DIABETIC RETINOPATHY CLASSIFICATION USING MACHINE LEARNING AND DEEP LEARNING MODELS

PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the Degree of Master of Technology in Electronics & Communication Engineering with specialization in Communication Systems Engineering by the A P J Abdul Kalam Technological University

by

AFRA K

TKM20ECCS01

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

TKM COLLEGE OF ENGINEERING

KOLLAM, 691 005.

JULY 2022

DIABETIC RETINOPATHY CLASSIFICATION USING MACHINE LEARNING AND DEEP LEARNING MODELS

PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the Degree of Master of Technology in Electronics & Communication Engineering with specialization in Communication Systems Engineering by the A P J Abdul Kalam Technological University

by

AFRA K

TKM20ECCS01

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

TKM COLLEGE OF ENGINEERING

KOLLAM, 691 005.

JULY 2022

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

TKM COLLEGE OF ENGINEERING

KOLLAM, 691 005.

CERTIFICATE

Certified that this report titled ***"Diabetic Retinopathy Classification Using Machine Learning and Deep Learning Models"*** *is a bonafide record of the work done by* ***AFRA K.*** *(Reg. No. TKM20ECCS01) under my supervision, in partial fulfillment of the requirements for the award of the Degree of Master of Technology in Electronics & Communication Engineering with specialization in Communication Systems by the A P J Abdul Kalam Technological University.*

Guide
Prof. Ajitha S. S.
Assistant Professor
Dept. of ECE, TKMCE.

Coordinator
Dr. Nishanth N.
Associate Profssor
Dept. of ECE, TKMCE.

HoD
Prof. Abid Hussain
Head, Dept. of ECE.
TKMCE

Acknowledgement

I take this opportunity to convey my deep sense of gratitude to all those who have been kind enough to offer their advice and assistance when needed which has led to the successful completion of this project.

First of all, I thank God Almighty for all his blessing throughout this endeavour without which it would not have been possible.

It is my privilege and pleasure to express my profound sense of respect, gratitude and indebtedness to Dr. T. A. Shahul Hameed, Principal, TKM College of Engineering for guiding and providing facilities for the successful completion of my project.

I sincerely thank Prof. Abid Hussain Muhammed, Head of the Department, Department of Electronics and Communication Engineering, for his guidance, valuable support and constant encouragement given to me during this project.

I express my sincere thanks to my PG coordinator, Dr. Nishanth N., Associate Professor, Department of Electronics and Communication Engineering, for the support and encouragement during the course of this project.

I take this opportunity to express my sincere gratitude and profound thanks to my guide, Prof. Ajitha S. S., Assistant Professor, Department of Electronics and Communication Engineering, for her advice, guidance and reference materials provided during the course of the project.

AFRA K.

ABSTRACT

The majority of people worldwide have the risk of getting the eye disease called Diabetic Retinopathy (DR). This is a major problem that could impair vision for most of diabetic patients. High glucose levels in retinal blood vessels are the main reason for this. Color fundus images are used to diagnose DR from databases like IDRiD, KAGGLE, MESSIDOR and DIARETDB1 etc. The traditional method requires trained doctors to determine the presence of lesions in the image, that may be utilised to effectively detect the illness, making it a time-consuming process. The effective classification of the disease relies heavily on feature extraction. Due to the superior image grading efficiency of deep learning approach, computer diagnosis of DR has developed into a viable tool for rapid detection and evaluation of the severity of DR. In this work, different types of CNN architectures are used to extract the features. The CNN output features are used as input for different types of machine learning methods (Support Vector Machines, Decision Tree, Random Forest, Naive Bayes Classifier and K-Nearest Neighbour). Deep learning techniques like (ResNet-50, VGG-16, MobileNet, EfficientNet-B3, Inception-v3, and Self-designed models) are also implemented for classification. Here performed a performance comparison of different techniques using different datasets. The models are evaluated using various evaluation metrices such as accuracy, precision, recall and auc etc. The model having highest value for evaluation metrices is selected as the best model.

ABBREVIATIONS

AUC	Area Under Curve
CNN	Convolutional Neural Network
CWS	Cotton Wool Spots
DL	Deep Learning
DR	Diabetic Retinoparhy
EX	Hard Exudates
FN	False Negative
FP	False Positive
GPU	Graphic Processing Unit
HE	Hemorrhages
IRMA	Intra-retinal Micro Vascular Abnormalities
KNN	K- Nearest Neighbour
MA	Microaneurysm
ML	Machine Learning
NPDR	Non-Proliferative Diabetic Retinopathy
NV	Neovascularization
RCA	Reverse Cross Attention
SE	Soft Exudates
SFCN	Supervised Fuzzy Clustering Network
SPP	Spatial Pyramid Pooling
TFA	Two Fold Feature Augmentation
TN	True Negative
TP	True Positive

Contents

List of Figures

Chapter 1

Introduction

Human eye is a unique sense organ that can pick up visual images, which are subsequently sent to the brain. It is contained within the eye sockets of the skull and is secured there by muscles. External structures include cornea, conjuctiva, pupil, iris and sclera. Internal structures include aqueous humour, vitreous humour, lens, optic nerve and retina. The cornea, which protects the iris and pupil of our eye, is the transparent front portion of the eye. The primary purpose is to refract light with the lens. Conjuctiva is a stratified squamous epithelium that borders the sclera. It produces mucus and tears, which lubricate our eyes and keep them moist and clear. Pupil is the tiny opening in the middle of the iris. It enables light to pass through and concentrate on the retina. Iris is the coloured, pigmented area of the eye that is visible from the outside. The primary job of iris is to adjust the pupil's size to the type of light entering the eye. Sclera is the visible white area. It shields the internal organs and is composed of thick connective tissue.

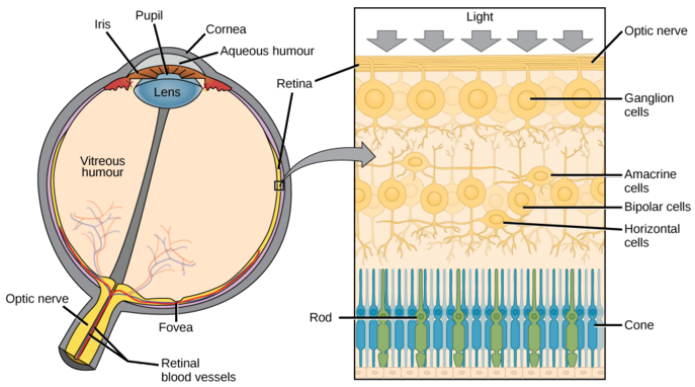


Figure 1.1: Illustration of human eye.

Fig. 1.1 shows the structure of the eye. A watery substance called aqueous humour is found between the lens and the cornea. It maintains the eye expanded and nourishes it. Vitreous humour is a clear jelly-like liquid found between retina and the lens. The translucent, biconvex lens of the eye is called the lens. Ligaments affix the lens to the ciliary body. Light is refracted by the lens and cornea to focus on the retina. Optic nerve is found in the back of the eyes. All impulses from the retina that are needed for perception are transported by the optic nerves to the brain. The deepest layer of the eye is called the retina. It functions as the film of a camera and is light sensitive. Ganglion, bipolar, and photoreceptor cells make up the three layers of these brain cells. It transforms a picture into electrical nerve impulses that the brain may use for visual experience.

Diabetic Retinopathy (DR) is one of the eye diseases which is caused by extreme glucose levels and high pressure in blood vessels. It adversely affects the retina or the rear of the eye and eventually leads to impaired vision. Most of the people around us experience vision threatening due to high blood glucose levels have a chance for having DR at lower levels. So diagnosing DR in the early stage will reduce risk. But accurately grading DR is a time-consuming and challenging task for ophthalmologists. So it is more important for creating a system for automated DR diagnosis. Diabetic retinopathy is classified in to different classes according to the severity. Grading depends on the size and the number of different lesions in images. There are five stages from zero to four which include Healthy (0), Mild (1), Moderate (2), Severe (3) and Proliferative (4) [1].

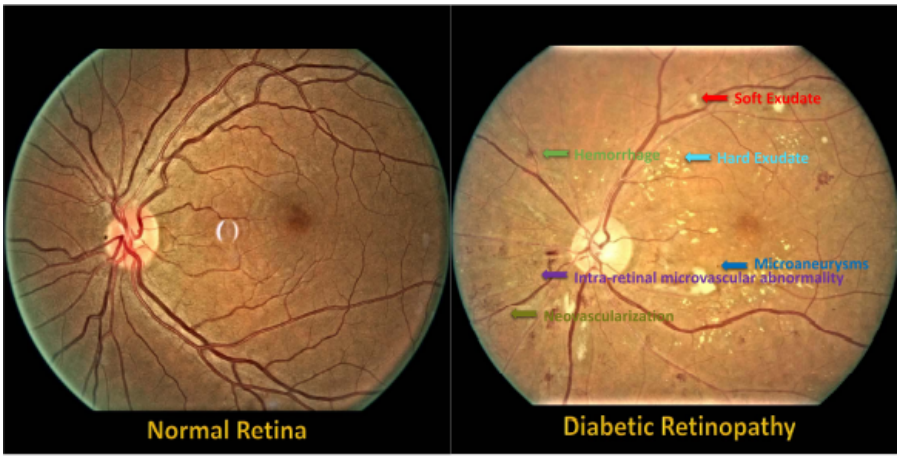


Figure 1.2: Illustration of DR retina.

Fig. 1.2 shows a picture of two retinas in which the left one is a healthy retina and the right one is affected with diabetic retinopathy having different types of lesions. There are different types of lesions which include Microaneurysms (MAs), Hemorrhages (HEs), Hard Exudates (EXs), Soft Exudates (SEs), Intra Retinal Microvascular Abnormalities (IRMAs) and Neovascularization (NVs) [2]. In the first stage of diabetic retinopathy, there will be the presence of microaneurysms (MAs). These appear as small red dots when local capillary dilatations occur. In the case of moderate Non Proliferative Diabetic Retinopathy (NPDR), there will be the presence of dot or blot-shaped hemorrhages along with microaneurysms. Hard exudates are the intra-retinal deposits that appear in yellow-white color and might range in size from tiny specks to big swaths. They are most frequently seen in macular area. Soft exudates are referred to as Cotton Wool Spots (CWS). They are pre-capillary arterial occlusions or white greyish patches of discolorations in the layer of nerve fibres. The severe DR stage often contains hard exudates or soft exudates. The regions of capillary dilatations and formation of new intra-retinal vessels are referred to as Tntra Retinal Microvascular Abnormalities (IRMAs).

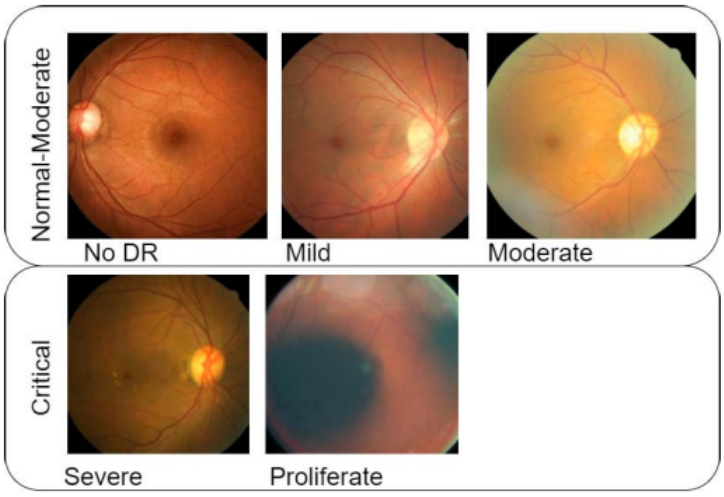


Figure 1.3: Illustration of DR stages.

Once IRMA is visible in sufficient numbers, a pre-proliferative DR stage can be predicted. Neovascularization (NV) is the major symptom of proliferative DR. New blood vessels may arise in the periphery of retina or from the optic disc as the retina becomes more ischaemic. Fig. 1.3 shows the different stages of diabetic retinopathy.

Limited amount of training data and inconsistent annotations are the two main road blocks to the development of the systems in computer-aided diagnostics for diabetic retinopathy. But kaggle dataset highly contribute to image classification task due to the large number of images than other existing dataset like IDRiD, MESSIDOR , DRIVE and DIARETDB1. Eventhough certain data augmentation techniques can be used to overcome the shortage of diabetic retinopathy training data include random distortion, zoom-in or zoom-out, horizontal flip and vertical flip etc. The medical image research community has been highly contributed by computer vision, machine learning-based algorithms and deep learning-based algorithms over the past decade. There are different types of machine learning classifiers including Logistic Regression, Naive Bayes Classifier, Nearest Neighbour, Support Vector Machines, Decision Tree, Boosted Trees, Random Forest and Neural Networks etc [3].

