conquer approach. The Hybrid Features framework achieved a maximum performance using SVM classifier with values of 88.02%, 88.10%, 88.00%, and 88.00% for accuracy, precision, recall and f-measure using the features extracted from ResNet and VGG16 models. The divide and conquer approach again outperform the hybrid features extraction method with values of 89.46%, 89.57%, 89.46%, 89.39% for accuracy, precision, recall and f-measure using the features extracted from modified ResNet model. The comparison with recent works is also given in table 4.25 which indicates that our proposed framework provides good performance

improvement for multiclass DR classifications. The last section of the chapter shows the results of full five class DR classification of fundus images according to the ICDR severity scales. The performance of the system reduces a little in five class classification due to inter-class similarities between stage 1-2 and stage 3-4 of DR classes. However, our results still provide performance improvement when compared to the recent works. The hybrid classification technique using features extracted VGG16 and ResNet was able to achieve performance metrics of 83.62%, 83.76%, 83.62%, and 83.59% for accuracy, precision, recall and f-measure. Lastly, the Divide and Conquer approach using the features from the modified VGG16 model and SVM classifier was able to achieve maximum performance gains with values of 85.64%, 85.79%, 85.64%, and 85.58% for accuracy, precision, recall and f-measure. In summary, the proposed frameworks can provide reasonable performance improvements over the recent work in binary and multiclass DR classification using fundus images of the eye.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

Diabetic Retinopathy is one of the leading causes of blindness in diabetic patients. The disease remains mostly undetected in the early stages. Routine eye checkups play important role in detecting DR and providing appropriate treatment. A study estimates that by 2040, approximately 200 million diabetic patients are expected to have DR (Dai et al., 2021). The ICDR severity scale distributes the DR stages into five different categories. Fundus images help the ophthalmologists to analyse various lesions inside the retina that helps in grading DR. Automatic classification of DR into multiple classes require careful extraction of features that help train ML models for accurate DR classification. The process involves enhancing the fundus images that suffer from various degradations followed by high quality features extraction. The feature vectors are used to train the ML algorithms for classification of DR.

In this chapter, the main contributions of this thesis for DR detection from fundus images of the eyes are discussed. The limitations of the proposed work and future work in this area of research are also presented in this chapter.

5.2 Contributions

One of the main objectives of this thesis was to implement an effective pre-processing technique that can deal with varying quality of fundus images for DR classification. Besides that, another goal of the work was to provide high quality features of fundus images for multiclass DR classification using ML and deep learning methods. Three

frameworks were proposed in this thesis. The first framework implements an effective pre-processing method for fundus image enhancement. The next two frameworks proposed the use of hybrid features, divide and conquer approach, and layer customisation of CNN to achieve a high-performance multiclass DR classification system. The summary of the contributions in this work is given below.

5.2.1 Pre-Processing Framework for Fundus Images of Varying Size and Quality

The pre-processing framework proposed in this work deals with fundus images of varying size and quality. The list of contributions in the pre-processing stage relative to the research question and objective 1 are given below:

- i. For varying image sizes in the fundus images datasets, we apply the Otsu's thresholding method and bounding box to extract the retinal structure from the input images. The retinal structure is then resized to a constant high resolution fixed image of dimension $672 \times 672 \times 3$.
- ii. In the next phase of pre-processing, the changes in illumination and colour of the input fundus images are addressed. The standard deviation of the blue channel of degraded images is higher than the normal fundus images. The illumination and colour of the degraded fundus images is matched with the reference fundus image and passed on to the next phase.
- iii. Next, CLAHE is applied on the fundus images for enhancing the vessel structures and lesions. Gaussian filtering is also applied to create a pre-processed image that is passed on to the next stages for DR classification.

The pre-processing model proposed attains a maximum binary classification accuracy of 87.22% using the standard VGG16 CNN model. The pre-processed images are then passed on to the next stages of the classification frameworks.

5.2.2 DR Image Classification using Hybrid Features

The second framework presented in this work deals with research question 2 and research objective 2. The goal is to extract high quality fundus image features using multiple CNNs that can be used by different ML classifiers for multiclass DR classification. The contributions made in this framework are listed below:

- i. A hybrid features framework for extracting high quality features from the fundus images is proposed. The method uses transfer learning to freeze the initial layers of different CNN models, such as AlexNet, ResNet, GoogleNet, EfficientNet and VGG16, and extract feature vectors from the classification layers of these models.
- ii. The extracted features are combined using a hybrid approach in which the feature vectors of best performing CNN models are combined to create a hybrid feature vector.
- iii. The hybrid feature vector is passed on to different ML classifiers for DR image classification. The high-quality features extracted in this method provide good classification accuracy for multiclass DR classification.

