2.1. Traditional models in diabetic retinopathy

In the field of DR, different state-of-the-art models called ML models are used for the classification of DR images. This includes decision trees, RF, and SVM(Suganthi et al., 2020) Adaboost, naïve Bayes, logistic regression, neural networks(Bhatia et al., 2016), the Adaboost algorithm, CNN with LSTM, E-densenet (Yasashvini et al., 2022),VGG 16, VGG 19, AlexNet, Inception v3, Inception ResNet, hybrid CNN, Google Net, and Densenet(Qomariah et al., 2019; Kolla and Venugopal, 2021; Asia et al., 2022; Yasashvini et al., 2022) DL models such as COA-DN, DRNET, VGGNET, ResNet-50, Inception-V3, ResNet101, and VGG19 were used. The detailed description is provided in the subsequent subsection.

2.1.1. ML models

Machine learning can learn from its features effectively and produce success outcome. In recent times, previous papers that treat the classification of DR levels using techniques such as decision tree, RF, and SVM(Suganthi et al., 2020), adaboost, naïve bayes, logistic regression, neural network (Bhatia et al., 2016), adaboost algorithm, CNN with LSTM, E-densenet (Yasashvini et al., 2022), VGG 16, VGG 19, AlexNet, Inception v3, Inception ResNet, hybrid CNN, Google Net, and DenseNet (Qomariah et al., 2019; Kolla and Venugopal, 2021; Asia et al., 2022; Yasashvini et al., 2022). A few recent studies on DR for disease detection and treatment are highlighted in the following paragraphs. (Suganthi et al., 2020) conducted research on detecting characteristics of disease pattern. Generally, DR can be classified as normal, moderate, mild, and severe. In this work, the classification of DR, including decision tree, RF, and SVM, has been applied to detect DR disease classification. Among this classification, DT found to be the superior performance of classifiers. However, this classifier has some drawbacks including lack of detection of additional features, high error and low accuracy. (Bhatia et al., 2016) explored an algorithm that is based on ensemble learning. The dataset is processed; features were extracted in this approach. The application of an ensemble of ML classification helps to predict the presence of DR with the proposed algorithm to improve the accuracy and flexibility of the model that is comprised of alternating DT, adaboost, Naïve Bayes, RF, and SVM. On the other hand, the proposed algorithm suffers from the lack of a computation problem. (Pratt et al., 2016) explored a CNN-based architecture to detect DR from digital fundus images. The proposed model has performed colour normalisation, image resizing and data augmentation. This dataset comprises 5000 images for validation. The proposed model has achieved the accuracy of 75%, sensitivity of 30% and specificity of 90%.

(Qomariah et al., 2019) have proposed a CNN architecture based on SVM for the detection of DR. The model is trained on 77 images from base 12 and 70 images from base 13. The model has used 224*224 for the VGG Net architecture, 227*227 for AlexNet, and 229*229 for the ResNet and Inception architectures. The model has compared the performance of VGG 16, VGG 19, AlexNet, Inception v3, Inception ResNet, Google Net, and DenseNet and has achieved the accuracies of 87.50%, 75%, 79.17%, 75%, 91.67%, and 83.33%, respectively. (Kolla and Venugopal, 2021) have developed an automated system to detect DR using 80000 fundus images from the Kaggle dataset. The binary CNN model is proposed to train the images using the max pooling technique and dense layers that are binarized. This model has compared the performance of VGG 16, Inception v3, AlexNet, DenseNet, ResNet, and BCNN using Inception v3. Among these, BCNN has achieved five-class binary classification accuracy of 91.04% respectively. (Asia et al., 2022) have proposed a CNN architecture to automatically diagnose DR. This dataset comprises 1607 images. This model has used image cropping, resizing, regularisation, and augmentation. The proposed CNN is trained using ResNet 101 and has achieved an accuracy of 98.88% with a loss of 34.99%. (Yasashvini et al., 2022) have proposed an algorithm for an automated model for DR detection using NPDR images and have identified DR using a CNN and transfer learning algorithm. The proposed methodology has utilised the images from Kaggle dataset. The model has performed preprocessing including cropping, resizing, masks, image smoothing and blending. The model has compared the performance of the AdaBoost algorithm, e-DenseNet, CNN with LSTM, CNN, CNN with DenseNet, and CNN with ResNet and has achieved the accuracy of 88.21%, 91.6%, 90%, 75.61%, 96.22%, and 93.18%. It is observed that proposed CNN with DenseNet is best model that outperform than others.

Amongst all these, ResNet have produced effective results on fundus images. The performance of CNN is less than other models. As the results of previous studies indicated that ML methods are ineffective when it comes to fundus images and it is proving only suboptimal results. In addition to this, it suffers from high generalisation error. As a result, the researcher move on to DL models to extract patterns and relationship from high-dimensional image data.

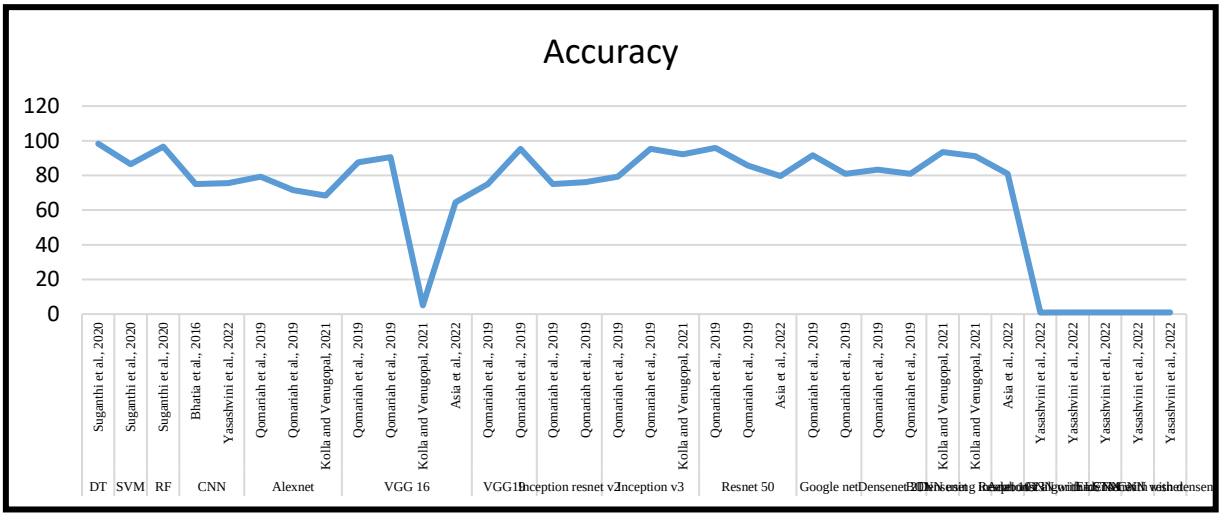

Figure 1. Accuracy of ML models

An effective, quick, and precise method for identifying and classifying medical images is provided by Yolo (Wei et al., 2025). YOLO is not new to classify diabetic retinopathy images. YOLO has been used widely to detect DR automatically to overcome the challenges of detecting small objects difficulty in digital images. (Alyoubi et al., 2021) have proposed a methodological study using different preprocessing techniques including enhancing, noise removing, cropping, colour normalisation and data augmentation for DR detection. This model has used batch normalisation layer, adam optimiser for DDR dataset. This proposed framework has offered an effective accuracy to locate all lesion types and categorise the DR images into stages. YOLO v5 has offered promising results in detecting clinical pictures of DR (Santos et al., 2022). (Wahab Sait, 2023) demonstrates that the information obtained by YOLO v7 assisted to detecting DR images in a short span of time. Also, the study assisted to finding out the tiny spots in DR severity. Shah et al. (2024) have proposed YOLO v8 to detect fundus images of DR using machine learning SVM classification. The model has extracted retinal fundus disease triage and has achieved high computational accuracy. (Devi Appari et al., 2024) have proposed a yolo model for the detection of DR through zero padding, regression, object detection and classification. This model is compared with mainstream models, including CNN, VGG 16, ResNet 50, and SVM. The novel YOLOv8 model has achieved the best fit in comparison to other models. (Rizzieri et al., 2024) have proposed an architecture called Yolo V8 and Yolo V9 using the Messidor dataset. This model has used 100 images for DR classification. YOLO v8 and v9 have achieved excellent performance to evaluate medical images and diagnose DR in a healthcare setting. However, this study is not good paradigm as it fails to use image preprocessing processing, data augmentation in the proposed model. (Akella and Kumar, 2024) have proposed YOLO v3 to assess DR retinal fundus images. The model has measured the

performance on the basis of colour and fundus images. This has classified five classes with more than 80% accuracy on DR images. (Kumar and Priyadharsini, 2021) Diagnosed DR images with 93% accuracy from DR images with ResNet and YOLO architecture. Thus, analysing existing research reveals that YOLO has been largely used to classify DR images.