But in the case of image dataset of large size, deep learning plays an important role. There are many parameters associated with the deep learning techniques which can be adjusted to get the best result unlike machine learning models. They are named as hyper parameters and include activation functions, number of epochs, batch size, optimization algorithm and regularization etc. A lot of changes and developments occurred in Deep Convolutional Neural Networks (DCNN) [4]. Nowadays they have been implemented to perform many tasks such as object detection, image synthesis, semantic segmentation and image classification, etc. Some works focussed on diabetic retinopathy grading, which aims to achieve direct learning of local features and thereby classification. Some models used to train the DR images are basic CNN, ResNet-50, VGG-16, MobileNet, Inception-v3 and EfficentNet-B3 . Some of the models are also combined with various optimization algorithms such as Adaboost, Adagrad, SGD, and Adadelta, etc., which are used to optimize the neural networks to achieve fast results and better performance [5].

Chapter 2

Literature Review

Zubair *et al.* [6] suggested DR detection using VGG-NiN, a deep learning architecture. They outline a method for classifying different DR stages with the fewest learnable parameters possible to hasten model training and convergence. A highly non-linear, scale-invariant depth model known as the VGG-NiN model was produced by stacking the VGG16, the Spatial Pyramid Pooling layer (SPP), and the network in network (NiN) architecture. Because of the benefit of the SPP layer, the suggested model can process DR pictures at any size. Moreover, NiN stacking will increase the model's nonlinearity. RGB photos with a size of 224x244 are given as input to the VGG16.

The image is processed by blocks of three fully connected layers, followed by blocks of convolutional layers with 3x3 receptive field filters. A variety of input sizes can be handled by the convolutional layer. It makes use of kernel stack to push inputs and produce output feature maps. The SPP layer combines the features and generates an output vector of a defined size that complies with the needs of subsequent fully linked layers. The performance indicators of the suggested model were validated by the authors using an unbalanced version of the Kaggle dataset. The experimental findings demonstrate that, when compared to contemporary approaches, the suggested model is more accurate and efficient in its use of computer resources. The vast number of learnable parameters is a significant drawback.

Liu *et al.* [7] suggested a new diabetic retinopathy detection method relied on a advanced convolutional neural network. The efficiency of feature extraction is increased by the usage of symmetric convolution structures. The suggested strategy can also address the disproportion between negative and positive samples that leads to overfitting by enlarging the width and depth of the images. Several network topologies (pooling, convolution) are also utilised to accomplish various filters in the feature extraction step. It is primarily concerned with detecting Microaneurysms (MAs) and Hard exudates (HEs).

Experimental results show that the proposed method outperforms from the other techniques on a publicly available dataset DIARETDB1 (DB1). The object recognition accuracies are 93.6 %, 93.2% and 92.0%. each uses a different filter structure (convolution and pooling). The perception of microaneurysms is significantly enhanced by using average pooling layer for feature filtering. Detection of hard-exudates can be done by using the max-pooling layer. The disadvantage is that it only detects MAs and HEs of DR.

Luo *et al.* [8] proposed a retinal image classification by self-supervised fuzzy clustering network. Specifically, this paper proposes a Self-supervised Fuzzy Clustering Network (SFCN) with a feature learning module, reconstruction module and a fuzzy self-supervision module. Feature learning and reconstruction modules ensure network representativeness. The fuzzy self-monitoring module serves to further provide training directions for the entire network. In addition, three reconstruction losses, self-monitoring and fuzzy monitoring combine to optimize SFCN in an unsupervised way. The authors employ the proposed network on three widely used retinal image datasets to assess the effectiveness of the proposed method.

The results show sufficient performance for unsupervised retinal image classification tasks. Extensive validation experiments have been performed on the MESSIDOR, DRIVE, and DIARTDB1 datasets. Analysis of the results demonstrates the effectiveness of the SFCN approach. The proposed self-monitoring fuzzy clustering network achieves 87.6%, 81.7%, and 84.7% accuracy on the MESSIDOR, DRIVE and DIARETDB1 datasets respectively. Therefore, it also points out a major drawback of this network that it only possesses accuracy in the range of 80%.

Juan Wang *et al.* [9] suggested the concurrent diagnosis of features and severity of diabetic retinopathy in fundus photography using deep learning. In this paper, the authors proposed a multi-tasking hierarchical deep learning network for simultaneous diagnosis of features and severity in fundus images. A pyramid structure has been established to explain the accidental relationship between diabetic retinopathy related characteristics and the severity of DR. There were two different test sets in the experiment. The proposed approach was evaluated on both of them.

Additionally, a grading research was carried out to assess how well the suggested method performed in comparison to that of normal ophthalmologists with various degrees of competence. The outcomes demonstrate the potential of the suggested strategy to raise output compared to traditional deep-learning-based approaches. This provides performance close to that of a typical ophthalmologist of five year experience in the severity diagnosis of DR and a general ophthalmologist having ten year experience in detecting reference diabetic retinopathy. One of the main disadvantage of this network is its lower accuracy.

Lifeng Qiaq *et al.* [10] presented an early diagnosis approach for non-proliferative diabetic retinopathy based on deep learning algorithms for the detection of diabetic retinopathy. The proposed system uses a convolutional neural network algorithm that incorporates deep learning as a Graphics Processing Unit (GPU) accelerated core component to analyze the appearance of microaneurysm in fundus images, which will perform image detection and segmentation with high-performance and low-latency inference. Semantic segmentation algorithms are used to classify fundus images as infected and normal.

Semantic segmentation subdivides image pixels based on shared semantics to identify features of microaneurysms. This introduce an automated system that assists ophthalmologists in classifying fundus images as mild, moderate and severe non proliferative diabetic retiopathy. Prognosis of microaneurysm and early detection method for non proliferative diabetic retinopathy has been proposed to train convolutional networks effectively for fundus image semantic segmentation. This increases the accuracy and efficiency of non proliferative diabetic retinopathy detection. Therefore, this paper presents an unregulated classification approach based on sparse principal

component analysis to detect microaneurysms. Although it possesses good efficiency, it only detects the early stages and non-proliferative stages of DR. This is a disadvantage.

Sehrish Qummer *et al.* [11] suggested a deep learning ensemble approach for diabetic retinopathy detection. In this paper, authors used freely accessible Kaggle dataset of retinal images to train a group of five deep convolutional neural network models (Resnet-50, Inception-v3, Xception, Dense121 and Dense169) for rich functionality. The collective method is a meta-algorithm that combines multiple machine learning methods in a forecasting model. It is used for different tasks such as boosting, bagging and stacking. Stacking is a method employed for combining the details from several forcasting models to create a new one.

The stacked method often works better than individual models because of its lightweight nature. Stacking accentuates each base model's strongest points and exposes its weaknesses. Therefore stacking is the most efficient method when the base models differ significantly. So, in this work, the authors used stacking to enhance the prediction of the model. The authors utilized the biggest publicly accessible dataset of fundus images (Kaggle dataset) for training and testing of the model. The proposed ensemble model outperforms other modern methods and can also determine all stages of diabetic retinopathy. But the disadvantage is that it only classifies the DR's different stages by not disclosing the reason for a particular classification.

Renzhen Wang *et al.* [12] suggested a weakly supervised lesion detection from fundus images. In this article, a weak monitoring method requires a series of normal and abnormal retinal images without need to specify, annotate their locations and types, is proposed. In particular, fundus images are understood as a superposition of background blood vessels and background noise. The background is formulated as a low-rank structure after a series of simple pre-processing steps such as spatial alignment, color normalization and blood vessel removal. Therefore, it has the following characteristics. The normal image area tends to be displayed well, and the lesion area is diverse with significantly different in structure so that the lesion area can be appropriately suppressed. Background noise is considered as a stochastic variable and modeled through gaussian for normal images and a mixture of gaussian for abnormal

images respectively. The proposed method encodes both the background knowledge and background noise of the fundus image into its own model. It also optimizes the model using normal and abnormal images. The model is designed in three parts: background modeling, normal fundus image modeling and abnormal fundus image modeling. Then they form an overall model of the task under investigation rather than detecting specific types of lesions. This method focuses on detecting common types of lesions. Experimental results demonstrate that the proposed method is capable of finely detecting different types of lesions from abnormally colored fundus images and is clearly superior to other methods visually and quantitatively. But the disadvantage of this model is the lower accuracy.

Xianglong Zeng *et al.* [13] proposed automated diabetic retinopathy detection based on binocular siamese-Like convolutional neural network. In this article, the author trained a new convolutional neural network model with the Siamese architecture using transfer learning techniques. The suggested model takes fundus images captured with binoculars as input and grasp their correlations to make predictions.The suggested binocular model produces an area under the receiver motion curve of 0.951 with a training set of only 28,104 images and a test set of 7024 images, which is 0.011 greater than that produced by the current monocular model. The most important parts of the deep learning model to focus on, are datasets, network architectures and training methods. The fundus image dataset from the public source is enhanced and pre-processed before it is used to train the model.

In essence, the model transmits the left and right eye fundus images into the Siamese-like blocks serving as inputs. The fully linked layer gathers the data from the two images and then the model outputs the diagnostic findings for each fundus image. The main source of inspiration for this model's pipeline was the actual clinical diagnosis procedure for DR. The evaluation results show that the proposed binocular model achieved good performance with an AUC of 0.951, sensitivity of 82.2% on the high sensitivity operating point and a specificity of 70.7% on the high specificity operating point, which performs better than that of the existing model based on Inception V3 network. The binocular model is also trained for the original five classes of DR classification. It gains a kappa score higher than the existing model on the

10% validation data. In addition, the model is evaluated with various inputs and It demonstrates that the suggested binocular configuration is effective. In addition, the proposed binocular model can be modified and utilized for the automatic detection of other eye diseases without much effort. However, binocular models are difficult to train and test with different datasets where paired fundus images are not available.

Cam-Hao Hua *et al.* [14] suggested a convolutional network with two fold feature augmentation for diabetic retinopathy recognition from multi-modal images. The proposed architecture includes a backbone convolutional network associated with a Twofold Feature Augmentation mechanism, namely TFA-Net. The former contains several convolutional blocks that extract typical features at various points. The latter has a two-step structure that uses a convolutional kernel with shared weights and a Reverse cross attention (RCA) stream. If it uses a separate convolution layer for each input type, the amount of training data should be sufficient to make the acquired functionality more robust. On the other hand, using a weight distribution strategy at the beginning of the model allows the same kernel to selectively collect the clinical details of interest from the two modalities.

The learned parameters are then continually revised to effectively represent the essential cross-modal characteristics. The proposed model achieves a 90.2% quadratic weighted kappa rate on the small-sized internal KHUMC dataset. Despite achieving promising performance by detecting DR severity using two different input modalities, there are significant limitations: First, both benchmark datasets contain samples from a small population. The manual evaluation process can suffer from subjective bias. Second, there are several functional visualization tools available, but there are still challenges in interpreting DR-oriented characters in the feedforward process within the DL architecture. This then raises concerns for clinical ophthalmologists to track accurate risk factors that are exacerbated in the displayed images.

Sohini Roychowdhury *et al.* [15] suggested DREAM: diabetic retinopathy analysis using machine learning. In this article, the authors proposed a three-layer computer-aided screening system (DREAM) for DR that recognizes and ranks fundus images for the severity of NPDR. In this study, the authors identified a set of best 30 features out of 78 features using feature evaluation by AdaBoost, such that the 30 feature set classifies bright and red lesions with AUC greater than 0.83 using different classifiers. In addition, the time complexity analysis of the proposed two-level hierarchical classification method using all 78 features and a reduced set of 30 features, shows an increase in computational speed by more than 94% with this feature reduction operation. This time complexity analysis is an important contribution as there are no previous studies reporting the time complexity of DR acquisition systems. The last 30 features can classify lesions in less than 6 seconds per image and these 30 features are scalable for image datasets with different fields of view.

It is worth noting that the first step in the DREAM system is an important step for correctly proving the existence of DR. The optic disc (OD) and blood vessels were always correctly segmented in all images from the DIARETDB1 and MESSIDOR public datasets (including 1289 fundus images) analyzed in this article. However, some fundus images with additional retinal lesions such as myelination and peripapillary atrophy may not completely cover the entire OD region. This may allow the system to detect additional bright lesions. The proposed system is superior to all existing DR screening systems, but to further improve the specificity of DR classification, retrain the lesion classifier for each image in the new test set or lesion classification and use a cost-sensitive Support Vector Machines (SVM) or AdaBoost classifier or use other cost sensitive classifier combination algorithms. However, retraining a classifier or a cost sensitive classifier may require additional computing power and time complexity.

Chapter 3

Methodology and Proposed Architectures

This chapter mainly focuses on methodology which is common to all models and disusses about two different types of models. One of them is machine learning models and the other one is deep learning models for diabetic retinopathy classification [16]. There are a lot of architectures to perform these functions. But here have some particular classifiers are selected from machine learning techniques and some architectures from deep learning techniques for the interpretation and analysis of several models. Machine learning classifiers incorporate support vector machines, decision trees, random forest, naive bayes classifier and KNN classifiers. Deep learning architectures incorporate ResNet-50, VGG-16, Mobile Net, Efficient Net, Inception-v3 and self-designed models. The next section provides a brief explanation of the methodolgy and models .