The results of this approach are tested on DDR dataset that is merged in different methods for binary, multiclass (3 Stages) and five stage DR classification. The reported results show maximum weighted accuracy, precision, recall and f-measure of 92.69%, 92.70%, 92.69%, and 92.68% respectively for binary classification of fundus images into NDR (No DR) and PDR (Proliferative DR) stages. The maximum performance metrics in

binary classification are observed using the hybrid features extracted from the VGG16 and EfficientNet CNN models. The multiclass classification is divided into two parts. First the stages 1-2 of the ICDR severity scale are combined into MDR class and afterwards the stages 3-4 of the ICDR severity scale are combined into PDR class for three class DR classification. Features were extracted from VGG16 and ResNet CNN models. The reported results show maximum weighted accuracy, precision, recall and f-measure of 88.02%, 88.10%, 88.00%, and 88.00% respectively for multiclass (3 class) classification of fundus images into NDR, MDR, and PDR classes. Lastly, DR classification for the full ICDR severity scale covering five classes is performed. In this classification, the maximum performance acquired on the DDR dataset was reported as 83.62%, 83.76%, 93.62%, 83.59% for the weighted accuracy, precision, recall and f-measure respectively. The hybrid classification framework can improve the performance of DR classification but still there is room for improvement which is addressed in the next framework proposed.

5.2.3 DR Image Classification using Divide and Conquer Approach

The third methodology presented in this work deals with research question 3 and research objective 3. The goal is to further improve the five stage DR classification by utilising the high-resolution image features from the fundus images without affecting the memory consumption. The contributions made in this methodology are listed below:

- i. In the first step of the framework, a modified CNN model is trained using a resized set of high-resolution input fundus images. The CNN model has an additional fully connected layer that is used to extract the local features from the image patches in the later stage.

- ii. For feature extraction, the input high-resolution images are divided into small patches. Transfer learning is applied to each of the patches using the CNN trained from the previous step. The features acquired from each patch of the image are concatenated to create a feature vector for the high-resolution image.
- iii. In the last stage of the proposed methodology, the feature vectors for the images are passed on to the ML classifiers for training and testing the DR classification framework.

The proposed mechanism provides maximum performance when SVM classifier is used to train and test the extracted features. In binary classification, maximum performance for DDR dataset is observed with values of 97.60%, 97.72%, 97.60%, and 97.75% for weighted accuracy, precision, recall, and f-measure respectively. Similarly, for multiclass three stage classification, our divide and conquer approach provides a maximum performance metric value of 89.46%, 89.57%, 89.46%, 89.39% for weighted accuracy, precision, recall, and f-measure respectively. Lastly, the divide and conquer approach also provides best performance for five stage DR classification with weighted average of 85.64%, 85.79%, 85.64%, and 85.58% for accuracy, precision, recall, and f-measure respectively. Our proposed frameworks provide valuable contribution in performing multiclass DR classification from fundus images of the eye.

5.3 Limitations

Our proposed algorithms provide a good performance improvement for multiclass DR classification of fundus images. However, the feature extraction methodologies and classifications are still limited by the amount of training data available for the fundus images of the eye. Most of the five stage DR classification datasets available online suffer from major imbalance issues which limit the amount of performance gains that can be attained by

our proposed frameworks. The custom dataset presented in this work for five stage multiclass classification can provide bias to the results when compared to other research works. However, the results are also tested on APTOS dataset to reduce this bias. In the divide and conquer approach, the main limitation lies in the extra time taken to extract features from the image patches compared to regular full size image feature extraction. This can increase the time taken to extract features and ultimately affect the performance of the system.

5.4 Future Work

DR is still one of the major causes of blindness in diabetic patients. The field of automatic classification of DR from fundus images is still growing rapidly throughout the world. It was observed in our frameworks that combine feature vectors from different ML methods and extracting local features from fundus images provided performance improvements in the DR classification system. The global features of the image also play vital role in classification of fundus images. In future the global features of the fundus images can also be combined along with the high-resolution local features acquired using divide and conquer approach to build DR detection systems that can provide better multiclass classification performance. Another work that can be carried out in the future is the creation of a large balanced dataset for DR images by collecting patients' information from various hospitals.

5.5 Summary

In this chapter, the contributions of the thesis, limitations, and future work is discussed. Our work primarily provided contributions in enhancing pre-processing stage of fundus images and extracting useful image features for DR classification. The work also

provides a memory efficient approach of extracting high-resolution features from input fundus images without the need for excessive GPU memory. Researchers in the field of fundus image classification for DR will find the work presented in this thesis of great interest.