2.1.2. Hybrid Models

Several hybrid models have been developed to classify DR images and detect DR. Previous studies have proposed DR detection using the VGG 16 and XGBoost classifiers and the DenseNet 121 network (Mohanty and Prakash, 2014), DL with transfer learning(Gupta et al., 2019), VGG 16 architecture and Logistic regression classifier (Singh and Dobhal, 2024), VGG 16 and Inception V3 (Shazia et al., 2024), CNN and RCNN (Kumari et al., 2024), hybrid DCNN, E-denseNet, ResNet 50 & Inception V3, IR-CNN (Ali et al., 2023), Mobile Net V2 using SVM classifier (Lahmar and Idri, 2023), ResNet 18 and GoogLeNet (Butt et al., 2022). (Mohanty et al., 2023) have proposed a twin architecture that combine VGG 16 and XG boost classifier and DenseNet 121 network. In the feature extraction stage, this study used an Gaussian filter integration with ben Ben Graham approach and cropping the images to the region of interest. The decision of presence of DR was achieved using the DenseNet 121 model classifier. This model has achieved an accuracy with 97.30%. (Gupta et al., 2019) proposed a model to detect and prognose DR using DL with transfer learning. In this study, preprocessing images from the IDRID dataset by applying data augmentation. This model has achieved an accuracy of 90% across different stages of DR. (Singh and Dobhal, 2024) introduced a classifier by utilising the VGG 16 architecture and logistic regression classifier for the detection of DR images. During the feature extraction, the VGG 16 architecture was used, and the model has achieved an impressive accuracy of 90.4%. (Shazia et al., 2024) proposed a deep hybrid model for the automatic detection of DR diseases. This model has used resizing and augmentation for classification of DR images. The proposed model is trained using VGG 16 and Inception V3, which has achieved an accuracy of more than 98%. (Kumari et al., 2024) proposed a model that preprocess the images on the basis of filtering and color enhancement, resizing and reduce over fit issues in the five datasets. The proposed methodology has used CNN and RCNN to predict the presence and severity of DR. As observed from the proposed model, both outperform well that show spatial and temporal images in DR that increases accuracy and resilience. Ali et al. (2023) proposed a model with twin DL models including ResNet50 and Inception v3. The proposed model has utilised image enhancement and data augmentation for preprocessing and extracting features with Inception v3 and ResNet50. This model has detected all stages of DR

and has obtained an accuracy of 96.85% with data augmentation. (Lahmar and Idri, 2023) detected DR with 88% accuracy using DenseNet and MobileNet V2 using an SVM classifier. (Butt et al., 2022) classified DR with 89.29% with ResNet and GoogleNet models using an SVM classifier.

2.2. Research gap

- Automatic detection of eye-related diseases is one of the emerging areas for academicians. Many researchers have come forward to solve the classification of DR through employing CNN and deep learning models.
- These models offered an efficient classification of DR detection and assisted in finding out the diseases and problems.
- As observed from the previous studies, it has been identified that traditional methods of VGG 16 are widely used. Although VGG 19 is also one of the conventional methods, the application of DR image classification is minimal. In addition to this, modern deep learning techniques, including YOLO, are used largely to categorise and predict DR issues.
- However, there is a lack of literature available on DR detection with the usage of new models of YOLO to detect DR.
- This thesis has utilised a pretrained conventional model, including VGG 16 & VGG 19, and a new YOLOv11 model used to classify DR images into healthy, mild, moderate, severe, and proliferative DR Images

CHAPTER-4

ANALYSIS

4.1. Introduction

This chapter show case a concise and precise description of the results that obtained from VGG 16, VGG 19 and YOLO 11 algorithm. The algorithm is performed with the help of the Keras framework. This helps to simulate the classification model for DR detection on a Google Colab Pro with 2 TB of storage, 25 GB of RAM, and a CPU-P 100. As discussed in the previous chapter, the image data generator class was used to preprocess the image with the help of normalisation, scaling, and conversion to an array of data for VGG 16 and VGG 19. The results derived from the model are presented in the subsequent section.

4.2. Data preparation, Augmentation and split

4.2.1. Data preparation

The present study utilised 224*224 size DR images obtained from Kaggle. These images predominantly assist to diagnosing DR and detect effectively. In order to make sure to have a good consistency with the models, pretrains were used. All the images were resized to 224*224 resolution. The primary purpose of resizing is to organise the images in a structured manner and map them into the existing categories of DR. There are five attributes presented in DR. This is segmented as healthy, mild, moderate, proliferative, and severe DR.

4.2.2. Class Balancing

As the Messidor image dataset has been affected with class imbalance, it is illustrated in the Figure 10. This imbalance has an effect when processing whole images. This makes the tasks to be highly complex. In order to resolve the problem, the present study adopts oversampling, and undersampling is illustrated in Figure to tackle class imbalance in the Messidor image dataset. A balanced distribution of the training dataset is presented in the Figure 13.

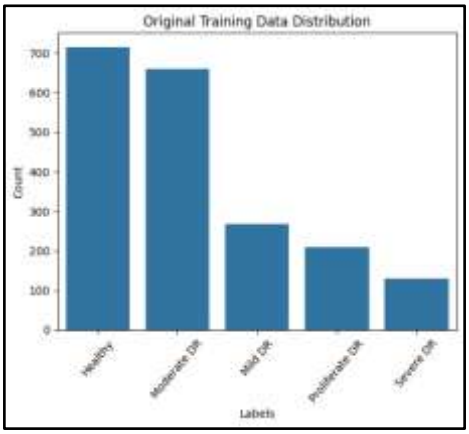

Figure 2: Class imbalance in original training data distribution

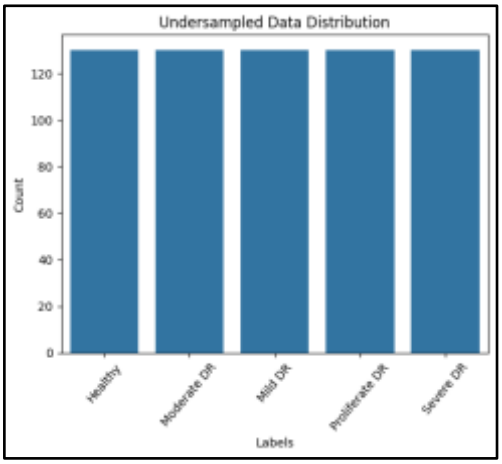

Figure 3: Under sampling

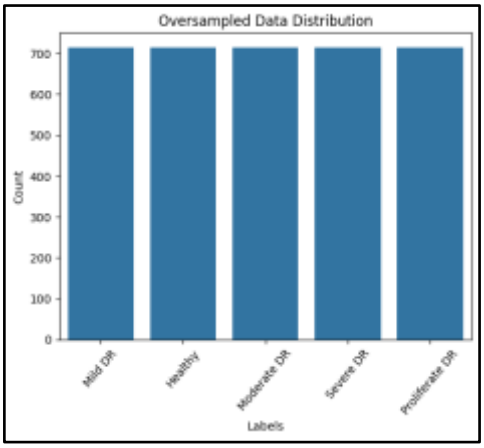

Figure 4: Oversampling

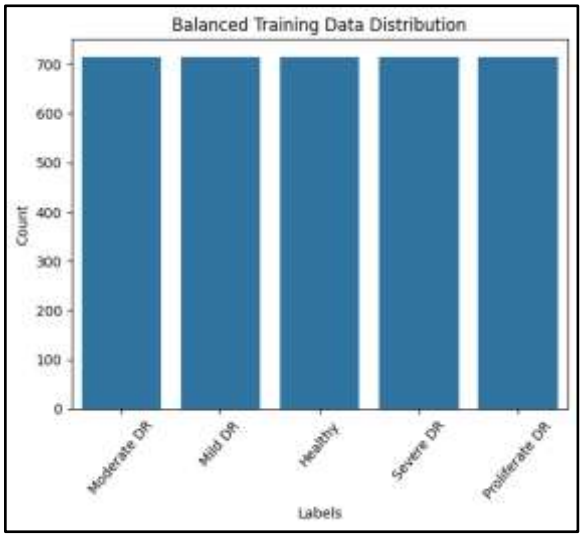

Figure 5: Balanced Training data

4.2.3. Augmentation

One of the most popular methods for creating images from the dataset is data augmentation. Data augmentation is a collection of techniques that increase the amount of training data without actively obtaining new data. In data augmentation, images can be created through rotation flips, cropping and padding, and zooming from the existing training data and generalising well to new data. The dataset images were better understood through the use of data augmentation. The present study used the Keras image data generator class that is illustrated in the figure 3 &4, and it makes sure that a variation of the images at each epoch will be obtained by the chosen model(VGG16 &VGG19). In addition to this, this has the potential to transform the images that will not expand the range of original images and to prevent overfitting problems. Moving on to one-hot encoding, it was used to change the labels because they are categorical features that are not applicable to the models in their current state.