3.1 Methodology

Methodolgy incudes data collection, data preparation, model training and evaluation. Data collection is by downloading the dataset which is already annotated by expert ophthalmologists. It can be huge or small dataset. Diabetic retinopathy affected retinal images can be collect directly from hospitals. The next step is data prepara-

tion which includes data pre-processing and data augmentation techniques [17]. Data pre-processing includes size normalization, shape normalization, colour normalization etc. Besides these pre-processing techniques, the Contrast Limited Adaptive Histogram Equalization (CLAHE) method is also employed in this work. Various types of data augmentation techniques can be used to mitigate the shortage of data include flipping of image horizontally or vertically, random zoom in or zoom out of images and random distortion of images. The last step is the model training and evaluation. Fig.

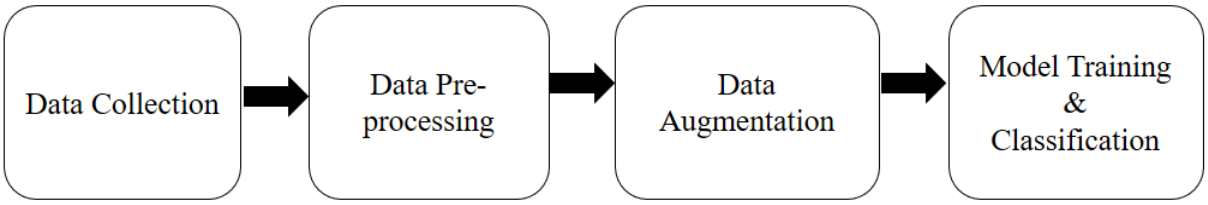


Figure 3.1: General workflow of proposed architectures.

3.1 shows the general work flow of proposed architectures. Input images are given to the model and the features of the images are extracted by the layers and then learn them effectively. The layers include one input layer, output layer and several hidden layers. Number of hidden layers can vary according to various models. Generally hidden layers involves convolutional layer, pooling layer, batch normalization and dense layers etc. After the output layer, there will be a classification layer which takes the no of classes as input and gives the output classes of the images [18].

3.2 Deep Learning Architectures

Deep learning architectures are widely employed for diabetic retinopathy classification purpose. This architectures takes the input images and learns the features without any human support unlike machine learning techniques. The motivation to employ deep learning is its flexibility in changing various parameters to improve accuracy and occurrence of feature extraction by itself. But it requires powerful hardware and very long training time according to huge dataset. Even though, deep learning architectures provide better performance than machine learning classifiers.

3.2.1 ResNet-50

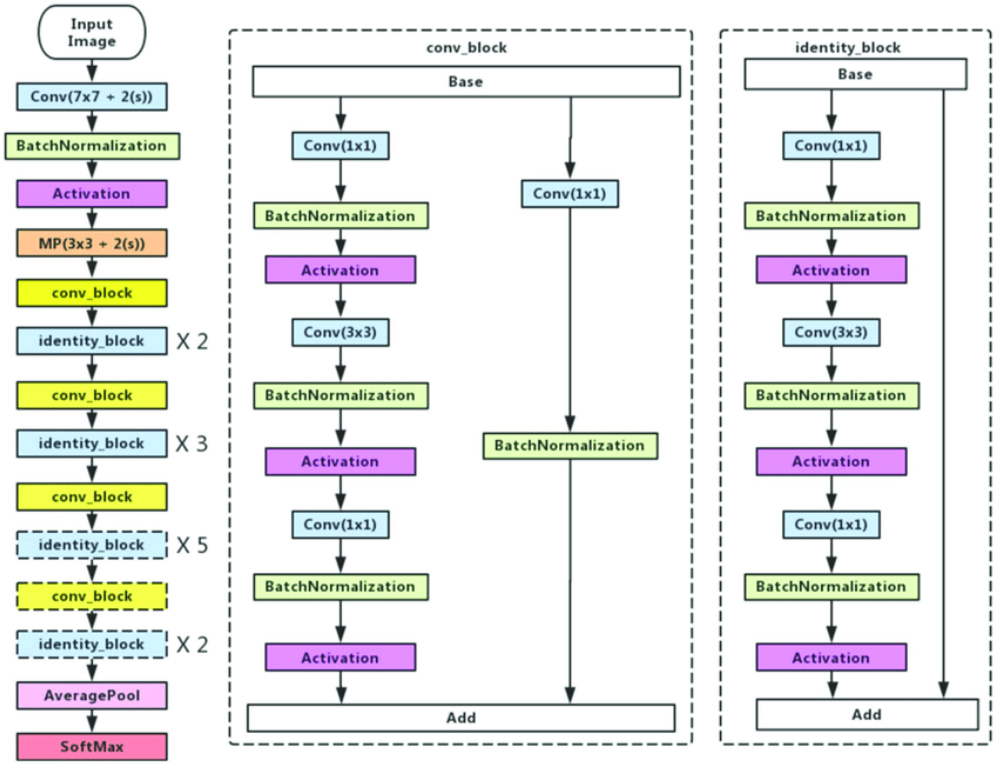


Figure 3.2: Resnet-50 architecture.

ResNet-50 is one of the major convolutional neural network. Fig. 3.2 shows the architecture. It is a 50 layer deep architecture. This is a common neural network that serves as the foundation for many computer vision applications. The main innovation with this architecture was that, it enabled us to train very deep neural networks with more than 150 layers. The vanishing gradient problem is a serious drawback for convolutional neural networks. Gradient value greatly reduces during backpropagation, therefore weights scarcely change at all. It is employed to get around this. It employs a skip connection. A direct link known as a skip connection bypasses some model layers.

Identity blocks and convolutional blocks are the two types of blocks used in this architecture [19]. One layer is produced using a convolution with a kernel size of 7x7 and 64 distinct kernel, each having a stride size of 2, which is followed by a max-pooling layer with a stride size of 2. There are three levels in the following convolution: a 1x1, 64 kernel, a 3x3, 64 kernel and finally a 1x1, 256 kernel, which are all repeated three times, giving us a total of nine layers in this phase. Kernel of 1x1,

128 is the next to be seen, followed by kernel of 3x3, 128 and finally Kernel 1x1,512. This phase was done four times, resulting in a total of 12 layers. Then there is a 1x1, 256 kernel, 3x3, 256 and 1x1, 1024 kernels repeated 6 times for 18 layers. Then a 1x1, 512 kernel with 3x3, 512 and 1x1, 2048 repeated 3 times for 9 layers. Then there is an average pool, finish it with a completely linked layer made up of 1000 nodes and add a softmax function to produce one layer.

3.2.2 VGG-16

Figure 3.3: VGG-16 architecture.

It is a convolutional neural network which has 21 layers out of which 16 layers have weights. Fig. 3.3 shows the architecture. Image passes via 2 convolution layers with 3x3 receptive size and then ReLU activations. Each layer has 64 filters. Convolution stride and padding are 1 pixel. The spatial resolution is maintained in this setup and the output activation map's dimensions match those of the input image. The activation maps are then run via spatial max pooling with a 2-pixel stride. This halves activations. The end-of-first-stack activations are 112x112x64. In a second stack comparable to the first, activations are sent through 128 filters instead of 64 in the first [20]. As a result, after the second stack, the final dimension is 56 x 56 x 128. The third stack has three convolutional layers and a maximum pool layer. The stack's output size is 28x28x256 filters. Two stacks of three convolutional layers with 512 filters follow the network. Both stacks will output 7x7x512. Three fully linked layers and a flattening layer follow the convolutional layers. The first two layers have 4,096 neurons apiece, while the output layer has 5 neurons for the 5 classes. After the output layer, there is a categorical classification softmax activation function.

3.2.3 Inception-V3

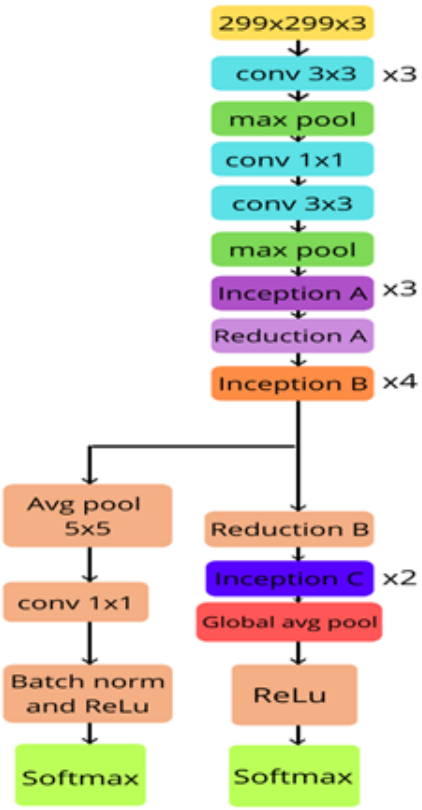


Figure 3.4: Inception-V3 architecture.

It is a convolutional neural network architecture that uses label smoothing, factorized 7x7 convolutions and an auxiliary classifer to transport label information down the network. Fig. 3.4 shows the architecture. It consists of 42 layers. There are three modules in this architecture. Which are inception - A, inception - B and inception - C [21]. The well-designed convolution modules used in this can both produce distinguishing characteristics and minimise the parameters. Each inception module has convolution and pooling layers. Inception modules employ 3x3, 1x3, 3x1 and 1x1 convolutional layers to minimise parameters. Three Inception A modules, five Inception B modules and two Inception C modules are stacked one on top of the other in Inception-v3. It has a default image size of 299x299. The feature map dimensions after the convolutional layers and Inception modules were 5x5 with 2,048 channels. Then the last softmax activation layer has a categorical classification.

3.2.4 MobileNet

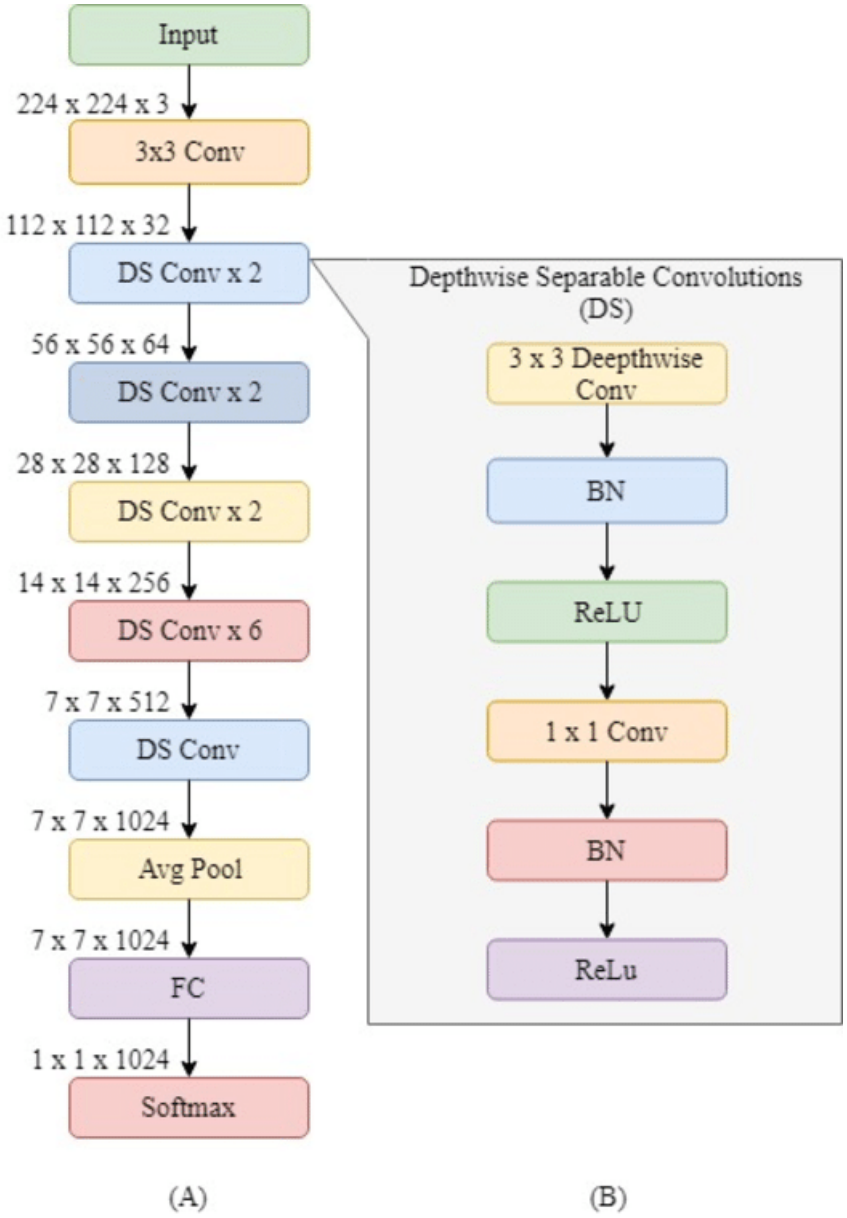


Figure 3.5: MobileNet architecture.