REFERENCES

- Abdillah, B., Bustamam, A., & Sarwinda, D. (2017). Classification of diabetic retinopathy through texture features analysis. *2017 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 333–338. <https://doi.org/10.1109/ICACSIS.2017.8355055>
- Adriman, R., Muchtar, K., & Maulina, N. (2021). Performance evaluation of binary classification of diabetic retinopathy through deep learning techniques using texture feature. *Procedia Computer Science*, 179, 88–94. <https://doi.org/10.1016/j.procs.2020.12.012>
- Agarwal, S., & Bhat, A. (2023). A survey on recent developments in diabetic retinopathy detection through integration of deep learning. *Multimedia Tools and Applications*, 82(11), 17321–17351. <https://doi.org/10.1007/s11042-022-13837-5>
- Alyoubi, W. L., Abulkhair, M. F., & Shalash, W. M. (2021). Diabetic retinopathy fundus image classification and lesions localization system using deep learning. *Sensors*, 21(11). <https://doi.org/10.3390/s21113704>
- Baker, N., Lu, H., Erlikhman, G., & Kellman, P. J. (2020). Local features and global shape information in object classification by deep convolutional neural networks. *Vision Research*, 172, 46–61. <https://doi.org/10.1016/J.VISRES.2020.04.003>
- Banzi, W., Kambutse, I., Dusabejambo, V., Rutaganda, E., Minani, F., Niyobuhungiro, J., Mpinganzima, L., & Ntaganda, J. M. (2021). Mathematical modelling of glucose-insulin system and test of abnormalities of type 2 diabetic patients. *International*

- Journal of Mathematics and Mathematical Sciences*, 2021.
<https://doi.org/10.1155/2021/6660177>
- Ben-Nun, T., & Hoefler, T. (2019). Demystifying parallel and distributed deep learning. *ACM Computing Surveys (CSUR)*, 52(4). <https://doi.org/10.1145/3320060>
- Bhulakshmi, D., & Rajput, D. S. (2024). A systematic review on diabetic retinopathy detection and classification based on deep learning techniques using fundus images. *PeerJ Computer Science*, 10, e1947. <https://doi.org/10.7717/peerj-cs.1947>
- Bodapati, J. D., Shaik, N. S., & Naralasetti, V. (2021). Composite deep neural network with gated-attention mechanism for diabetic retinopathy severity classification. *Journal of Ambient Intelligence and Humanized Computing*, 12(10), 9825–9839. <https://doi.org/10.1007/s12652-020-02727-z>
- Butt, M. M., Iskandar, D. N. F. 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. <https://doi.org/10.3390/diagnostics12071607>
- Butt, M. M., Latif, G., Iskandar, D. N. F. A., Alghazo, J., & Khan, A. H. (2019). Multi-channel convolutions neural network based diabetic retinopathy detection from fundus images. *Procedia Computer Science*, 163, 283–291. <https://doi.org/10.1016/j.procs.2019.12.110>
- C, P., & R, J. K. (2023). Retinal image enhancement based on color dominance of image. *Scientific Reports*, 13(1), 7172. <https://doi.org/10.1038/s41598-023-34212-w>
- Canon. (2018). <https://eu.medical.canon/products/eye-care/cr-2-af>

- Carrera, E. V, González, A., & Carrera, R. (2017). Automated detection of diabetic retinopathy using SVM. *2017 IEEE XXIV International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, 1–4. <https://doi.org/10.1109/INTERCON.2017.8079692>
- Chatziralli, I. P., Kanonidou, E. D., Keryttopoulos, P., Dimitriadis, P., & Papazisis, L. E. (2012). The value of fundoscopy in general practice. *The Open Ophthalmology Journal*, 6, 4–5. <https://doi.org/10.2174/1874364101206010004>
- Chetoui, M., & Akhloufi, M. A. (2020). Explainable diabetic retinopathy using efficientnet. *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 1966–1969. <https://doi.org/10.1109/EMBC44109.2020.9175664>
- Chetoui, M., Akhloufi, M. A., & Kardouchi, M. (2018a). Diabetic retinopathy detection using machine learning and texture features. *2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE)*, 1–4.
- Chetoui, M., Akhloufi, M. A., & Kardouchi, M. (2018b). Diabetic retinopathy detection using machine learning and texture features. *2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE)*, 1–4. <https://doi.org/10.1109/CCECE.2018.8447809>
- Chowdhury, M. M. H., & Meem, N. T. A. (2020). A machine learning approach to detect diabetic retinopathy using convolutional neural network. In *Proceedings of International Joint Conference on Computational Intelligence: IJCCI 2018* (pp. 255-264). Springer Singapore. https://doi.org/10.1007/978-981-13-7564-4_23