Table 1: Parameter setting for image augmentation

Method	Setting
Rotation range	40
Width shift range	0.2
Height shift range	0.2
Shear range	0.2
Zoom range	0
Horizontal flip	True
Flip mode	Nearest

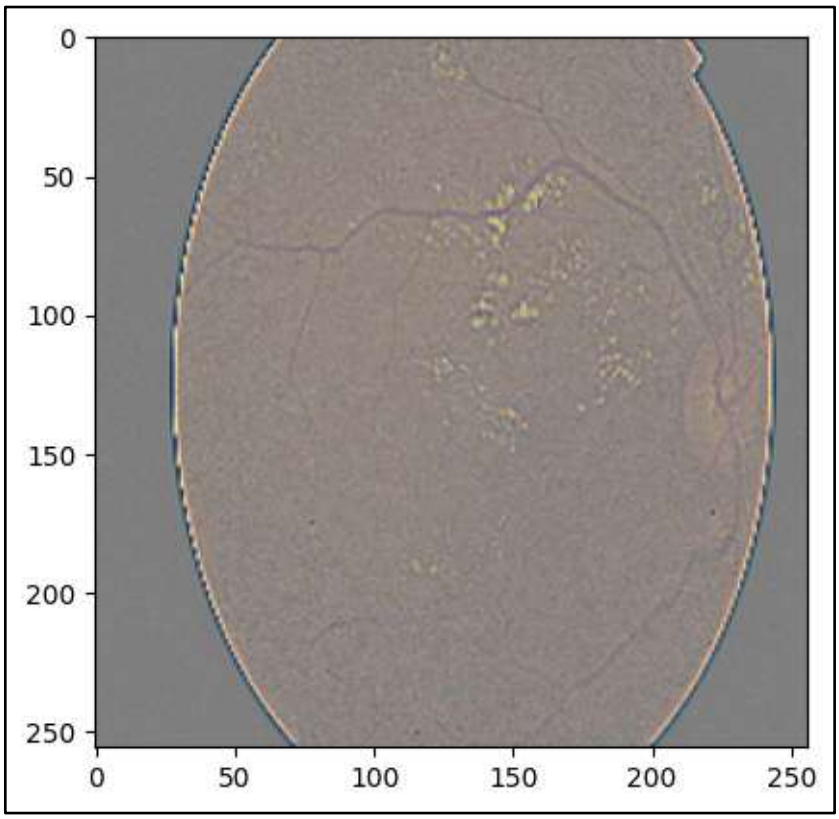

Figure 6: Original image

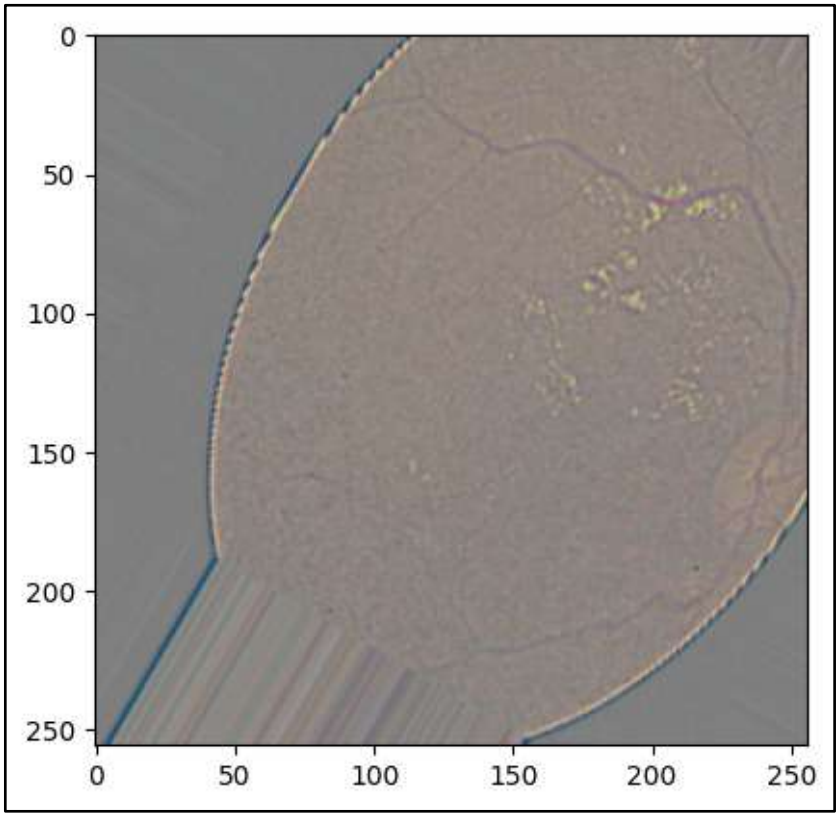

Figure 7: Augmented image

4.2.4. Training and Splitting

Initially, the data was loaded and pre-processed by resizing the image into 224 x 224 pixels and converting it to the RGB channel. The process of transformation is to get the image ready for VGG 16 and VGG 19. The intensities of the pixels were scaled from 0 to 1. Moving on to the central point of the study is the selection of training samples. This has a more significant effect on classification accuracy rather than the algorithm itself. The selection of samples is on the basis of how different classifiers are affected by the size of training data. This problem is particularly important in deep learning techniques, where a large number of well-labelled training samples with proper labelling is required to keep the classifier from overfitting (Pawluszek-Filipiak and Borkowski, 2020). Following the theoretical knowledge, the data has been divided into training, validation, and testing as 80%, 10%, and 10%, respectively.

4.2.5. Implementation

The experiments were conducted using the Google Colab web IDE. The term IDE refers to this IDE offering a GPU instance at no cost. The implementation was carried out using the open-source Keras library running on the TensorFlow backend. The weights of the pretrained models were imported from Keras. In this experiment, three base models such as VGG 16, VGG 19, and Yolo 11 were used. When applying these models in the training dataset, the learning curves were used to track the model's learning performance. In addition to this, the learning curves were used to detect overfit or underfit issues in the model learning. This helps to diagnose the generalisation behaviour that exists in the model.

4.3. Model training

4.3.1. VGG 16

This model tested on 3570 images and its has been performed with preprocessing and extraction. The labelling was also performed on the images. The sequential structure of VGG 16 model is shown in the Figure. This model is trained over 50 epoch and thirty iterations were employed. An epoch is a single loop across the entire training dataset. This was trained using adam optimiser. This is considered to be the efficient and best optimiser because it requires minimum time to train the model. In addition to this, it was trained using cross entropy loss function, a batch size of 64 and a softmax activated layer. The learning rate of 0.001 is default parameter for the model.

Table 2: Sequential model of VGG 16

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, 224, 224, 3)	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1,792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36,928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73,856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147,584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295,168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590,080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590,080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1,180,160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2,359,808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2,359,808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2,359,808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2,359,808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2,359,808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
average_pooling2d (AveragePooling2D)	(None, 1, 1, 512)	0
flatten (Flatten)	(None, 512)	0
dense (Dense)	(None, 224)	114,912
dropout (Dropout)	(None, 224)	0
dense_1 (Dense)	(None, 5)	1,125

Total params: 14,838,725 (56.61 MB)
Trainable params: 116,037 (453.27 KB)
Non-trainable params: 14,714,688 (56.13 MB)

This model has achieved an accuracy of 98.4% in this regard. This also occurred a loss of 0.0875 over the course of 50 epochs. The accuracy, validation loss and confusion matrix is described in the subsequent subsection of Training performance.

4.3.2. VGG 19

Taking a look at the structure of VGG 19 model is shown in the Figure. This model is trained over 50 epoch and thirty iterations were employed. An epoch is a single loop across the entire training dataset. This was trained using adam optimiser. This is considered to be the efficient and best optimiser because it requires minimum time to train the model. In addition to this, it was trained using cross entropy loss function, a batch size of 64 and a softmax activated layer. The learning rate of 0.001 is default parameter for the model.

Table 3: Sequential model of VGG 19

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv4 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv4 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv4 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
average_pooling2d (Average Pooling2D)	(None, 1, 1, 512)	0
flatten (Flatten)	(None, 512)	0
dense (Dense)	(None, 256)	131328

This model has achieved an accuracy of 82.74% in this regard. This also occurred a loss of 0.6478 over the course of 50 epochs. The accuracy, validation loss and confusion matrix is described in the subsequent subsection of Training performance.

4.3.3. Yolo 11

For yolo 11, the images were trained, validated and tested in a ratio of 80:10:10. There is no adoption of image preprocessing for YOLO 11 images. The pre-trained model was used to compare and evaluate the performance of the model using raw data collected and downloaded from the Kaggle dataset. The results are illustrated in the training performance section.

4.4. Training performance

4.4.1. Training performance of VGG 16

The training curve demonstrated a favourable model learning rate, and the validation curve demonstrated how the model generalised. The learning curves of the models are shown in the

subsequent figure, and it did not get generalised. The reason is that there is a large gap that exists between training and validation performance. The training loss of the model remains low and stable and is well performed in training data. Although the learning curve showed that the model was learning well, testing data was not generalised during the pre-established epochs. The indication of poor generalisation is that training accuracy increases and the validation accuracy improves initially but plateaus and fluctuates after a certain epoch in VGG 16. This indicates that the training curve has higher training accuracy, and decreasing training loss looks to be satisfactory as shown in the Figure 12 and 13.