The architecture is small and low power model which consists of 28 layers. Fig. 3.5 shows the architecture. Depthwise separable convolutions are used by this model. The name separable comes from the idea that the depth and spatial dimension of a filter can be separated. When compared to the network with regular convolutions of the same depth in the nets, it significantly reduces the number of parameters [22]. As a result, lightweight deep neural networks are created. The primary distinction between this architecture and traditional CNN architecture is that instead of a single

3x3 convolution layer, it is followed by the batch normalization layer and ReLU layer [23]. This architecture divided the convolution into three 3x3 depth-wise convolutions and one 1x1 pointwise convolution. The width multiplier hyper parameter can have its value adjusted suitably in order to further minimise the standard architecture's parameter count, which is now 4.2 million. The dimensions of the image that is being read in are 224x224x3.

3.2.5 EfficientNet-B3

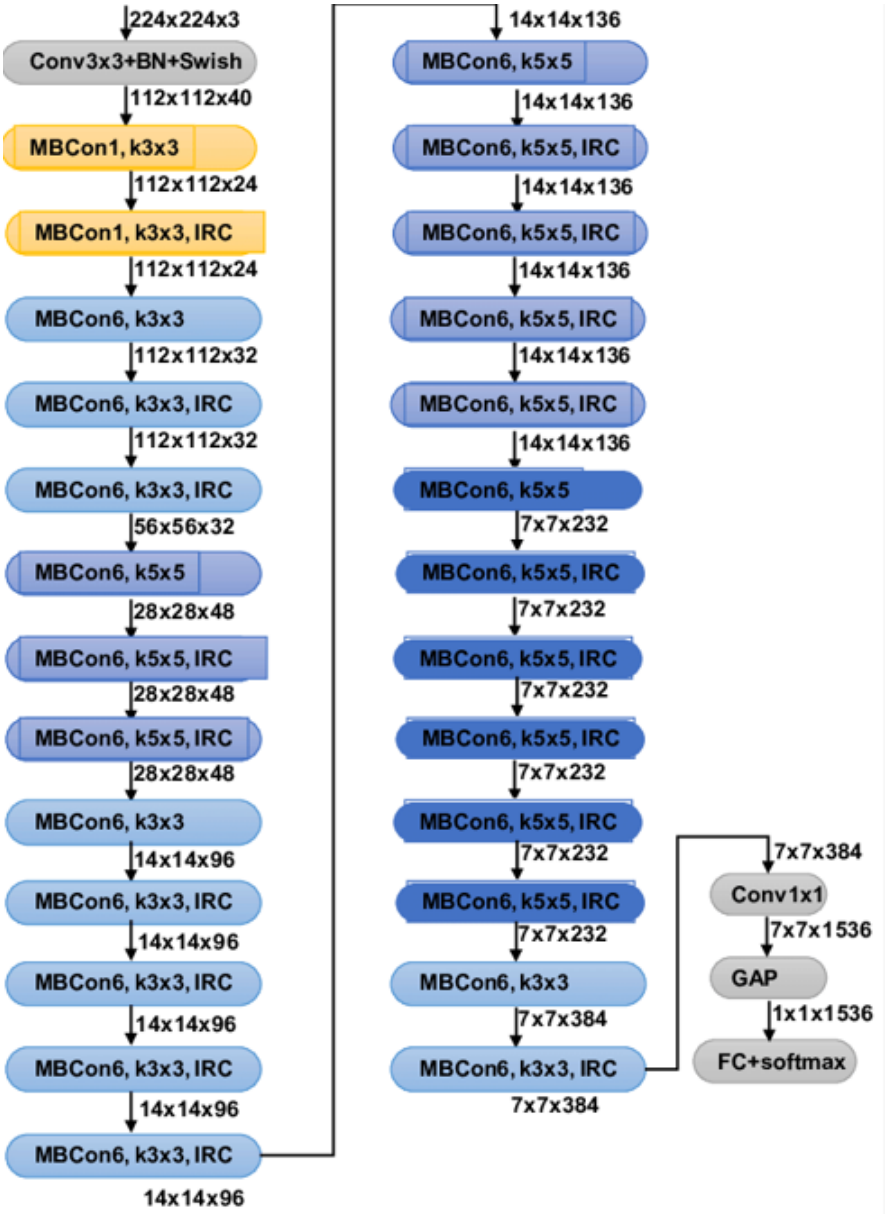


Figure 3.6: EfficientNet-B3 architecture.

Fig. 3.6 shows the architecture of the model. It involves re-evaluating model scaling for convolutional neural networks and the most powerful convolutional architecture existing among the others. It is a mobile size base line network. A multi-objective neural network architecture search that optimises both accuracy and FLOPS was used in the development of the baseline network. Its primary component is mobile inverted bottleneck MBConv, to which others contribute squeezing and excitation optimization [24]. Each individual MBConv block is a combination of the Conv2D and depthwise Conv2D algorithms, in addition to the squeeze and excitation operations. This structure is capable of being enlarged in a time and cost-effective manner while maintaining adherence to the compound coefficient's set of formulas.

$$\begin{aligned} d &= \alpha^{\theta} \\ \omega &= \beta^{\theta} \\ r &= \gamma^{\theta} \\ \alpha * \beta^2 * \gamma^2 &= 2 \\ \alpha \geq 1, \beta \geq 1, \gamma &\geq 1. \end{aligned} \tag{3.1}$$

θ is a coefficient which specifies how many resources are attainable . While α determines depth (d), β determines width (w) and finally γ determines resolution (r). Scaling up any dimension of the network improves accuracy, but the accuracy decreases for bigger models. So it is mandatory to balance all dimensions of network during scaling including depth, width and resolution. Resolution scaling enables to extract more features and fine-grained patterns. Depth scaling employing by increasing the number of layers to handle high resolution large dataset. Width scaling increases the number of channels or feature maps. So it will give maximum area of covering for a particular image. There are different types of efficientnets in use . In this work , B3 and B7 models are utilized. The number of layers varies from 237 to 813.

3.2.6 Self-designed Model

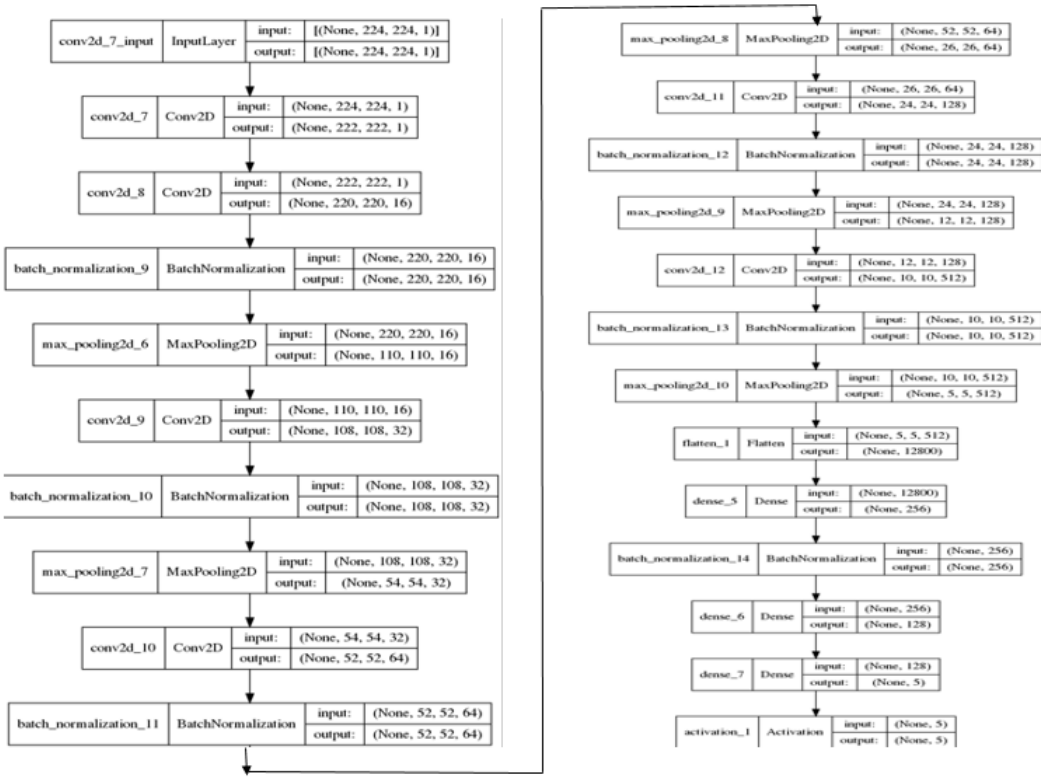


Figure 3.7: Architecture of Sequential-6 model.

Fig. 3.7 shows the architecture of self-designed model Sequential-6. It is a simple convolutional network of 23 layers, Which includes an input layer, six convolutional layers, six batch normalization layers, five max pooling layers, three dense layers, one flatten layer and the output layer. It takes the input images of size 224x224x1. Images were converted from rgb to grey scale. Convolutional layer extract the features, max pooling layer performs dimensionality reduction, batch normalization layer normalize the parameters to linear array, flatten layer flatten the array into a single column and dense layers again consists of repetition of above layers. It uses softmax activation function to classify images into five classes.

3.3 Machine Learning Classifiers

In machine learning, a classifier is an algorithm that automatically categorises data into different classes. Algorithms for machine learning are useful for automating manual operations. They can significantly save costs and increase productivity in businesses. Here are some machine learning classifiers and their explanations.

3.3.1 Support Vector Machines (SVM)

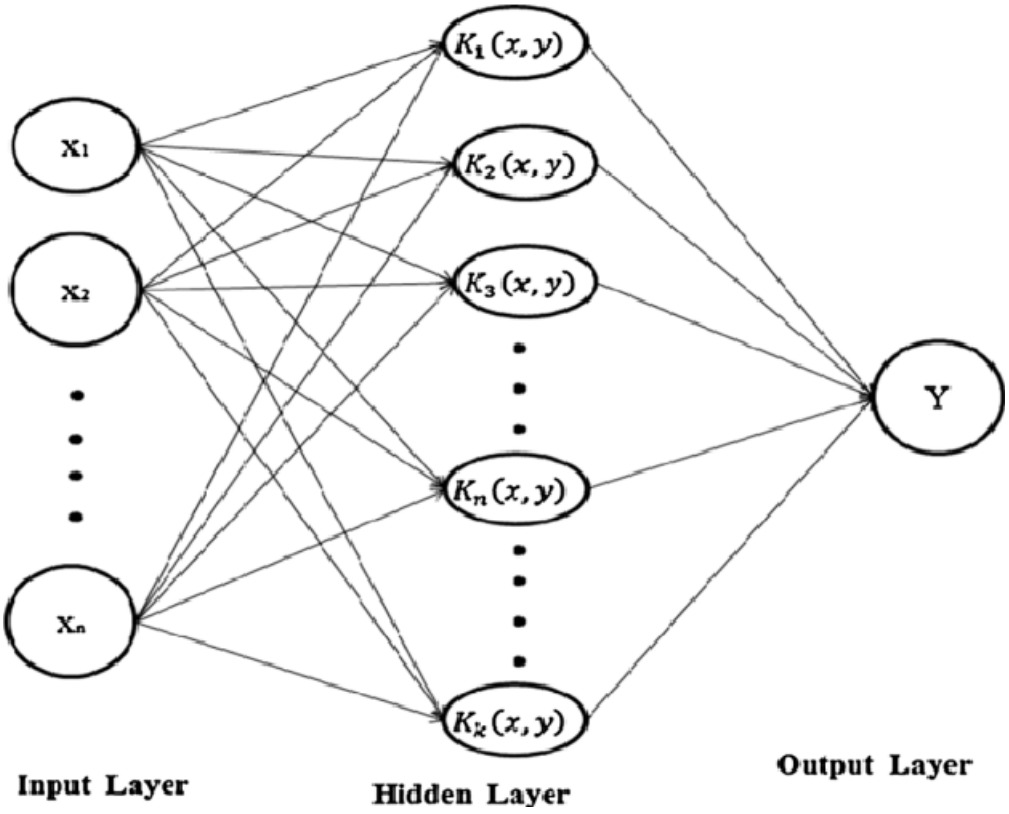


Figure 3.8: Architecture of SVM classifier.