- Dai, L., Wu, L., Li, H., Cai, C., Wu, Q., Kong, H., Liu, R., Wang, X., Hou, X., Liu, Y., Long, X., Wen, Y., Lu, L., Shen, Y., Chen, Y., Shen, D., Yang, X., Zou, H., Sheng, B., & Jia, W. (2021). A deep learning system for detecting diabetic retinopathy across the disease spectrum. *Nature Communications*, 12(1), 3242. <https://doi.org/10.1038/s41467-021-23458-5>
- Daniel, E., & Anitha, J. (2015). Optimum green plane masking for the contrast enhancement of retinal images using enhanced genetic algorithm. *Optik*, 126(18), 1726–1730. <https://doi.org/10.1016/j.ijleo.2015.05.027>
- Deepak, N. A., Savitha, G., Deepak, D. J., & Supraj, P. K. (2021, September). Retinopathy Based Multistage Classification of Diabeties. In *3rd International Conference on Integrated Intelligent Computing Communication & Security (ICIIC 2021)* (pp. 54-60). Atlantis Press. <https://doi.org/10.2991/ahis.k.210913.008>
- Dekhil, O., Naglah, A., Shaban, M., Ghazal, M., Taher, F., & Elbaz, A. (2019). Deep learning based method for computer aided diagnosis of diabetic retinopathy. *2019 IEEE International Conference on Imaging Systems and Techniques (IST)*, 1–4. <https://doi.org/10.1109/IST48021.2019.9010333>
- Deng, J., Dong, W., Socher, R., Li, L.-J., Kai Li, & Li Fei-Fei. (2009). ImageNet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. <https://doi.org/10.1109/CVPR.2009.5206848>
- El Houby, E. M. F. (2021). Using transfer learning for diabetic retinopathy stage classification. *Applied Computing and Informatics*. <https://doi.org/10.1108/ACI-07-2021-0191>

- El-Ateif, S., & Idri, A. (2022). Single-modality and joint fusion deep learning for diabetic retinopathy diagnosis. *Scientific African*, 17, e01280. <https://doi.org/10.1016/j.sciaf.2022.e01280>
- Elwin, J. G. R., Mandala, J., Maram, B., & Kumar, R. R. (2022a). Ar-HGSO: autoregressive-henry gas sailfish optimization enabled deep learning model for diabetic retinopathy detection and severity level classification. *Biomedical Signal Processing and Control*, 77, 103712. <https://doi.org/10.1016/j.bspc.2022.103712>
- Elwin, J. G. R., Mandala, J., Maram, B., & Kumar, R. R. (2022b). Ar-HGSO: autoregressive-henry gas sailfish optimization enabled deep learning model for diabetic retinopathy detection and severity level classification. *Biomedical Signal Processing and Control*, 77. <https://doi.org/10.1016/j.bspc.2022.103712>
- Emma Dugas Jared, J. W. C. (2015). Diabetic retinopathy detection. Kaggle. <https://kaggle.com/competitions/diabetic-retinopathy-detection>
- Farag, M. M., Fouad, M., & Abdel-Hamid, A. T. (2022). Automatic severity classification of diabetic retinopathy based on densenet and convolutional block attention module. *IEEE Access*, 10, 38299–38308. <https://doi.org/10.1109/ACCESS.2022.3165193>
- Gangwar, A. K., & Ravi, V. (2021). Diabetic retinopathy detection using transfer learning and deep learning. In *Evolution in Computational Intelligence: Frontiers in Intelligent Computing: Theory and Applications (FICTA 2020), Volume 1* (pp. 679-689). Springer Singapore. https://doi.org/10.1007/978-981-15-5788-0_64
- Gangwar Akhilesh Kumar and Ravi, V. (2021). Diabetic retinopathy detection using transfer learning and deep learning. In S.-L. and S. S. C. and Z. Y.-D. Bhateja Vikrant and Peng (Ed.), *Evolution in Computational Intelligence* (pp. 679–689). Springer Singapore.

- Gardner, M. W., & Dorling, S. R. (1998). Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmospheric Environment*, 32(14–15), 2627–2636. [https://doi.org/10.1016/S1352-2310\(97\)00447-0](https://doi.org/10.1016/S1352-2310(97)00447-0)
- Ghamdi, A. H. Al. (2020). Clinical Predictors of Diabetic Retinopathy Progression; A Systematic Review. *Current Diabetes Reviews*, 16(3), 242–247. <https://doi.org/10.2174/1573399815666190215120435>
- Gilliland, M. G. F., & Folberg, R. (1992). Retinal hemorrhages: Replicating the clinician's view of the eye. *Forensic Science International*, 56(1), 77–80. [https://doi.org/https://doi.org/10.1016/0379-0738\(92\)90149-Q](https://doi.org/https://doi.org/10.1016/0379-0738(92)90149-Q)
- Gour, N., & Khanna, P. (2021a). Multi-class multi-label ophthalmological disease detection using transfer learning based convolutional neural network. *Biomedical Signal Processing and Control*, 66, 102329. <https://doi.org/https://doi.org/10.1016/j.bspc.2020.102329>
- Gour, N., & Khanna, P. (2021b). Multi-class multi-label ophthalmological disease detection using transfer learning based convolutional neural network. *Biomedical Signal Processing and Control*, 66, 102329. <https://doi.org/https://doi.org/10.1016/j.bspc.2020.102329>
- Gour, N., & Khanna, P. (2021c). Multi-class multi-label ophthalmological disease detection using transfer learning based convolutional neural network. *Biomedical Signal Processing and Control*, 66, 102329. <https://doi.org/10.1016/j.bspc.2020.102329>
- Gu, Z., Li, Y., Wang, Z., Kan, J., Shu, J., & Wang, Q. (2023). Classification of diabetic retinopathy severity in fundus images using the vision transformer and residual