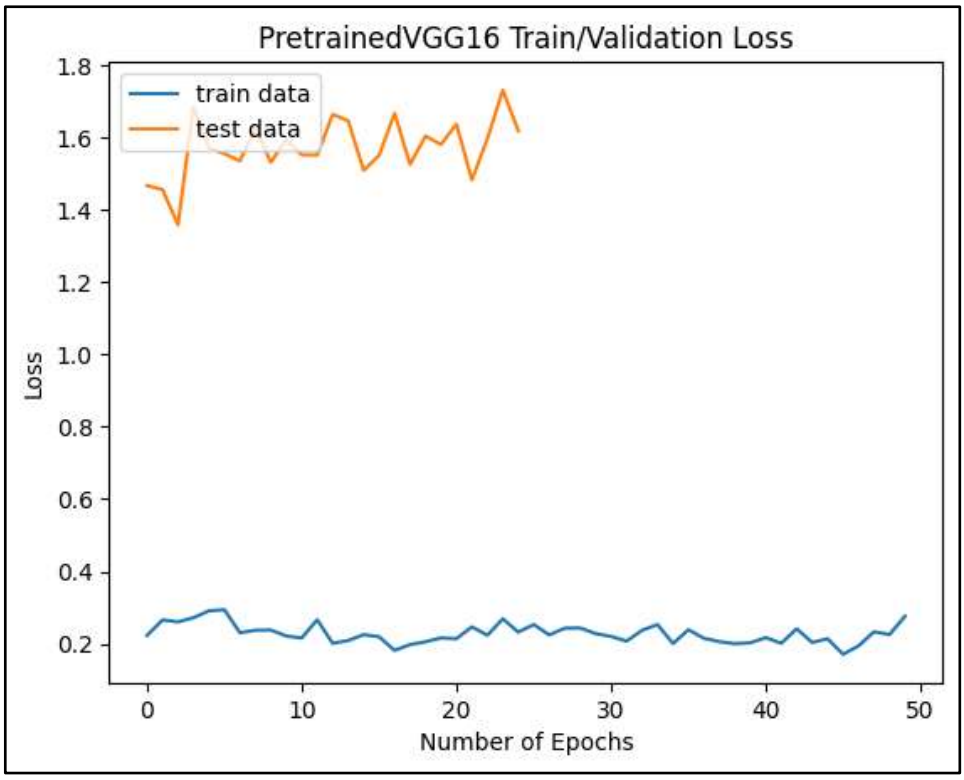

Figure 8: Training and validation loss – VGG 16

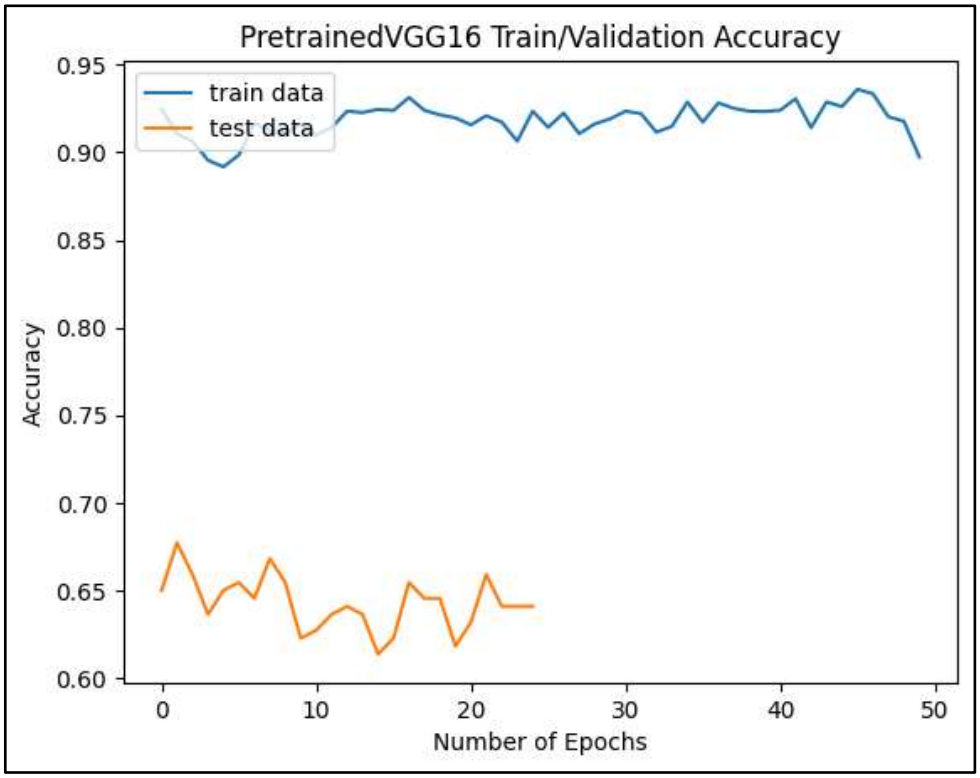

Figure 9: Training and testing accuracy -VGG 16

Training and validation accuracy VGG 16: The VGG 16 model accuracy and loss during training and validation were evaluated. Training accuracy remained unchanged at 97.42%. The validation accuracy was 70.40. This value was not constant and exhibits a nonlinear and inconsistent pattern as the number of repetitions increased. We employed five classes and evaluated the accuracy of using the approach. The model achieved an F1 score of 70.43% with a recall of 71.27%.

Confusion matrix: Figure 18 shows the confusion matrix of the classifier of VGG 16. The present model reveals that the VGG 16 model can classify the five severities of DR with the highest ratio to the diabetic retinopathy, starting from healthy (not DR), moderate DR, mild DR, proliferate DR, and lastly the severe DR.

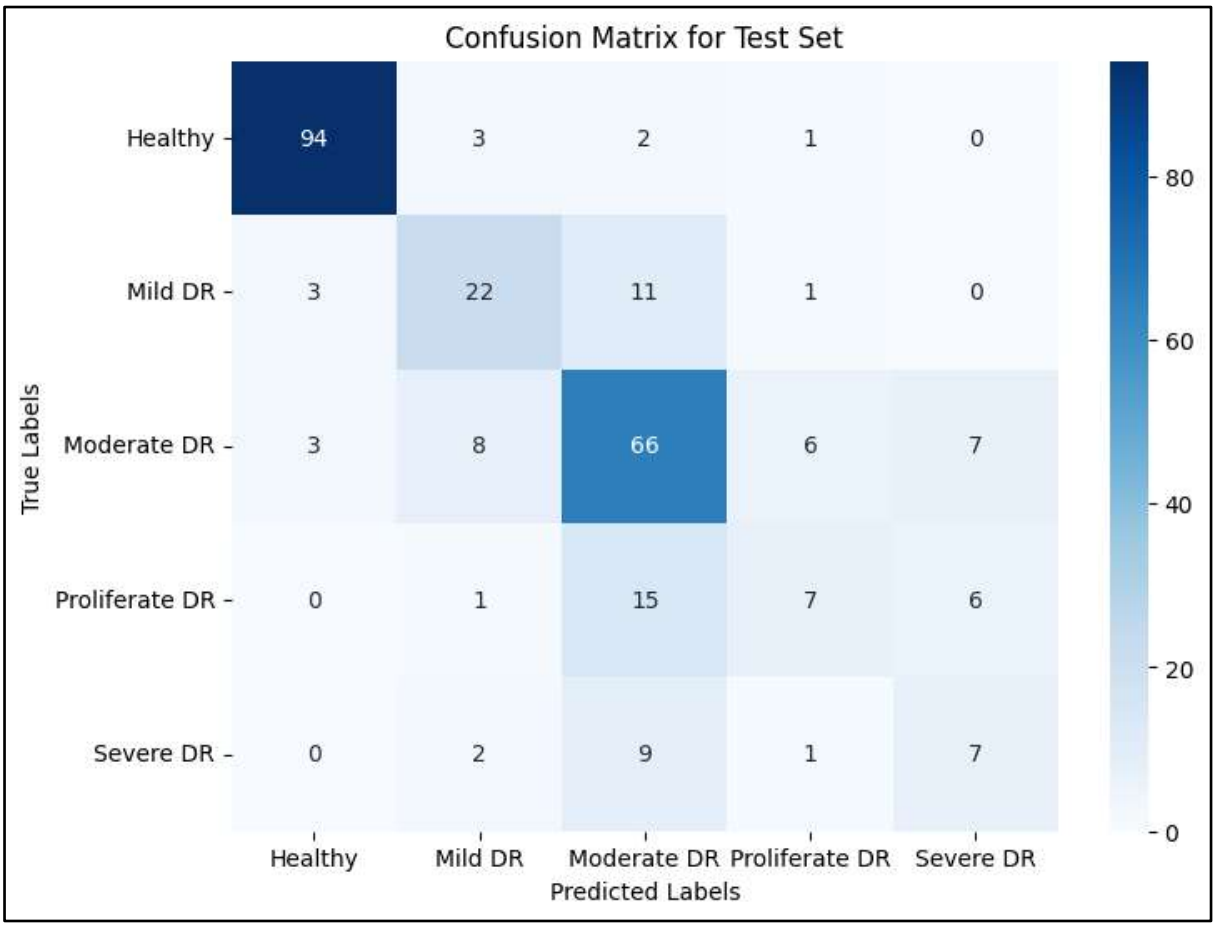

Figure 10: Confusion matrix for VGG 16

The present study adopts a multi-class deep learning classification model to incorporate and classify five classes of diabetic retinopathy, including healthy (not DR), mild DR, moderate DR, proliferative DR, and severe DR, using the VGG 16. The model architecture was built using a VGG 16 for feature extraction, and a fully linked network was used for classification. The proposed model was tested for precision, F1 score, recall, accuracy, and AUC.