Fig. 3.8 shows the architecture of SVM classifier. In the field of machine learning, support vector machines are supervised learning methods that have related learning algorithms [25]. It consists of an input layer, hidden layer and output layer. The input vectors are x_1, x_2...x_n and Y is the output. The SVM algorithm's objective is to establish the best line or decision boundary that can divide n-dimensional space into classes, allowing us to quickly classify fresh data points in the future [26]. A hyperplane is the name given to this optimal decision boundary. SVM selects the

extreme vectors and points that aid in the creation of the hyperplane. Support vectors, which are used to represent these extreme instances, form the basis for the SVM method. There are two types of SVM. They are linear and non-linear. Linear SVM is employed for data that can be separated linearly. Non-linear SVM is employed for data that is not linearly separable. If the vectors are not linearly separable in the space that they occupy, then the SVM can assist in making the vectors linearly separable in a space with more dimensions. In this study, the SVM is implemented using a RBF kernel with a gamma value chosen as the reciprocal of the product of the variance and the total number of features.

3.3.2 Random Forest

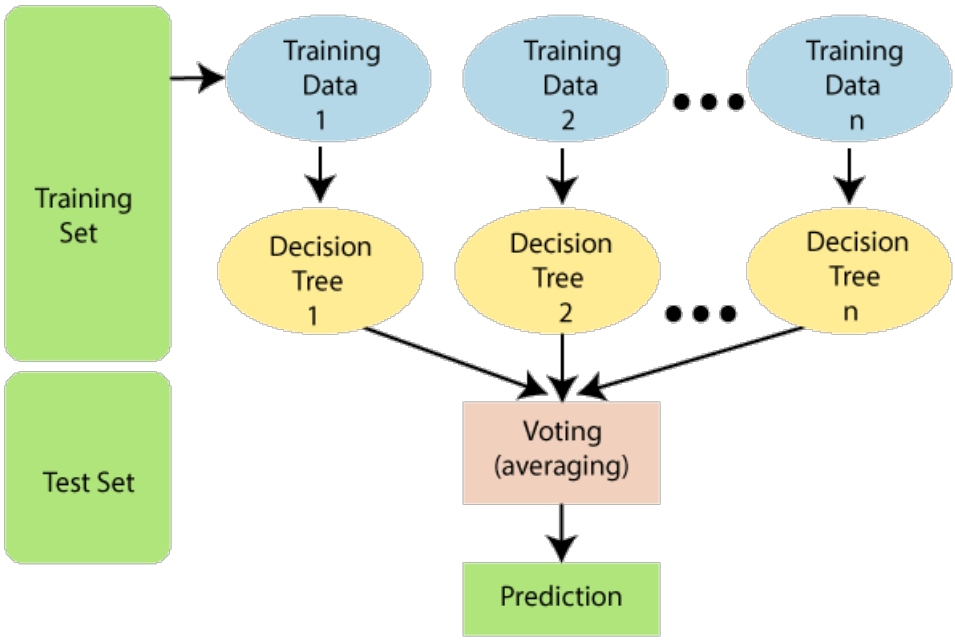


Figure 3.9: Architecture of Random forest classifier.

Fig. 3.9 shows the architecture of random forest classifier. A group of trees are developed together with each having a separate random vector in the Random Forest ensemble model classifier. i.e, the K^{th} tree generates a random vector AK that has the same distribution as earlier generated random vectors (A_1, A_2,..., $A_K - 1$). But it is independent from them [27]. There will be training test and test set to employ classification. Training set is further divided in to a number of small training data. They are given as input to different decision trees. The average of output of the

decision trees take and made the prediction. The best results come from a large number of very uncorrelated models (trees) working together as a committee. The uncorrelated models have the capacity to provide ensemble predictions that are more accurate than the individual forecasts. The random forest classifier is employed with hundred trees in this work.

3.3.3 Naive Bayes Classifier

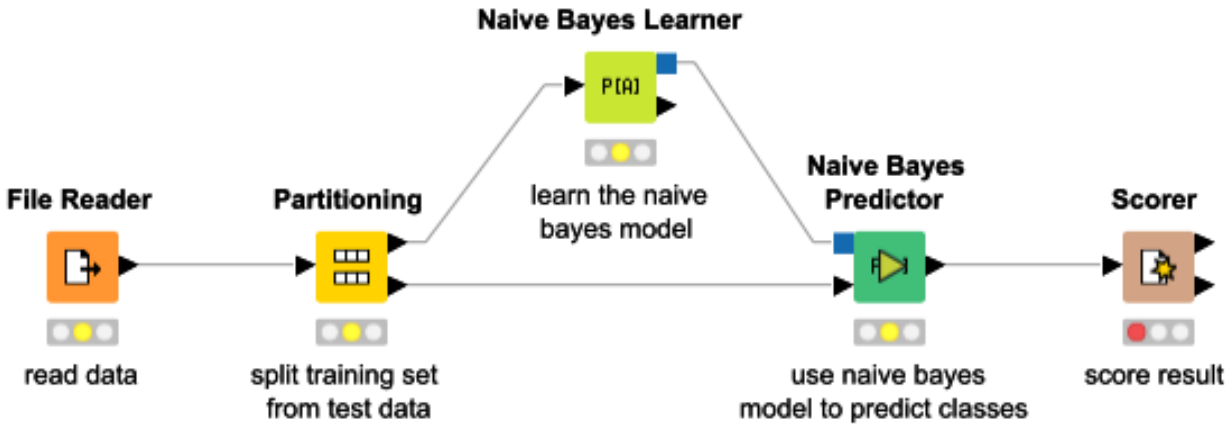


Figure 3.10: Naive bayes architecture.

Fig. 3.10 shows the architecture of naive bayes classifier. The file reader read the data and split in to training and testing set. Naive bayes model learn the data and predicts the classes. A score result is printed according to the predictions. It is a probabilistic classifier based on bayes theorem . The main advantage of this classifier is the requirement of small size dataset. If 'S' is the expected characteristics vector (S= s_1, s_2,....s_n) and 'r' is the target category variable with known pre-requisite probabilities $P(S)$ and $P(r)$ respectively [28]. Then treat $P(r/S)$ as likelihood and $P(S/r)$ as the later probability, It can be written in the form of

$$P(r \mid S) = \frac{P(S \mid r) \cdot P(r)}{P(S)} \tag{3.2}$$

The bayes theory is applied with solid (naive) hypothesis in the bayes method while in classification model, must determine the probability of a input dataset for all

opportunity values of category variable r. Then collect the outcome with highest likelihood.

3.3.4 Decision Tree

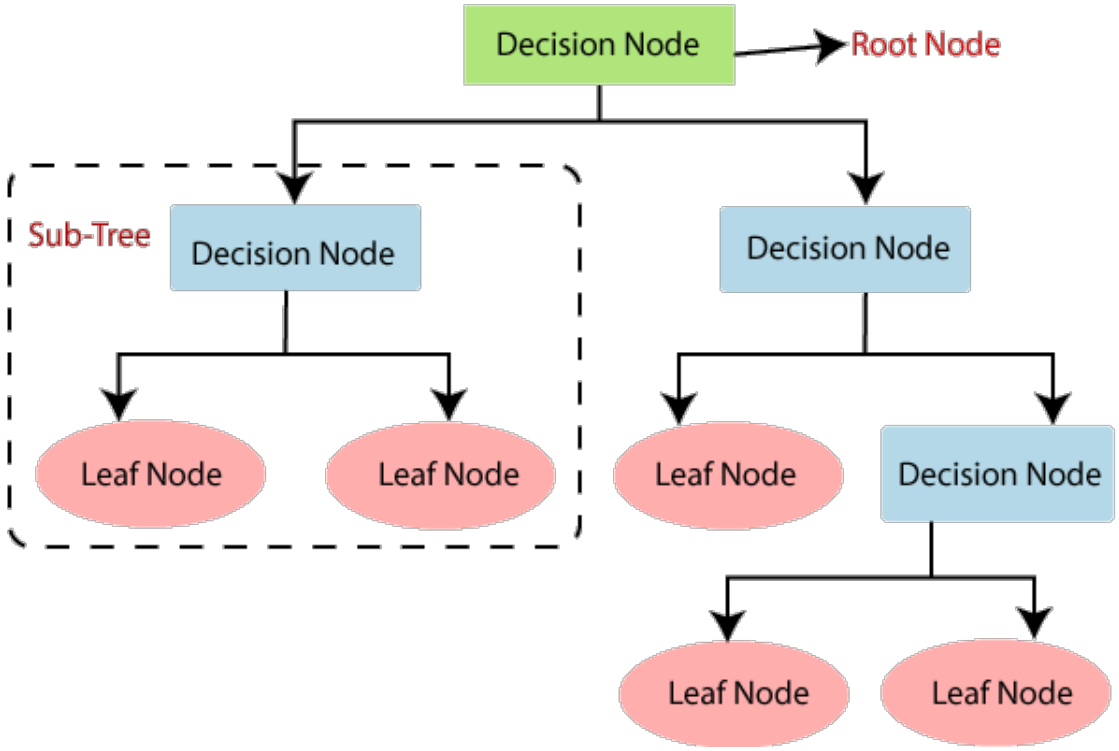


Figure 3.11: Naive bayes architecture.

Fig. 3.11 shows the architecture of decision tree. The objective of the decision tree is to learn straightforward decision rules inferred from the data features and then use them to build a model. An approximate piece wise constant can be thought of as a tree. It consists of decision nodes and leaf nodes. A decision tree include number of sub trees. The sub-trees are also begin with a decision node. The first decision node is termed as the root node. It is the most widely used and potent grading and forecasting tool. It is a pattern similar to the organizational chart In which , each inner node indicates a test of one attribute. Each limb represents one test result and each leaf node contains a class tag [29]. It categorizes the instances by assigning them to the root tree at certain leaf nodes. That provides the class of the instance.

3.3.5 K-Nearest Neighbour Classifier (KNN)

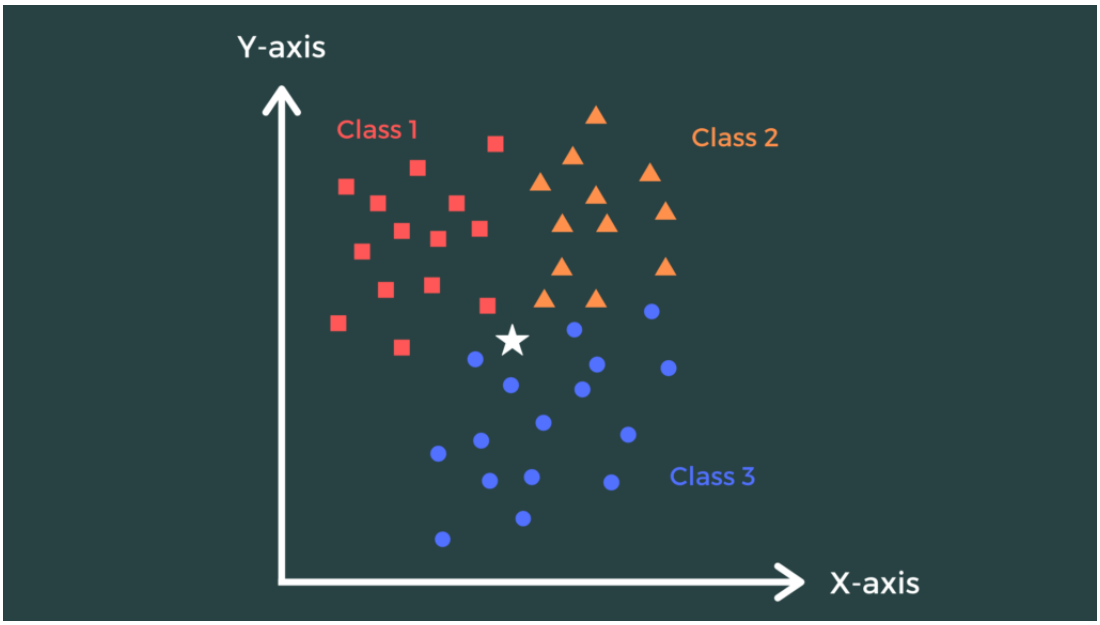


Figure 3.12: K-nearest neighbour distribution.

Fig. 3.12 shows the k-nearest neighbour distribution. The k-nearest neighbor algorithm sometimes referred to as KNN, is a supervised learning classifier that employs proximity to produce classification about the grouping of a single data point. Although it can be applied to classification or regression issues, it is commonly employed as a classification algorithm because it relies on the idea that comparable points can be discovered close to one another. It consists of number of points which is grouped in to different classes based on the similarity. The classes are class 1, class 2 and class 3. These algorithms are termed as the simplest and smartest because of requiring no learning phase. It is based on the computation of the distance between sampling points and the closest set of assigned neighbours of the points [30]. The decision shall be made by calculating the majority votes of the algorithm. To determine the closest neighbours, various types of distances can be used like triangular, cosine and gaussian etc.