- attention. *Computational Intelligence and Neuroscience*, 2023, 1–12.
<https://doi.org/10.1155/2023/1305583>
- Gupta, B., & Tiwari, M. (2019). Color retinal image enhancement using luminosity and quantile based contrast enhancement. *Multidimensional Systems and Signal Processing*, 30(4), 1829–1837. <https://doi.org/10.1007/s11045-019-00630-1>
- Haga, M., Kawasaki, R., & Yamashita, H. (2005). International clinical diabetic retinopathy severity scales. *Nihon Rinsho. Japanese Journal of Clinical Medicine*, 63 Suppl 6, 171–177.
- Haque, S. M., Latif, G., Hasan, R., Arifuzzaman, M., Shafin, S. S., & Rahman, Q. A. (2019). Scalable parallel svm on cloud clusters for large datasets classification. *2nd Smart Cities Symposium (SCS 2019)*, 1–5. <https://doi.org/10.1049/cp.2019.0186>
- Harun, N. H., Yusof, Y., Hassan, F., & Embong, Z. (2019). Classification of fundus images for diabetic retinopathy using artificial neural network. *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, 498–501. <https://doi.org/10.1109/JEEIT.2019.8717479>
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. <https://doi.org/10.1109/CVPR.2016.90>
- Hearst, M. A., Dumais, S. T., Osuna, E., Platt, J., & Scholkopf, B. (1998). Support vector machines. *IEEE Intelligent Systems and Their Applications*, 13(4), 18–28.
<https://doi.org/10.1109/5254.708428>
- Ikram, A., Imran, A., Li, J., Alzubaidi, A., Fahim, S., Yasin, A., & Fathi, H. (2024). A systematic review on fundus image-based diabetic retinopathy detection and grading:

- current status and future directions. *IEEE Access*, 12, 96273–96303.
<https://doi.org/10.1109/ACCESS.2024.3427394>
- Jawad, E. M., Hazim, H. J. M., & Daway, G. (2022). Retinal image enhancement by using adapted histogram equalization based on segmentation and lab color space. *Int. J. Intell. Eng. Syst*, 15(3), 614-622..
- John, G. H., & Langley, P. (2013). Estimating continuous distributions in bayesian classifiers. *CoRR*, abs/1302.4964. <http://arxiv.org/abs/1302.4964>
- Kar, S. S., & Maity, S. P. (2018a). Automatic detection of retinal lesions for screening of diabetic retinopathy. *IEEE Transactions on Biomedical Engineering*, 65(3), 608–618.
<https://doi.org/10.1109/TBME.2017.2707578>
- Kar, S. S., & Maity, S. P. (2018b). Automatic detection of retinal lesions for screening of diabetic retinopathy. *IEEE Transactions on Bio-Medical Engineering*, 65(3), 608–618.
<https://doi.org/10.1109/TBME.2017.2707578>
- Karthik Maggie, S. D. (2019). APTOS 2019 blindness detection. *Kaggle*.
<https://kaggle.com/competitions/aptos2019-blindness-detection>
- Kassani, S. H., Kassani, P. H., Khazaeinezhad, R., Wesolowski, M. J., Schneider, K. A., & Deters, R. (2019). Diabetic retinopathy classification using a modified xception architecture. *2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 1–6.
<https://doi.org/10.1109/ISSPIT47144.2019.9001846>
- Kobat, S. G., Baygin, N., Yusufoglu, E., Baygin, M., Barua, P. D., Dogan, S., Yaman, O., Celiker, U., Yildirim, H., Tan, R.-S., Tuncer, T., Islam, N., & Acharya, U. R. (2022). Automated diabetic retinopathy detection using horizontal and vertical patch division-

- based pre-trained densenet with digital fundus images. *Diagnostics*, 12(8), 1975.
<https://doi.org/10.3390/diagnostics12081975>
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In F. Pereira, C. J. Burges, L. Bottou, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems*. 25.
 Curran Associates, Inc.
https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf
- Kropp, M., Golubnitschaja, O., Mazurakova, A., Koklesova, L., Sargheini, N., Vo, T.-T. K. S., de Clerck, E., Polivka, J., Potuznik, P., Polivka, J., Stetkarova, I., Kubatka, P., & Thumann, G. (2023). Diabetic retinopathy as the leading cause of blindness and early predictor of cascading complications—risks and mitigation. *EPMA Journal*, 14(1), 21–42. <https://doi.org/10.1007/s13167-023-00314-8>
- Kuru, K., & Girgin, S. (2011). A bilinear interpolation based approach for optimizing hematoxylin and eosin stained microscopical images. In *Pattern Recognition in Bioinformatics: 6th IAPR International Conference, PRIB 2011, Delft, The Netherlands, November 2-4, 2011. Proceedings* 6 (pp. 168-178). Springer Berlin Heidelberg.
- Lam, C., Yi, D., Guo, M., & Lindsey, T. (2018). Automated detection of diabetic retinopathy using deep learning. *AMIA Joint Summits on Translational Science Proceedings. AMIA Joint Summits on Translational Science, 2017*, 147–155.
- Latif, G., Al Anezi, F. Y., Zikria, M., & Alghazo, J. (2020). EEG-ECG signals classification for arrhythmia detection using decision trees. *2020 Fourth International Conference on*