On the basis of validation of DR images, the VGG 16 can identify various DR. For Healthy DR, precision was 94%, recall was 94%, the F1 score was 94%, and the AUC was 98%. For Mild DR, precision was 61%, recall was 59%, the F1 score was 60%, and the AUC was 86%. For moderate DR, precision was 64%, recall was 73%, the F1 score was 68%, and the AUC was 85%. For Proliferate DR, precision was 44%, recall was 24%, the F1 score was 31%, and the AUC was 81%. For severe DR, precision was 35%, recall was 37%, the F1 score was 36%, and the AUC was 76%. To sum this up, the findings showed that the VGG 16 provided satisfactory classification performance with 71% accuracy.

On the basis of training of DR images, the VGG 16 can identify different DR. The precision of Healthy DR was 100%, recall was 99%, and the F1 score was 100%. The precision for Mild DR was 97%, recall was 99%, and F1-score was 98%. The precision for moderate DR was 00%, recall was 95%, and the F1 score was 97%. The precision, recall, and F1-score for proliferate DR were 99%. The precision for severe DR was 98%, recall was 100%, and the F1 score was 99%. The values of these metrics in the classification of DR images were all over 98%. It is obvious that the VGG 16 could contribute to detecting the DR images efficiently. Thus, VGG 16 is anticipated to help create a model for identifying diabetic retinopathy, preventing visual impairment and blindness, and potentially saving vision loss of patients.

The evaluation of the model was calculated with the ROC curve. This is regarded as one of the best performance measures for classification models. The term ROC curve here refers to how well a model can differentiate between the classes. Furthermore, the micro average and macro average are essential to elucidate the overall performance of the model. Among these two, the micro average is important as it is given more weight to sum up individual false, false negative, and true positive when the datasets are imbalanced. On the other hand, the average precision and recall determine the macro average. As observed from the results, the micro-AUC for VGG 16 is 0.92. The ROC curve for VGG 16 is shown in Figure 19.

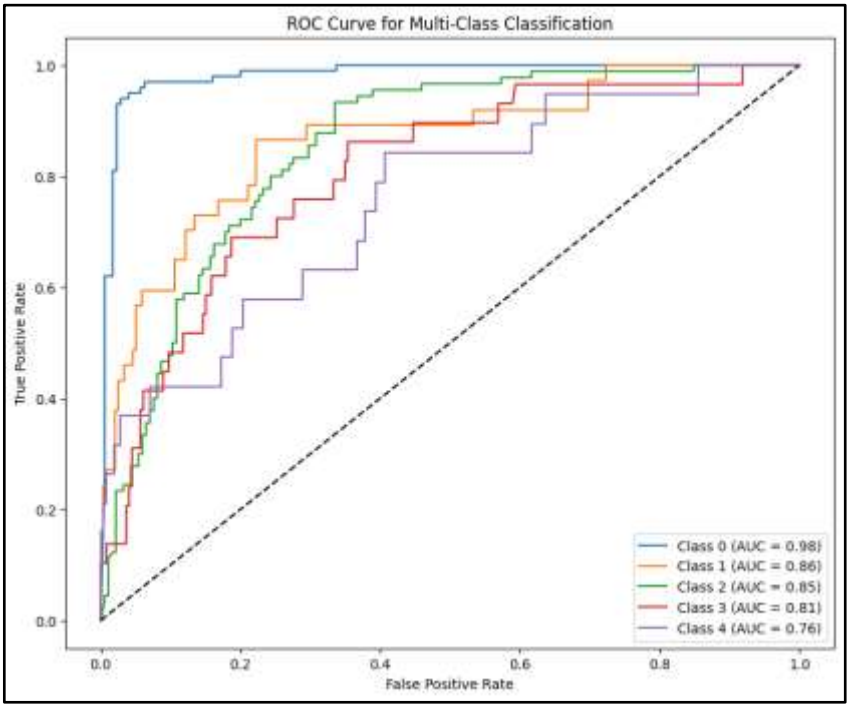

Figure 11: ROC Curve – VGG 16

In order to test the results of VGG 16, the prediction was obtained for a random image. Figure shows that after the model was trained, it was assessed, indicating that the likelihood that the severity of actual DR is different from the predicted image.

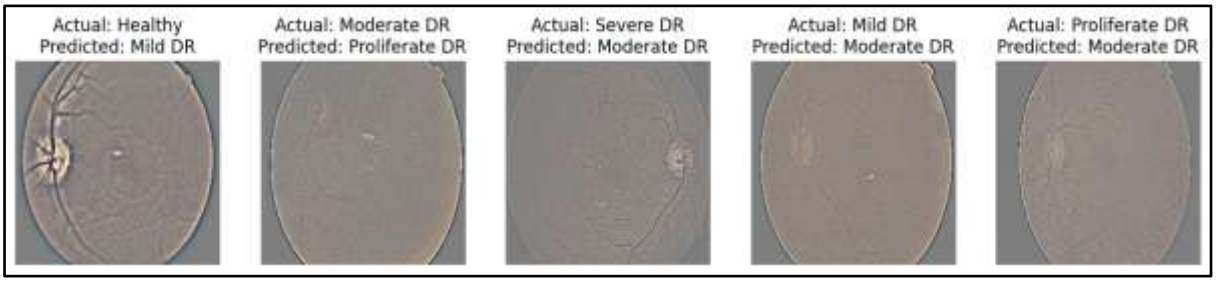

Figure 12: Image prediction for VGG 16

4.4.2. Training performance of VGG 19

The learning curves of the model with VGG 16 indicate that the training accuracy steadily improves over epochs even though it fluctuates slightly. The performance of the learning curves indicates that there is a significant gap between training accuracy and testing accuracy. This model is effectively learning the DR images in the training dataset. This indicates that the model could not perform well using unseen data. The characteristics showed how model performance decreased as the number of epochs increased.

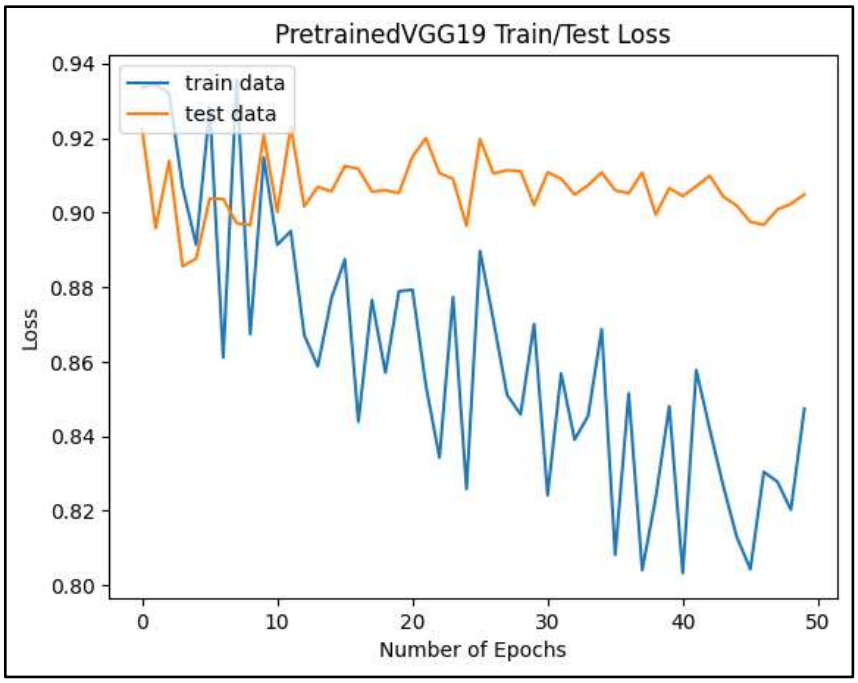

Figure 13: Training and testing loss – VGG 19

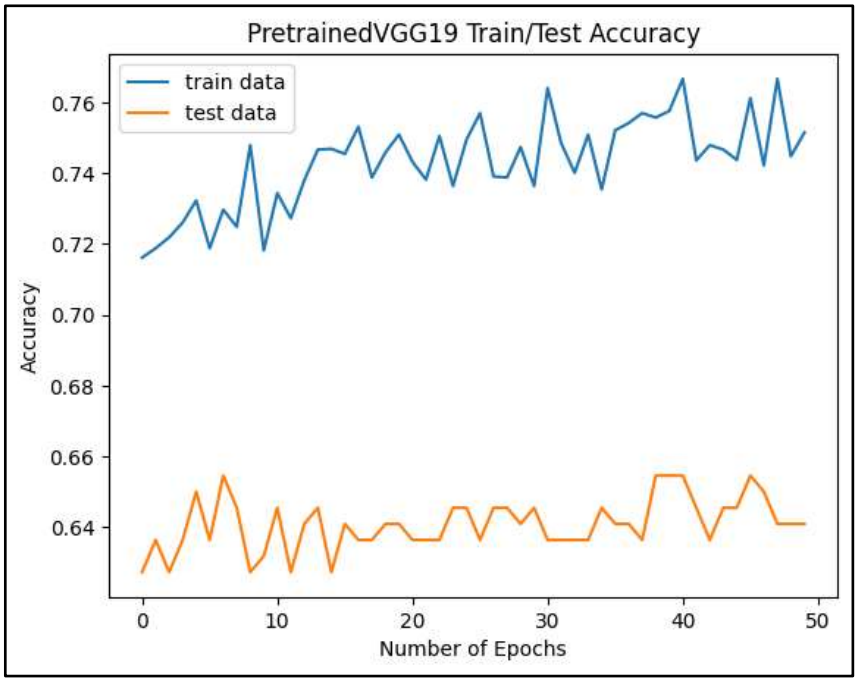

Figure 14: Training and testing accuracy – VGG 19

Training and validation accuracy VGG19: In the VGG 19 model, the total number of parameters and non-trainable parameters were 20,156,997 and 0, respectively. The target size of the model was 224*224, and it has 132,613 trainable parameters and 0 non-trainable parameters. This model has achieved the training and validation accuracy of 74.82% and 64.09%, respectively, in the 50^{th} epoch. In addition to this, it has a validation loss of 0.9049 and a training loss of 0.8481. This represents that the validation loss and training accuracy are greater than the respective validation accuracy and training loss. This suggests that overfitting and complexity are present in the model. This highlights that the numerous parameters allow it to memorise the training data; as a result, it can only extract information but is unable to create it.