Chapter 4

Datasets

There are a lot of dataset available nowadays for diabetic retinopathy classification. Most of the datasets were publicly available datasets which can be freely downloaded from the internet. There are dataset with huge number of images and also small number of images along with their labels. The size of dataset has high imapct on determining model accuracy. These datasets are optimized using various pr-eprocessing and then subjected to training and testing on different models. So from the view of dataset, recognized that the huge KAGGLE dataset gives maximum training and testing accuracy. Here are some dataset mentioned which are used for classification purpose. Fig. 4.1 shows the sample images from diabetic retinopathy dataset.

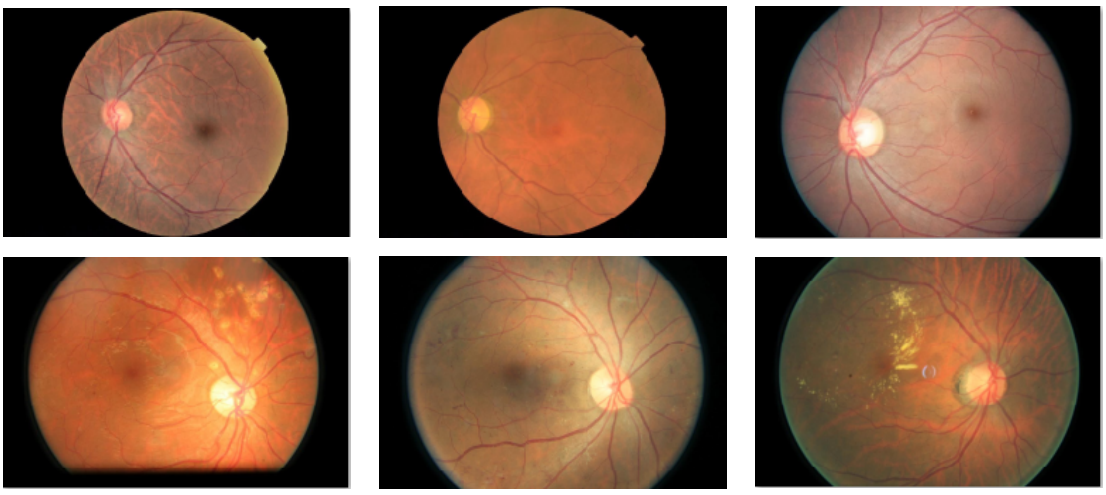


Figure 4.1: Sample images from diabetic retinopathy dataset.

4.1 Existing DR Classification Datasets

- ODIR-5K : This is a organised ophthalmic dataset of over 5000 patients. Image-level annotations with multiple labels for eye disease categories are provided, consists of glaucoma, diabetes, cataracts, age-related macular degeneration, hypertension, myopia, normality and other disorders [31]. Each patient may suffer from more diseases. This dataset will be used in the final task of transfer learning method for identifying multiple eye disorders.

- Kaggle-EyePACS : It consists of 35,126 training images and a test images containing only 53,576 rating labels [32]. Images are assembled from various sources with weak lettering quality and various lighting configurations. The appearance of DR in each image is evaluated on a scale of 0 to 4. Some of the images of this dataset are underexposed, overexposed or out of focus.

- MESSIDOR : It contains 1200 retinal images, but the DR rating scale differ from the previous records, with only 4 levels 0 to 3 [33] . In addition to the DR rating, the potential of macular edema is also given to each image with a rating label from 0 to 2.

- Kaggle-APTOSAUGMENTED : It is a balanced dataset containing 10000 images. So each classes like healthy (0), mild (1), moderate (2), severe (3) and proliferative (4) include 2000 images. In this dataset 8000 images are used for training and 2000 images are used for testing.

- Kaggle-APTOS2019 : It includes 3,662 training images and 1,928 test images, also with only a grading label. This dataset has noise in both labels and images [34].

- Kaggle-reduced-dataset : It consists of 2750 images , in which 2500 images are used for training and 250 images were used for testing.

- Kaggle-clean-dataset : It consists of total 2772 images in which 2500 image were used for training and 272 images were used for testing.

- IDRiD dataset : It consists of 413 training images and 103 testing images for the classification purpose [35].

Chapter 5

Results and Discussions

This chapter deals with the results obtained from various models used for training and testing. Five existing deep learning architectures mentioned in the previous chapter and various self-designed convolutional neural network models were used for the grading purpose. Here performed feature extraction using distinct deep learning models and the outputs are used as input for above mentioned machine learning classifiers. It is observed that the testing accuracy of deep learning architectures depends on various factors which include the learnable parameters (batch size, loss function, optimizers, regularizers, number of epochs, steps per epoch and activation functions etc.) that can be changed during training of the model. Also there is equal importance for the size of the dataset in the training and testing accuracies. In the case of machine learning classifiers, the accuracies only depends on datasets and models. There is no learnable parameters to adjust. There are several types of evaluation matrices which determines the best results out of all results obtained. This includes accuracy, precision, recall, f1-score, sensitivity and specificity etc. These are briefly explained below.

5.1 Evaluation Metrices

- **Accuracy** : Accuracy is the simplest and first observed metric in the model. This can be defined as the number of correctly identified test cases or images divided by the total number of test cases or images.

Accuracy = (TP + TN) / (TP + TN + FP + FN)

- TP = True Positive
- TN = True Negative
- FP = False Positive
- FN = False Negative

- **Precision** : The ratio of the number of positive samples that were correctly classified to all of the samples that were classified as positive is used to calculate precision (either correctly or incorrectly). In identifying a sample as positive, the model's precision reflects how accurately it did so. The denominator rises and the precision becomes low when the model makes many wrong Positive classifications or few correct Positive classifications. Precision will be maximum when the models make much correct positive categorization or make lesser incorrect positive categorization.

  Precision = (TP) / (TP + FP)

- **Recall** : It is determined as the proportion of positive samples that were correctly identified as positive to all positive samples. It assesses a model's capacity to identify Positive samples. More positive samples are found when recall is higher. The classification of the positive samples is all that matters for the recall. This is unrelated to the classification of the negative samples.

  Recall = (TP) / (TP + FN)

- **F1-score** : The accuracy of a model on a dataset is gauged by the F-score, also known as the F1-score. It is employed to assess binary categorization schemes that label examples as positive or negative. It is known as the harmonic mean of model's precision and recall, which is a technique to combine these two model attributes. It is frequently used to assess machine learning models for many different purposes, especially in natural language processing and information retrieval systems like search engines.

  F1- score = 2* precision * recall / (precision + recall)

- **Sensitivity** : It refers to the capacity of the model to identify positive instances. True positive rate is another name for this evaluation metric. It helps us to determine how many cases the model was able to accurately recognise, which is utilised to assess model performance. If a model has a high sensitivity, then it will have a low rate of false negative samples, which indicates that it will overlook some of the positive occurrences. The sum of true positive rate and false negative rate would be one. A high sensitivity indicates that the model is properly detecting the majority of the positive findings, whereas a low sensitivity indicates that the model is missing a significant number of the positive findings.

  Sensitivity = (TP) / (TP + FP)

- **Specificity** : It is defined as the fraction of true negative samples that are accurately detected by the model. This indicates that there will be another fraction of actual negative results that were projected as positive results and could be referred to as false positive samples. This can be also called true negative rate. The sum of true negative rate and false positive rate would be one. A specificity that is high indicates that the model is accurately identifying the majority of the negative findings, whereas a specificity that is low indicates that the model is incorrectly classifying a large number of negative samples as positive.

  Specificity = (TN) / (TN + FN)

5.2 Result 1 : Extraction of Features Using Deep Learning Architectures & Classification Using ML Classifiers.

Deep learning models are used to extract the features and machine learning classifiers are used for classification in the first set of experiments. The main motivation behind this idea was the lesser time taken by the machine learning classifiers. Also, machine learning program files can execute on a device having less processing power and lesser specifications. Here are the accuracies got from machine learning classifiers.

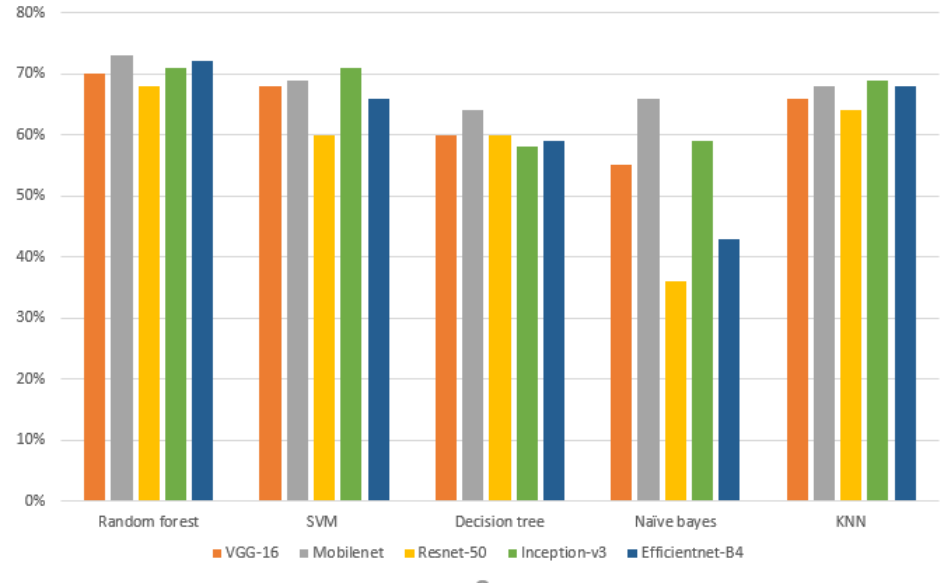
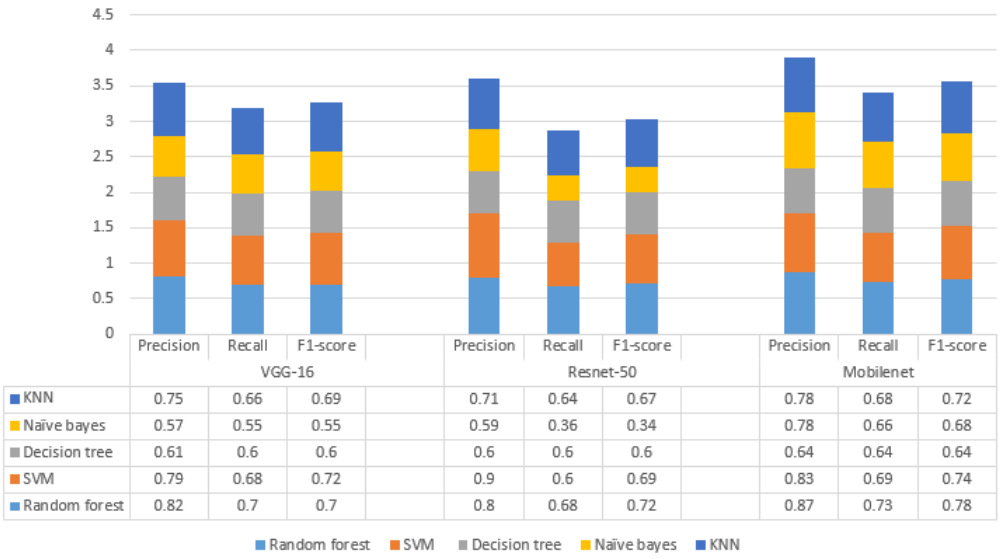
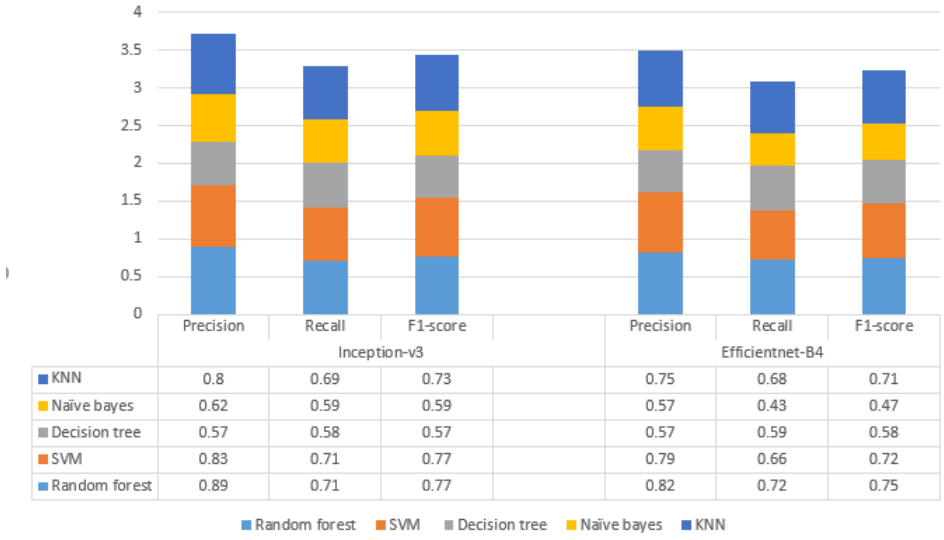


Figure 5.1: Accuracy comparison of machine learning classifiers.