- Inventive Systems and Control (ICISC)*, 192–196.
<https://doi.org/10.1109/ICISC47916.2020.9171084>
- Lazuardi, R. N., Abiwinanda, N., Suryawan, T. H., Hanif, M., & Handayani, A. (2020). Automatic diabetic retinopathy classification with efficientnet. *2020 IEEE REGION 10 CONFERENCE (TENCON)*, 756–760.
<https://doi.org/10.1109/TENCON50793.2020.9293941>
- Li, T., Gao, Y., Wang, K., Guo, S., Liu, H., & Kang, H. (2019a). Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening. *Information Sciences*, 501, 511–522.
- Li, T., Gao, Y., Wang, K., Guo, S., Liu, H., & Kang, H. (2019b). Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening. *Information Sciences*, 501, 511–522. <https://doi.org/10.1016/j.ins.2019.06.011>
- Li, X., Pang, T., Xiong, B., Liu, W., Liang, P., & Wang, T. (2017). Convolutional neural networks based transfer learning for diabetic retinopathy fundus image classification. *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 1–11. <https://doi.org/10.1109/CISP-BMEI.2017.8301998>
- Lu, B., & Malan, D. J. (2023). *CS50's Introduction to Artificial Intelligence with Python*. <https://cs50.harvard.edu/ai/2023/notes/4/>
- Menaouer, B., Dermene, Z., El Houda Kebir, N., & Matta, N. (2022). Diabetic retinopathy classification using hybrid deep learning approach. *SN Computer Science*, 3(5), 357.
<https://doi.org/10.1007/s42979-022-01240-8>

- Mohanty, C., Mahapatra, S., Acharya, B., Kokkoras, F., Gerogiannis, V. C., Karamitsos, I., & Kanavos, A. (2023). Using deep learning architectures for detection and classification of diabetic retinopathy. *Sensors*, 23(12), 5726. <https://doi.org/10.3390/s23125726>
- Monteiro, F. C. (2023). Diabetic retinopathy grading using blended deep learning. *Procedia Computer Science*, 219, 1097–1104. <https://doi.org/10.1016/j.procs.2023.01.389>
- Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. *Proceedings of the 27th International Conference on International Conference on Machine Learning*, 807–814.
- Nneji, G. U., Cai, J., Deng, J., Monday, H. N., Hossin, M. A., & Nahar, S. (2022). Identification of diabetic retinopathy using weighted fusion deep learning based on dual-channel fundus scans. *Diagnostics*, 12(2), 540. <https://doi.org/10.3390/diagnostics12020540>
- Olivas, E. S., Guerrero, J. D. M., Martinez-Sober, M., Magdalena-Benedito, J. R., & Serrano, L. (Eds.). (2009). *Handbook of research on machine learning applications and trends: Algorithms, methods, and techniques: Algorithms, methods, and techniques*. IGI global.
- Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1), 62–66. <https://doi.org/10.1109/TSMC.1979.4310076>
- Padmanayana, & B.K, Dr. A. (2022). Binary classification of dr-diabetic retinopathy using cnn with fundus colour images. *Materials Today: Proceedings*, 58, 212–216. <https://doi.org/10.1016/j.matpr.2022.01.466>

- Poly, T. N., Islam, M. M., Yang, H. C., Nguyen, P.-A., Wu, C. C., & Li, Y.-C. J. (2019). Artificial intelligence in diabetic retinopathy: insights from a meta-analysis of deep learning. *Studies in Health Technology and Informatics*, 264, 1556–1557. <https://doi.org/10.3233/SHTI190532>
- Quelleg, G., Lamard, M., Josselin, P. M., Cazuguel, G., Cochener, Bé., & Roux, C. (2008). Optimal wavelet transform for the detection of microaneurysms in retina photographs. *IEEE Transactions on Medical Imaging*, 27(9), 1230–1241. <https://doi.org/10.1109/TMI.2008.920619>
- Rakhlin, A. (2017a). Diabetic retinopathy detection through integration of deep learning classification framework. *BioRxiv*, 225508.
- Rakhlin, A. (2017b). Diabetic retinopathy detection through integration of deep learning classification framework. *BioRxiv*, 225508.
- Ramasamy, L. K., Padinjappurathu, S. G., Kadry, S., & Damaševičius, R. (2021a). Detection of diabetic retinopathy using a fusion of textural and ridgelet features of retinal images and sequential minimal optimization classifier. *PeerJ Computer Science*, 7, e456. <https://doi.org/10.7717/peerj-cs.456>
- Ramasamy, L. K., Padinjappurathu, S. G., Kadry, S., & Damaševičius, R. (2021b). Detection of diabetic retinopathy using a fusion of textural and ridgelet features of retinal images and sequential minimal optimization classifier. *PeerJ Computer Science*, 7, e456. <https://doi.org/10.7717/peerj-cs.456>
- Rish, I. (2001, August). An empirical study of the naive Bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence* (Vol. 3, No. 22, pp. 41-46). <https://api.semanticscholar.org/CorpusID:61568722>

- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408. <https://doi.org/10.1037/h0042519>
- Saini, M., & Susan, S. (2022a). Diabetic retinopathy screening using deep learning for multi-class imbalanced datasets. *Computers in Biology and Medicine*, 149, 105989. <https://doi.org/10.1016/j.compbiomed.2022.105989>
- Saini, M., & Susan, S. (2022b). Diabetic retinopathy screening using deep learning for multi-class imbalanced datasets. *Computers in Biology and Medicine*, 149, 105989.
- Sanjana, S., Shadin, N. S., & Farzana, M. (2021). Automated diabetic retinopathy detection using transfer learning models. *2021 5th International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*, 1–6. <https://doi.org/10.1109/ICEEICT53905.2021.9667793>
- Saranya, P., & Umamaheswari, K. M. (2023). Detection of exudates from retinal images for non-proliferative diabetic retinopathy detection using deep learning model. *Multimedia Tools and Applications*, 83(17), 52253–52273. <https://doi.org/10.1007/s11042-023-17462-8>
- Sebastian, A., Elharrouss, O., Al-Maadeed, S., & Almaadeed, N. (2023). A survey on deep-learning-based diabetic retinopathy classification. *Diagnostics*, 13(3), 345. <https://doi.org/10.3390/diagnostics13030345>
- Shin, H.-C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., & Summers, R. M. (2016). Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE*

- Transactions on Medical Imaging*, 35(5), 1285–1298.
<https://doi.org/10.1109/TMI.2016.2528162>
- Simonyan, K. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*. <http://arxiv.org/abs/1409.1556>
- Singh, N., Kaur, L., & Singh, K. (2019). Histogram equalization techniques for enhancement of low radiance retinal images for early detection of diabetic retinopathy. *Engineering Science and Technology, an International Journal*, 22(3), 736–745.
<https://doi.org/10.1016/j.jestch.2019.01.014>
- Singh Sisodia, D., Nair, S., & Khobragade, P. (2017). Diabetic retinal fundus images: preprocessing and feature extraction for early detection of diabetic retinopathy. *Biomedical and Pharmacology Journal*, 10(02), 615–626.
<https://doi.org/10.13005/bpj/1148>
- Sreng, S., Maneerat, N., Hamamoto, K., & Panjaphongse, R. (2018). Automated diabetic retinopathy screening system using hybrid simulated annealing and ensemble bagging classifier. *Applied Sciences*, 8(7). <https://doi.org/10.3390/app8071198>
- Stehman, S. V. (1997). Selecting and interpreting measures of thematic classification accuracy. *Remote Sensing of Environment*, 62(1), 77–89.
[https://doi.org/10.1016/S0034-4257\(97\)00083-7](https://doi.org/10.1016/S0034-4257(97)00083-7)
- Suryawanshi, V., & Setpal, S. (2017). Guassian transformed GLCM features for classifying diabetic retinopathy. *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, 1108–1111.
<https://doi.org/10.1109/ICECDS.2017.8389612>

- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. E., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2014). Going deeper with convolutions. *CoRR*, *abs/1409.4842*. <http://arxiv.org/abs/1409.4842>
- Szegedy, C., Wei Liu, Yangqing Jia, Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–9. <https://doi.org/10.1109/CVPR.2015.7298594>
- Tan, M., & Le, Q. V. (2020). EfficientNet: Rethinking model scaling for convolutional neural networks. *arXiv preprint arXiv:1905.11946*, 1
- Tan, X., & Triggs, B. (2010). Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Transactions on Image Processing*, 19(6), 1635–1650. <https://doi.org/10.1109/TIP.2010.2042645>
- Thambawita, V., Strümke, I., Hicks, S. A., Halvorsen, P., Parasa, S., & Riegler, M. A. (2021). Impact of image resolution on deep learning performance in endoscopy image classification: An experimental study using a large dataset of endoscopic images. *Diagnostics*, 11(12), 2183. <https://doi.org/10.3390/diagnostics11122183>
- Tin Kam Ho. (n.d.). Random decision forests. *Proceedings of 3rd International Conference on Document Analysis and Recognition*, 278–282. <https://doi.org/10.1109/ICDAR.1995.598994>
- Tsiknakis, N., Theodoropoulos, D., Manikis, G., Ktistakis, E., Boutsora, O., Berto, A., Scarpa, F., Scarpa, A., Fotiadis, D. I., & Marias, K. (2021a). Deep learning for diabetic retinopathy detection and classification based on fundus images: A review. *Computers*