Confusion matrix: Figure 21 presents the confusion matrix of the classifier of VGG 19. This model shows that the VGG 19 model can classify five severities of DR with the highest ratio to the diabetic retinopathy, starting from healthy (not DR), moderate DR, mild DR, proliferate DR, and lastly the severe DR.

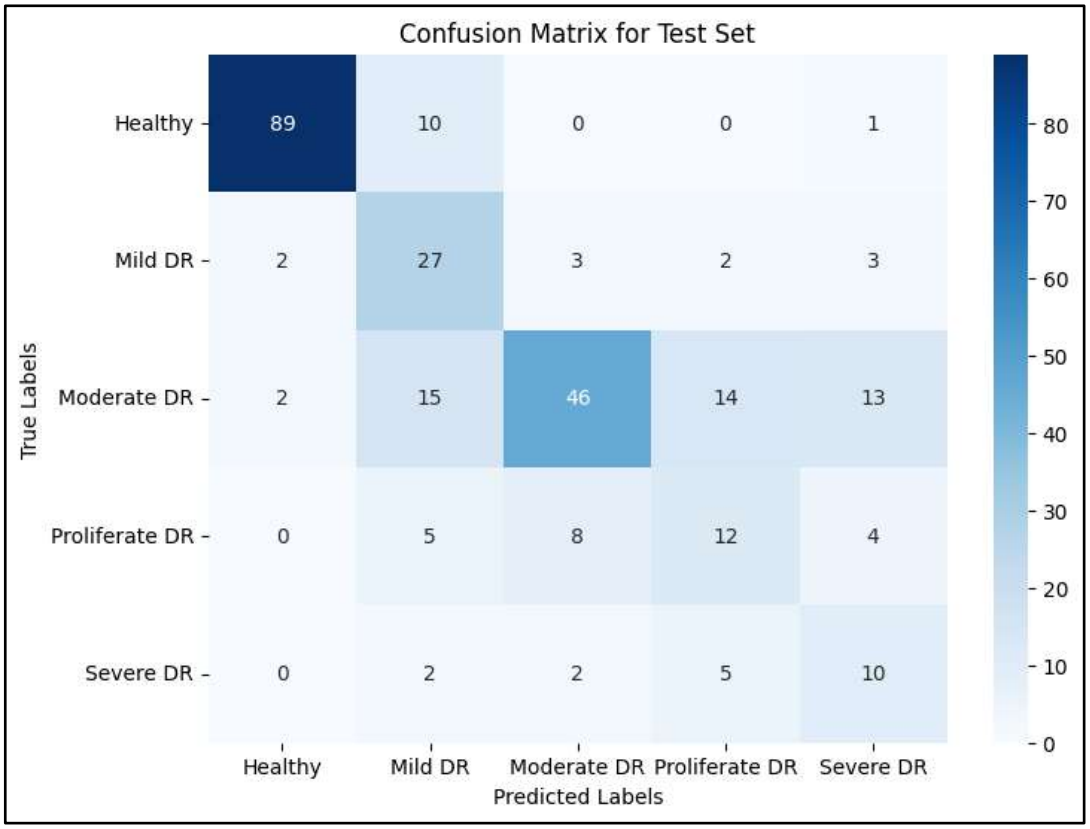

Figure 15: Confusion matrix – VGG 19

This study adopts the VGG 19 classification model to incorporate and classify five classes of DR, including healthy (not DR), mild DR, moderate DR, proliferative DR, and severe DR. The proposed model was tested for precision, F1 score, recall, accuracy, and AUC. This was evaluated with validation data and training data. On the basis of validation of DR images, the VGG 19 can identify various DR. For Healthy DR, VGG 19 achieved 96% precision, 89% recall, and a 92% F1 score. For mild DR, this model has achieved 46% precision, 73% recall, and 56% F1 score. For moderate DR, this model has achieved 78% of precision, 51% of recall, and 62% of F1-score. For proliferate DR, this model has achieved 36% of precision, 41% of recall, and 39% of F1-score. For severe Dr., this model has achieved 32% of precision, 53% of recall, and 40% of F1-score. On the whole, the overall accuracy of the model was 67%.

On the basis of training of DR images, the VGG 19 can identify various DR. For Healthy DR, VGG 19 achieved 96% precision, 96% recall, and 96% F1 score. For mild DR, this model has achieved 78% precision, 86% recall, and 82% F1 score. For moderate DR, this model has achieved 88% of precision, 51% of recall, and 65% of F1-score. For proliferate DR, this model has achieved 78% of precision, 86% of recall, and 82% of F1-score. For severe Dr, this model

has achieved 79% of precision, 95% of recall, and 86% of F1-score. On the whole, the overall accuracy of the model was 83%, with 82.74% recall and an 81.93% F1 score. The model performance is quite lower than the previous model (VGG16).

Moving on to the ROC curve, which shows the extent to which the model can differentiate between the classes. The multiclass classification with VGG 19 shows that this model has achieved an AUC of 0.99 for healthy DR, 0.85 for mild DR, 0.85 for moderate DR, 0.79 for proliferate DR, and 0.87 for severe DR. All these values were closer to 1, indicating that the classification performance was high in the model. The micro-AUC for VGG 19 is 0.90. The ROC curve for VGG 19 is shown in Figure.

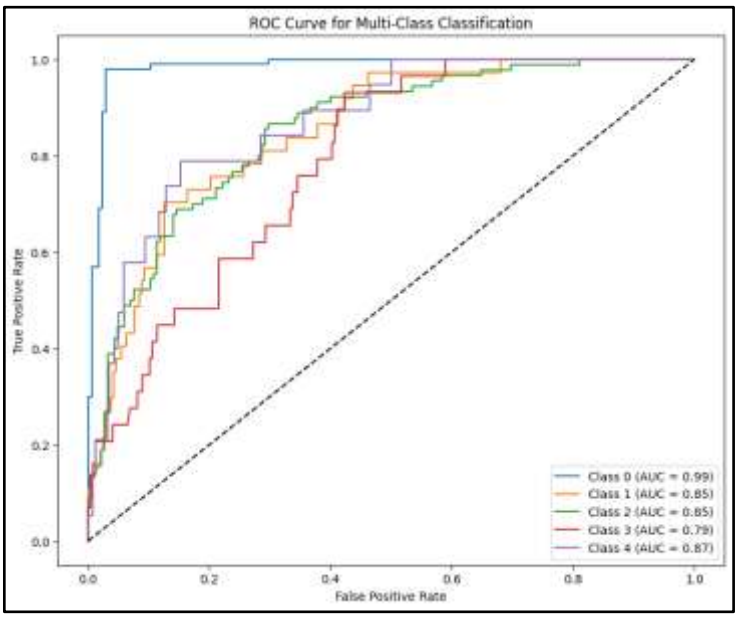

Figure 16: ROC curve for Multiclass classification for VGG 19

As the results were achieved, the next step is to test the prediction results with a random image. Figure shows that the training assessment revealed that the likelihood of severity of actual DR is different from the predicted image.

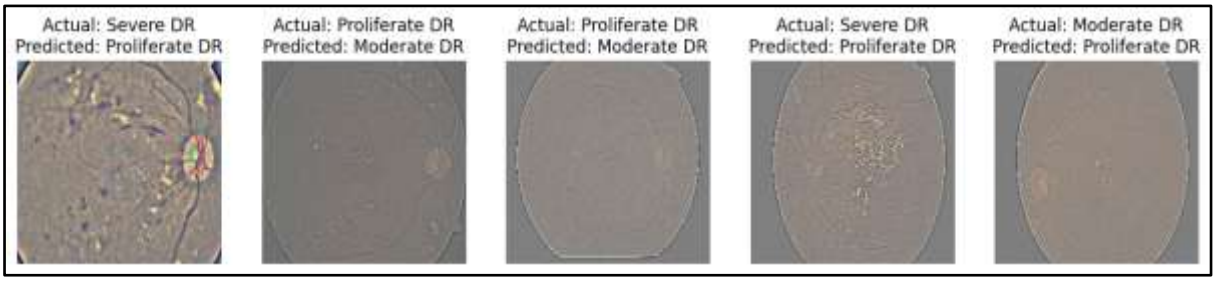

Figure 17: Prediction image of VGG 19

4.4.3. Training performance of Yolo 11

The training and validation loss curves of the model are shown in the figure. The Yolo11 model loss shows a steady decline represents effective learning. The training loss of the model was around 1.2 and rapidly decreased to 1. This indicates that the loss started high and it decreased slowly. The validation was 0.90, and it decreased to 0.60. All these indicate a minimum loss of the model.