From the Fig. 5.1 , understand that average accuracy of all classifiers is maximum when feature extractor is mobilenet architecture. So it is the best feature extractor among others and resnet-50 is the worst feature extractor among others. Random forest classifier gives maximum accuracy of 73% employing mobilenet architecture. From the results obtained, conclude that random forest classifier performs well out of other classifiers with overall average accuracy about 70.8%. Support vector machine classifier is showing the second better performance with an average accuracy of 67.5%. Also naïve bayes gives poor classification performance among other classifiers with overall average accuracy about 51.8% . Here are some other parameters like like precision, recall and f1-score which determines the classification performance.



	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
	VGG-16			Resnet-50			Mobilenet		
KNN	0.75	0.66	0.69	0.71	0.64	0.67	0.78	0.68	0.72
Naïve bayes	0.57	0.55	0.55	0.59	0.36	0.34	0.78	0.66	0.68
Decision tree	0.61	0.6	0.6	0.6	0.6	0.6	0.64	0.64	0.64
SVM	0.79	0.68	0.72	0.9	0.6	0.69	0.83	0.69	0.74
Random forest	0.82	0.7	0.7	0.8	0.68	0.72	0.87	0.73	0.78



	Precision	Recall	F1-score	Precision	Recall	F1-score
	Inception-v3			Efficientnet-B4		
KNN	0.8	0.69	0.73	0.75	0.68	0.71
Naïve bayes	0.62	0.59	0.59	0.57	0.43	0.47
Decision tree	0.57	0.58	0.57	0.57	0.59	0.58
SVM	0.83	0.71	0.77	0.79	0.66	0.72
Random forest	0.89	0.71	0.77	0.82	0.72	0.75

Figure 5.2: Performance comparison of ML classifiers on other parameters.

The above Fig. 5.2 shows performance comparison in terms of evaluation metrics other than accuracy. Random forest classifier possess highest overall average precision of 84% , highest overall average recall of 70.8% and highest overall average f1-score of 75.2% while the minimum precision is possessed by decision tree classifier about 59% , minimum recall and f1-sore is possessed by naïve bayes classifier about 51.8% and 52.6% respectively. The conclusion from the above results is that, random forest gives overall best classification performance in terms of evaluation metrics. But, here cannot change any parameters to increase the accuracy of the model.

5.3 Result 2 : Employing Deep Learning Architectures for Feature Extraction & Classification.

Deep learning architectures are widely used in classification purpose. This architectures takes the input images and learns the features without any human support unlike machine learning techniques. The motivation to employ deep learning is its flexibility in changing various parameters to improve accuracy and occurrence of feature extraction by itself. The number of epoch used was hundred. First, classification was performed in IDRiD dataset using various self-designed models and pre-trained models. But training accuracy was very small because of lesser number of images.

Then the number of images is increased by certain data augmentation techniques mentioned in chapter 3. Even after augmentation the final accuracy was very low. Then, used kaggle reduced dataset which contains almost 2750 imges to train and test the models. But, it also gave lower accuracy but much better than previous one. Then, used kaggle aptos augmented dataset which contains 10000 images and kaggle original dataset which includes 35126 images. Here are results obtained from various pre-trained models by employing different parameters.

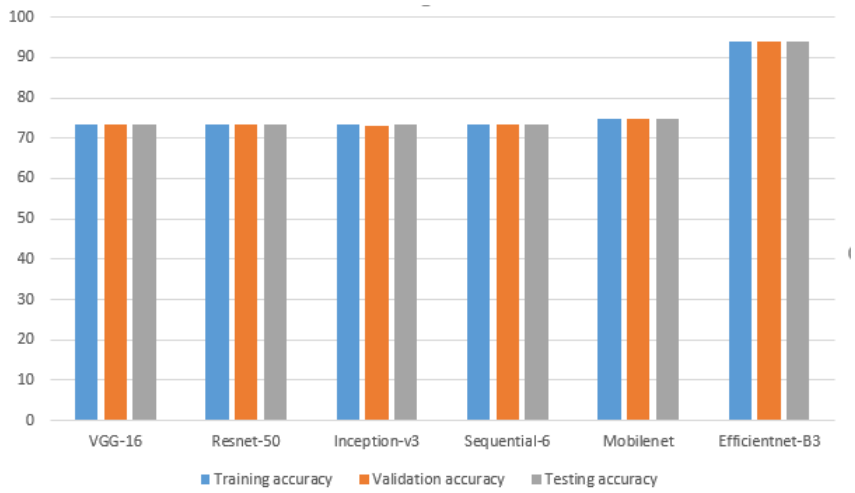
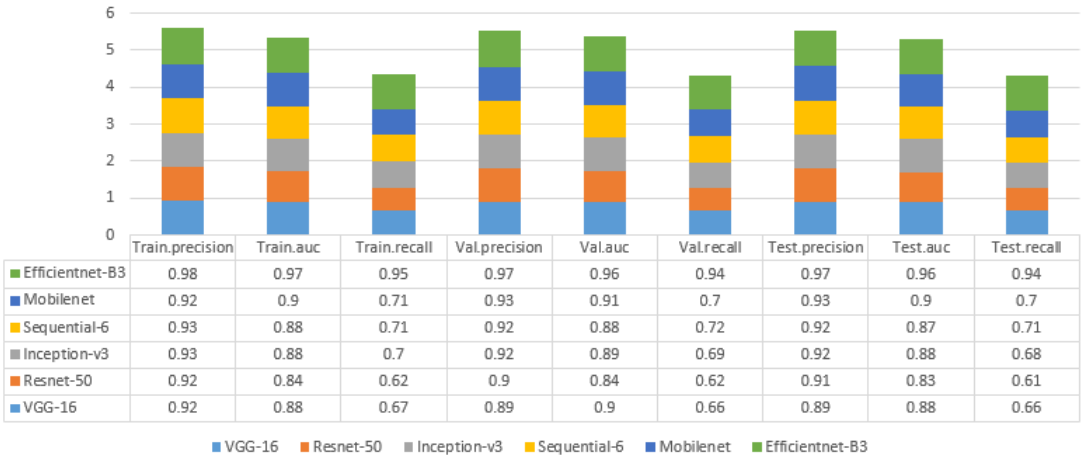


Figure 5.3: Accuracy comparison of DL architectures.

Fig. 5.3 shows the performance comparison of the various pre-trained models and self-designed model employed in terms of accuracy. The highest accuracy achieved by Efficientnet-B3 architecture. It has the training, testing and validation accuracies about 93%, 94% and 94% respectively. All other models employed possess approximate same value for training testing and validation accuracies about 73%.



	Train.precision	Train.auc	Train.recall	Val.precision	Val.auc	Val.recall	Test.precision	Test.auc	Test.recall
Efficientnet-B3	0.98	0.97	0.95	0.97	0.96	0.94	0.97	0.96	0.94
Mobilenet	0.92	0.9	0.71	0.93	0.91	0.7	0.93	0.9	0.7
Sequential-6	0.93	0.88	0.71	0.92	0.88	0.72	0.92	0.87	0.71
Inception-v3	0.93	0.88	0.7	0.92	0.89	0.69	0.92	0.88	0.68
Resnet-50	0.92	0.84	0.62	0.9	0.84	0.62	0.91	0.83	0.61
VGG-16	0.92	0.88	0.67	0.89	0.9	0.66	0.89	0.88	0.66

Figure 5.4: Performance comparison of DL architectures on other parameters.

Fig. 5.4 represents the performance comparison of the models in terms of other evaluation metrics including precision, recall and auc for training, validation and testing. Efficient-B3 has highest precision, auc and recall for training about 98%, 97% and 95% respectively. It possess highest precision, auc and recall for validation about 97%, 96% and 94% respectively. It also outperforms other models in testing according to precision, auc and recall. The values are about 97%, 96% and 94%. Sequential-6 is the self employed model.

The precision, recall and auc for training are about 93%, 88% and 71% respectively. These parameters for validation are 92%, 88% and 72% respectively and for testing are 92%, 87% and 71% . At the same time resnet-50 is under performed model out of other models. It possess lowest precision, auc and recall for training about 92%, 84% and 62% respectively. It possess lowest precision, auc and recall for validation about 90%, 84% and 62%. It also shows under performance for testing in terms of precision, auc and recall because of these evaluation metrics possess lowest value of about 91%, 83% and 61% respectively.

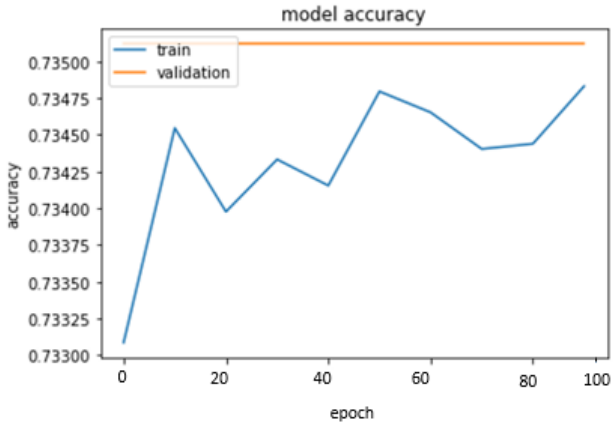


Figure 5.5: Accuracy vs epochs graph for Sequential-6.

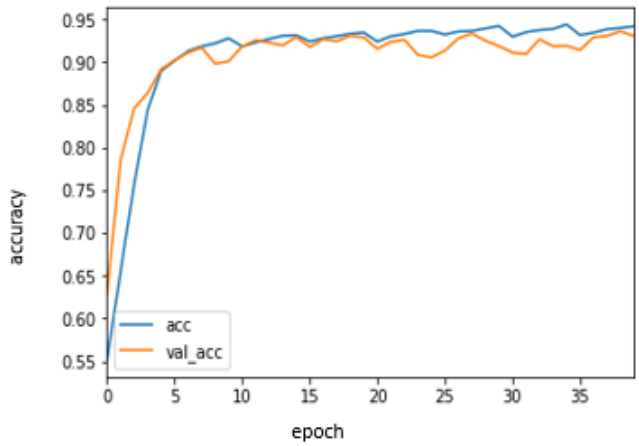


Figure 5.6: Accuracy vs epochs graph for Efficientnet-B3.

Fig. 5.5 shows the accuracy plot for sequential-6 model, in which x-axis shows the number of epochs and y-axis shows the accuracy of the model. There is some variations in accuracy eventhough accuracy increases as epochs increases. Sequential-6 is a self-designed architecture. Fig. 5.6 shows accuracy plot for Efficientnet-B3 architecture, in which x-axis is labeled with epochs and y-axis is labeled with accuracy. Accuracy directly increases as epoch increases without any variations.

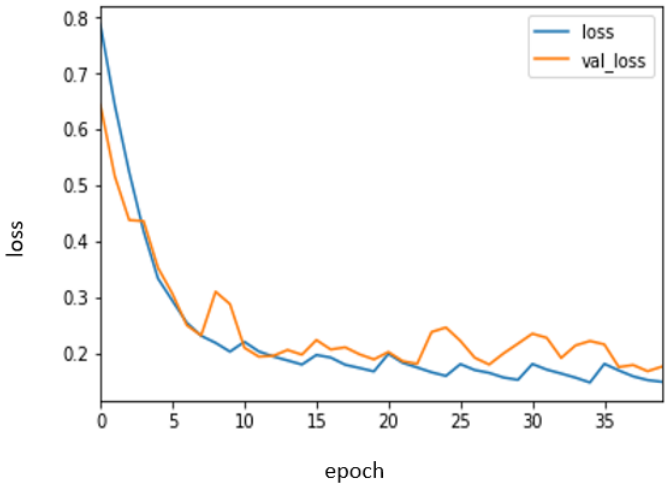


Figure 5.7: Loss vs epochs graph for Efficientnet-B3.

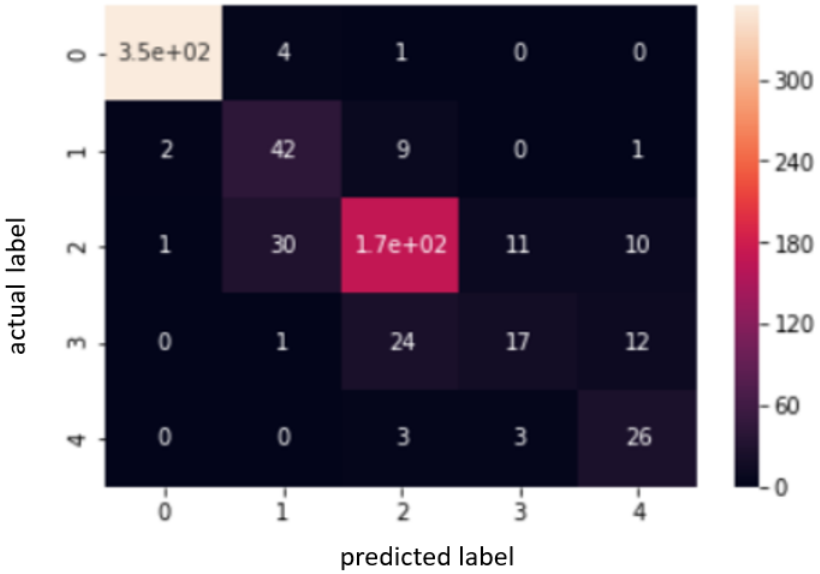


Figure 5.8: Confusion matrix for efficientnet-B3 architecture.