- in *Biology and Medicine*, 135, 104599.
<https://doi.org/10.1016/j.compbiomed.2021.104599>
- Tsiknakis, N., Theodoropoulos, D., Manikis, G., Ktistakis, E., Boutsora, O., Berto, A., Scarpa, F., Scarpa, A., Fotiadis, D. I., & Marias, K. (2021b). Deep learning for diabetic retinopathy detection and classification based on fundus images: A review. In *Computers in Biology and Medicine* (Vol. 135). Elsevier Ltd.
<https://doi.org/10.1016/j.compbiomed.2021.104599>
- Vives-Boix, V., & Ruiz-Fernández, D. (2021). Diabetic retinopathy detection through convolutional neural networks with synaptic metaplasticity. *Computer Methods and Programs in Biomedicine*, 206, 106094. <https://doi.org/10.1016/j.cmpb.2021.106094>
- Wajid, S. K., & Hussain, A. (2015). Local energy-based shape histogram feature extraction technique for breast cancer diagnosis. *Expert Systems with Applications*, 42(20), 6990–6999. <https://doi.org/https://doi.org/10.1016/j.eswa.2015.04.057>
- Wang, H., Sun, Y., Cao, Y., Ouyang, G., Wang, X., Wu, S., & Tian, M. (2022). Classification for diabetic retinopathy by using staged convolutional neural network. *2022 Asia Conference on Algorithms, Computing and Machine Learning (CACML)*, 228–233. <https://doi.org/10.1109/CACML55074.2022.00045>
- Wang, J., Li, Y.-J., & Yang, K.-F. (2021). Retinal fundus image enhancement with image decomposition and visual adaptation. *Computers in Biology and Medicine*, 128, 104116. <https://doi.org/10.1016/j.compbiomed.2020.104116>
- Wu, Z., Shi, G., Chen, Y., Shi, F., Chen, X., Coatrieux, G., Yang, J., Luo, L., & Li, S. (2020). Coarse-to-fine classification for diabetic retinopathy grading using convolutional

- neural network. *Artificial Intelligence in Medicine*, 108, 101936.
<https://doi.org/10.1016/j.artmed.2020.101936>
- Xu, W., Fu, Y.-L., & Zhu, D. (2023). ResNet and its application to medical image processing: Research progress and challenges. *Computer Methods and Programs in Biomedicine*, 240, 107660. <https://doi.org/10.1016/j.cmpb.2023.107660>
- Zago, G. T., Andreão, R. V., Dorizzi, B., & Salles, E. O. T. (2020). Diabetic retinopathy detection using red lesion localization and convolutional neural networks. *Computers in Biology and Medicine*, 116, 103537.
- Zhang, C., Lei, T., & Chen, P. (2022). Diabetic retinopathy grading by a source-free transfer learning approach. *Biomedical Signal Processing and Control*, 73, 103423. <https://doi.org/10.1016/j.bspc.2021.103423>
- 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.
<https://doi.org/10.1016/j.knosys.2019.03.016>
- Zhou, M., Jin, K., Wang, S., Ye, J., & Qian, D. (2018a). Color retinal image enhancement based on luminosity and contrast adjustment. *IEEE Transactions on Biomedical Engineering*, 65(3), 521–527. <https://doi.org/10.1109/TBME.2017.2700627>
- Zhou, M., Jin, K., Wang, S., Ye, J., & Qian, D. (2018b). Color retinal image enhancement based on luminosity and contrast adjustment. *IEEE Transactions on Biomedical Engineering*, 65(3), 521–527. <https://doi.org/10.1109/TBME.2017.2700627>

- Zimmet, P., Alberti, K. G., Magliano, D. J., & Bennett, P. H. (2016). Diabetes mellitus statistics on prevalence and mortality: Facts and fallacies. *Nature Reviews Endocrinology*, 12(10), 616–622. <https://doi.org/10.1038/nrendo.2016.105>
- Zuiderveld, K. (1994). Contrast limited adaptive histogram equalization. *Graphics Gems*, 474–485.

APPENDICES

Journal Publications

1. **Butt**, G., Iskandar, M. A., Khan, M. M., Latif, & Bashar, A. (2024). MEDCnet: A memory efficient approach for processing high-resolution fundus images for diabetic retinopathy classification using convolutional neural networks. *International Journal of Imaging Systems and Technology*, Submitted
2. **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.
3. **Butt**, M. M., Latif, G., Iskandar, D. A., Alghazo, J., & Khan, A. H. (2019). Multi-channel convolutions neural network based diabetic retinopathy detection from fundus images. *Procedia Computer Science*, 163, 283-291.