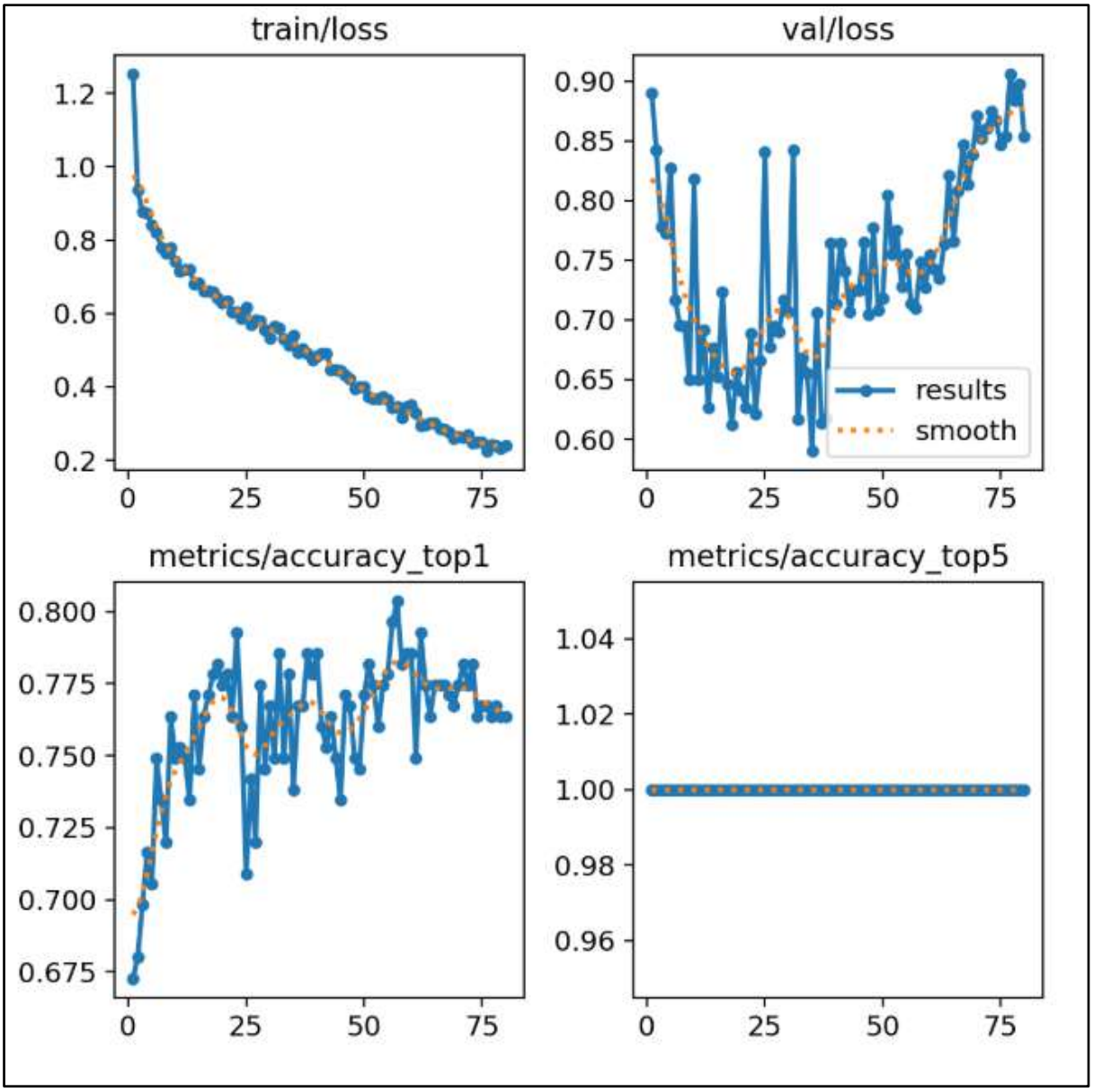

Figure 18: Training performance of Yolo 11

Confusion matrix: Figure 24 shows the confusion matrix of the classifier of Yolo 11. This model represents that it can classify five severities of DR with the highest ratio to diabetic retinopathy, starting from healthy, moderate, mild, proliferative, and lastly, severe DR.

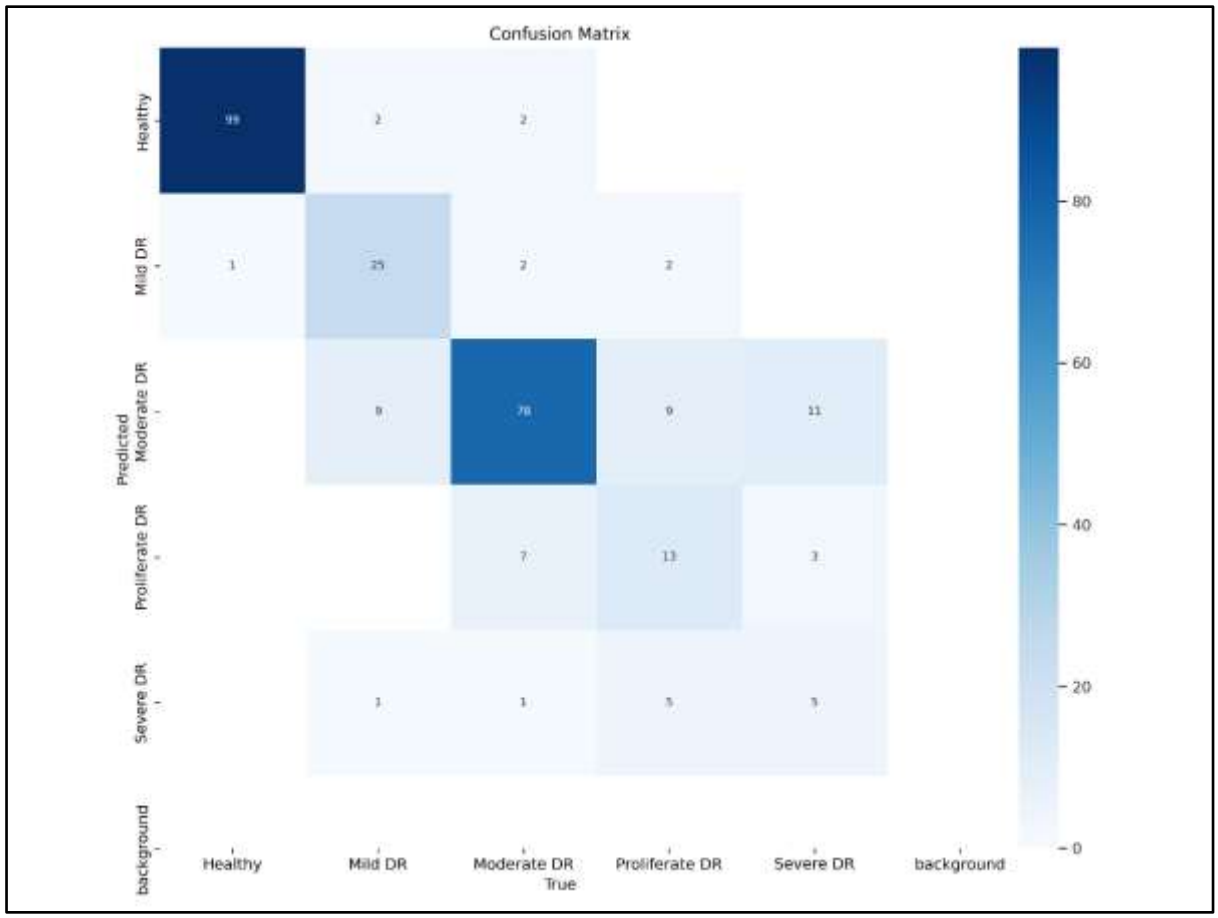

Figure 19: YOLO 11 classification model

This study adopts the YOLO 11 classification model to incorporate and classify five classes of DR, including healthy (not DR), mild DR, moderate DR, proliferative DR, and severe DR. The proposed model was tested for precision, F1 score, recall, accuracy, and AUC. This was evaluated with validation data. On the basis of validation of DR images, the Yolo 11 can identify various DR. For Healthy DR, Yolo11 achieved 98% precision, 99% recall, and 98.5% F1 score. For mild DR, this model has achieved 9.5% precision, 29.7% recall, and 14.4% F1 score. For moderate DR, this model has achieved 15.8% precision, 3.3% recall, and 5.5% F1 score. For proliferate DR, this model has achieved 20% precision, 6.9% recall, and 10.3% F1-score. For severe Dr., this model has achieved 0% precision, recall, and F-score. On the whole, the overall accuracy of the model was 41.8%. The model performance is quite lower than the previous model (VGG16 & VGG19).

The ROC curve of YOLO 11 indicates how the model can differentiate between the classes. The multi-class classification with YOLOv11 shows that this model has achieved an AUC of 0.991 for healthy DR, 0.9402 for mild DR, 0.9083 for moderate DR, 0.8992 for proliferative DR, and 0.9025 for severe DR. All these values were closer to 1, indicating that the classification performance was high in the model. The micro-AUC for YOLO 11 was 0.9589. The ROC curve for Yolo 11 is shown in the figure.