Fig. 5.7 shows the loss versus epochs graph for the same architecture. The loss is very small and it is decreasing as epochs are increasing. Fig. 5.8 shows the confusion matrix of the efficientnet architecture. The x-axis shows the predicted label and y-axis shows the actual label. The diagonal elements shows the correctly classified images. The values for evaluation metrices can be manually calculate once the confusion matrix is generated. Fig. 5.9 shows the sample images of correct prediction. Fig. 5.10 shows the sample images from incorrect predictions.

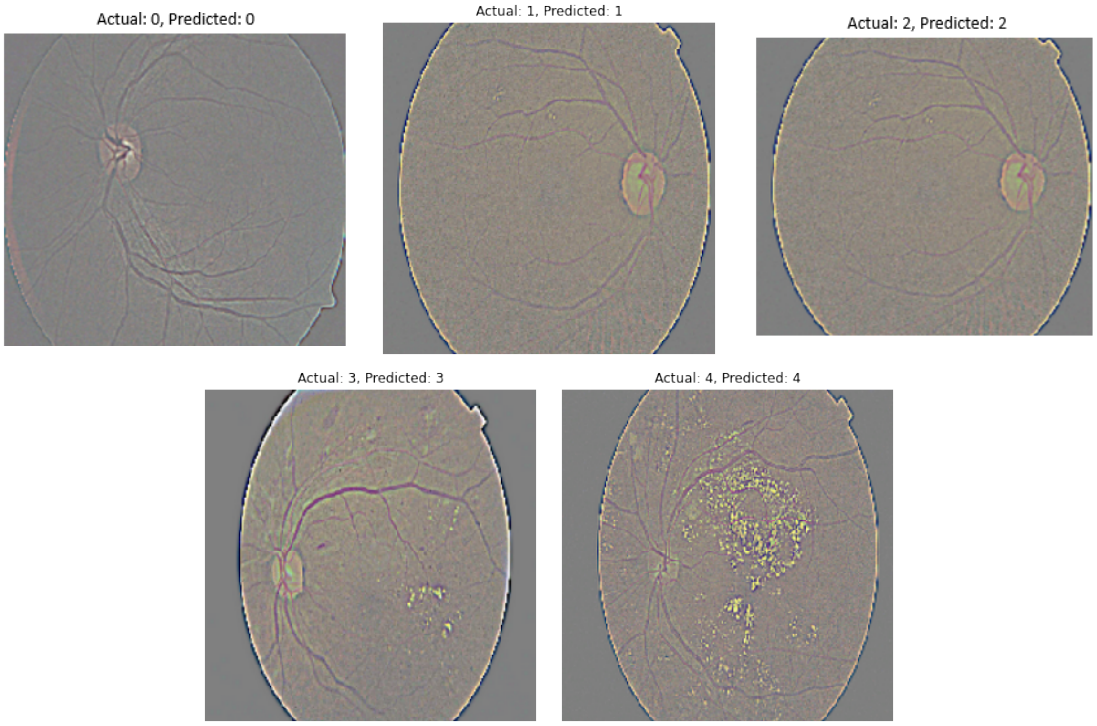


Figure 5.9: Sample images of correct prediction.

Figure 5.10: Sample images of incorrect prediction.

Chapter 6

Conclusion

The main objective behind this work was to find a model which gives better classification performance for diabetic retinopathy images. Machine learning classifiers and deep learning architectures were used here. In addition, conducted an extensive experiments to compare different architectures. Experimental results show the performance and efficiency of the models. The accuracies show that deep learning models are more accurate and flexible than machine learning models. Among the pretrained architectures, EfficientNet-B3 architecture gave highest performance in terms of evaluation metrics like accuracy, precison, recall and auc etc. It achieves 94% testing accuracy and the loss of the model was very less. Many self-designed models were implemented and tested on different datasets and got highest testing accuracy as 73.51% on kaggle diabetic retinopathy dataset. The visualization and analysis of the trained model provides information on how the model makes a diagnosis based on a given fundus image. In the future, data from more sources will be included and a wider pilot study will be launched. The collected data will be used to improve the accuracy of the model. Segmentation and classification together can be employed to get better results on the diabetic retinopathy grading task.

References

[1] G. G. Yen and W.-F. Leong, "A sorting system for hierarchical grading of diabetic fundus images: A preliminary study," *IEEE Transactions on Information Technology in Biomedicine*, vol. 12, no. 1, pp. 118–130, 2008.

[2] Y. Zhou, B. Wang, L. Huang, S. Cui, and L. Shao, "A benchmark for studying diabetic retinopathy: Segmentation, grading, and transferability," *IEEE Transactions on Medical Imaging*, vol. 40, no. 3, pp. 818–828, 2021.

[3] A. Guezzaz, Y. Asimi, M. Azrour, and A. Asimi, "Mathematical validation of proposed machine learning classifier for heterogeneous traffic and anomaly detection," *Big Data Mining and Analytics*, vol. 4, no. 1, pp. 18–24, 2021.

[4] J. Huang, S. Huang, Y. Zeng, H. Chen, S. Chang, and Y. Zhang, "Hierarchical digital modulation classification using cascaded convolutional neural network," *Journal of Communications and Information Networks*, vol. 6, no. 1, pp. 72–81, 2021.

[5] D. Gong, Z. Zhang, Q. Shi, A. van den Hengel, C. Shen, and Y. Zhang, "Learning deep gradient descent optimization for image deconvolution," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 12, pp. 5468–5482, 2020.

[6] Z. Khan, F. G. Khan, A. Khan, Z. U. Rehman, S. Shah, S. Qummar, F. Ali, and S. Pack, "Diabetic retinopathy detection using vgg-nin a deep learning architecture," *IEEE Access*, vol. 9, pp. 61 408–61 416, 2021.

[7] T. Liu, Y. Chen, H. Shen, R. Zhou, M. Zhang, T. Liu, and J. Liu, "A novel diabetic retinopathy detection approach based on deep symmetric convolutional neural network," *IEEE Access*, vol. 9, pp. 160 552–160 558, 2021.

[8] Y. Luo, J. Pan, S. Fan, Z. Du, and G. Zhang, "Retinal image classification by self-supervised fuzzy clustering network," *IEEE Access*, vol. 8, pp. 92 352–92 362, 2020.

[9] J. Wang, Y. Bai, and B. Xia, "Simultaneous diagnosis of severity and features of diabetic retinopathy in fundus photography using deep learning," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 12, pp. 3397–3407, 2020.

[10] L. Qiao, Y. Zhu, and H. Zhou, "Diabetic retinopathy detection using prognosis of microaneurysm and early diagnosis system for non-proliferative diabetic retinopathy based on deep learning algorithms," *IEEE Access*, vol. 8, pp. 104 292–104 302, 2020.

[11] S. Qummar, F. G. Khan, S. Shah, A. Khan, S. Shamshirband, Z. U. Rehman, I. Ahmed Khan, and W. Jadoon, "A deep learning ensemble approach for diabetic retinopathy detection," *IEEE Access*, vol. 7, pp. 150 530–150 539, 2019.

[12] R. Wang, B. Chen, D. Meng, and L. Wang, "Weakly supervised lesion detection from fundus images," *IEEE Transactions on Medical Imaging*, vol. 38, no. 6, pp. 1501–1512, 2019.

[13] X. Zeng, H. Chen, Y. Luo, and W. Ye, "Automated diabetic retinopathy detection based on binocular siamese-like convolutional neural network," *IEEE Access*, vol. 7, pp. 30 744–30 753, 2019.

[14] C.-H. Hua, K. Kim, T. Huynh-The, J. I. You, S.-Y. Yu, T. Le-Tien, S.-H. Bae, and S. Lee, "Convolutional network with twofold feature augmentation for diabetic retinopathy recognition from multi-modal images," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 7, pp. 2686–2697, 2021.

[15] S. Roychowdhury, D. D. Koozekanani, and K. K. Parhi, "Dream: Diabetic retinopathy analysis using machine learning," *IEEE Journal of Biomedical and Health Informatics*, vol. 18, no. 5, pp. 1717–1728, 2014.

[16] B. Schweisthal and M. Lascu, "Multiple convolutional neural networks for diabetic retinopathy classification," in *2021 International Conference on e-Health and Bioengineering (EHB)*, 2021, pp. 1–4.

[17] Z. Gao, J. Li, J. Guo, Y. Chen, Z. Yi, and J. Zhong, "Diagnosis of diabetic retinopathy using deep neural networks," *IEEE Access*, vol. 7, pp. 3360–3370, 2019.

[18] G. Zhong, K. Zhang, H. Wei, Y. Zheng, and J. Dong, "Marginal deep architecture: Stacking feature learning modules to build deep learning models," *IEEE Access*, vol. 7, pp. 30 220–30 233, 2019.

[19] Y. Jusman, M. A. F. Nurkholid, and F. Utomo, "Prostate image classification using pretrained models: Googlenet and resnet-50," in *2021 15th International Conference on Signal Processing and Communication Systems (ICSPCS)*, 2021, pp. 1–6.

[20] M. Ye, N. Ruiwen, Z. Chang, G. He, H. Tianli, L. Shijun, S. Yu, Z. Tong, and G. Ying, "A lightweight model of vgg-16 for remote sensing image classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 6916–6922, 2021.

[21] C. Wang, D. Chen, L. Hao, X. Liu, Y. Zeng, J. Chen, and G. Zhang, "Pulmonary image classification based on inception-v3 transfer learning model," *IEEE Access*, vol. 7, pp. 146 533–146 541, 2019.

[22] H. Pan, Z. Pang, Y. Wang, Y. Wang, and L. Chen, "A new image recognition and classification method combining transfer learning algorithm and mobilenet model for welding defects," *IEEE Access*, vol. 8, pp. 119 951–119 960, 2020.

[23] Y. Wang, J. Yan, Q. Sun, J. Li, and Z. Yang, "A mobilenets convolutional neural network for gis partial discharge pattern recognition in the ubiquitous power internet of things context: Optimization, comparison, and application," *IEEE Access*, vol. 7, pp. 150 226–150 236, 2019.

[24] H. Alhichri, A. S. Alswayed, Y. Bazi, N. Ammour, and N. A. Alajlan, "Classification of remote sensing images using efficientnet-b3 cnn model with attention," *IEEE Access*, vol. 9, pp. 14 078–14 094, 2021.

[25] J. Li, N. Allinson, D. Tao, and X. Li, "Multitraining support vector machine for image retrieval," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3597–3601, 2006.

[26] G. Shi, C. S. Chan, W. J. Li, K.-S. Leung, Y. Zou, and Y. Jin, "Mobile human airbag system for fall protection using mems sensors and embedded svm classifier," *IEEE Sensors Journal*, vol. 9, no. 5, pp. 495–503, 2009.

[27] V. K. Gupta, A. Gupta, D. Kumar, and A. Sardana, "Prediction of covid-19 confirmed, death, and cured cases in india using random forest model," *Big Data Mining and Analytics*, vol. 4, no. 2, pp. 116–123, 2021.

[28] L. Jiang, H. Zhang, and Z. Cai, "A novel bayes model: Hidden naive bayes," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 10, pp. 1361–1371, 2009.

[29] S. Tsang, B. Kao, K. Y. Yip, W.-S. Ho, and S. D. Lee, "Decision trees for uncertain data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 1, pp. 64–78, 2011.

[30] S. Zhang, "Challenges in knn classification," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2021.

[31] "International competition on ocular disease intelligent recognition. available: https://odir2019.grand-challenge.org," 2019.

[32] "Kaggle diabetic retinopathy detection competition with kaggle dataset.available: https://www.kaggle.com/c/diabetic-retinopathy-detection," 2015.

[33] E. Decencière*et al.*, ""feedback on a publicly distributed image data-base: The messidor database,*Image Anal. Stereology*," vol. 33, no. 3, p. 231–234, 2014.

[34] "Kaggle aptos 2019 blindness detection competition using diabetic retinopathy dataset. available: https://www.kaggle.com/c/aptos2019-blindness-detection," 2019.

[35] P. Porwal*et al.*, "Indian diabetic retinopathy image dataset (idrid):a database for diabetic retinopathy screening research." vol. 3, no. 3, p. 25, 2014.