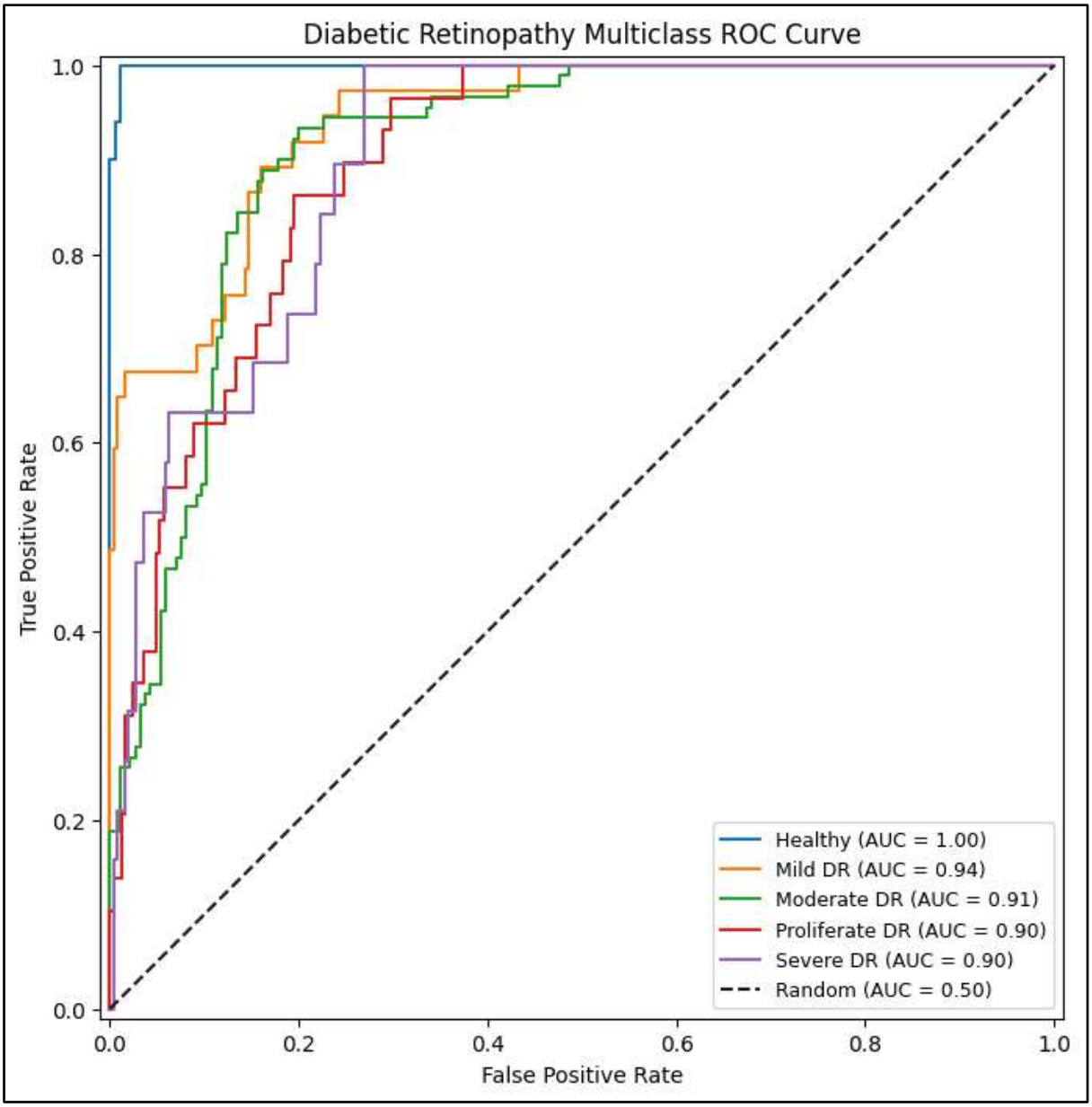

Figure 20: ROC curve of YOLO 11

4.5. Summary

The results of the model are presented and highlighted with the performance of the model. This offers insights about the evaluation of the model. This analysis forms the basis for drawing conclusions about the effectiveness of the model. The results will be discussed in the upcoming chapter in light of the body of existing literature.

CHAPTER-5

RESULTS AND DISCUSSION

5.1. Introduction

The result of the models executed in the previous chapter and its outcome is summarised in the Results and Discussion section. This section starts by presenting the models, and their accuracy and loss have been described. In order to promote a deeper understanding of the model, present the empirical findings and compare them to what is already known from the previous literature studies. The discussion section seeks to shed light on the importance and implications of our findings in the large framework.

5.2. Results

Using a Messidor image dataset, the proposed work seeks to demonstrate the most advanced DL models for early DR detection. Assessing the image dataset reveals that DL architectures have shown different behaviours. The comparative performance of DL models is illustrated in the table. The training loss and validation loss and their difference are shown in the table. Based on observation of these results, it was found that the architectures used in the study have a higher validation loss than the corresponding training loss. On the other hand, the training accuracy is substantially greater than that of validation accuracy. This suggests that the architectures fall into majority classification, which results in overfitting and poor generalisation due to data imbalance. Although imbalance exists, the best DL architecture for early DR detection is also determined by analysing important observations that are discussed in the subsequent points.

- In DL architectures, VGG 16 is considered to be a simple deep architecture. Adding more layers in VGG 16 can function more effectively with increasing depth. The findings of this study reveal that VGG 16 has achieved the highest validation accuracy in comparison to all other DL models. More depths and pre-initialisation of network weights are used to enforce implicit regularisation. The focal point of using these is to prevent gradient instability. It also employs small convolutional filters to add more nonlinearity to its structure. This also uses nontrivial receptive

fields to efficiently capture spatial context. VGG 16 has a lot of weight parameters; the model looks good and takes less time to infer the results for the Messidor image dataset. Although the model has good accuracy, it suffers from an overfitting problem due to its simple yet overly deep network stacking of 3*3 convolutional layers. Despite this, it is proved from the results that VGG 16 is simple yet a better model than other DL models.

- One of the other pre-trained architectures is VGG-19, which is considered to be more expensive to train the model. When compared to other DL models, VGG-19 is a variant of VGG architectures with 19 connected layers. Due to the increased number of layers, VGG-19 has relatively higher training and testing accuracy. As a result, this has continuously produced better results. However, the present study utilised the model that found that VGG-19 has relatively less training and validation accuracy than VGG-16. This was not a good performer and had a moderate connection with the Messidor image dataset. The reason is that performance problems like the vanishing gradient problem diminish the impact of the loss function on the activation function. This causes VGG 19 to perform poorly in the study.
- Yolo 11 is the new pretrained architectures that has been utilised to train the DR images. Comparing it with the conventional models, the performance was slightly less than that of VGG 16 and VGG 199. Consequently, this fails to produce better results for DR detection with the Messidor image dataset.

5.3. Discussion

This section shows a comprehensive comparison of the DR classification models using deep learning algorithms: VGG 16, VGG 19, and YOLO 11. With these algorithms, multiclass classification was made to find out the diabetic eye diseases automatically. According to this study, using publicly available data from Kaggle. Assessing the data has revealed that quantity and quality of the data is important rather than algorithm. Labelled images can provide more reliable, realistic, and useful results for computer-aided clinical application. In fact, the significance of each disease is the reason why this study uses the algorithms to classify against the multiclassification of DR. DR is caused by diabetes, and these conditions always result in significant and irreversible damage to visual acuity. In order to protect the patient from DR, the evaluation is performed with DR images. Evaluation of DR images with DL algorithms, and

the comparison is made on the basis of performance. The present study uses the same dataset and ensures a meaningful evaluation of performance.

The present study follows predefined DL procedures to find out DR images (Gharaibeh et al., 2021; Paradisa et al., 2020). A VGG 16 was trained on a small DR image dataset with five classes. The current work achieved an accuracy of 98% through data augmentation techniques. A dataset of 5 classes, including 726 healthy DR images, 726 moderate DR images, 726 mild DR images, 726 proliferate DR images, and 726 severe images, was used for training. Image segmentation and feature extractions were done. This was performed to achieve higher accuracy values for multi-class classification. Multiclass classification is an indicator to structure well for the model. The classification performance developed in our study (VGG16) was quite higher than that of other algorithms. The performance of the state of the art of AbdelMaksoud et al. (2022a) is much more comparable with our VGG 16 model. The precision difference between (AbdelMaksoud et al., 2022a) and VGG 16 is 25.5%. This proves that this correctly distinguish DR images using five categories. Also, there is no changes made in pre trained architecture even then it increases the model performance. There are one studies showing a least performance using different dataset for classification AbdelMaksoud et al., 2022a). The accuracy values of the dataset was 72% only. Similar to this, (Mateen et al., 2020) is much comparable with VGG 19 model and the difference identified was 13.06%. There are no previous studies available for Yolo11. Despite this, the study has achieved an accuracy of 80.7%.

VGG 19 and YOLO 11 perform quite poorly in terms of accuracy whereas VGG 16 has a competitive rate. Our model is accurate, computationally efficient, and extremely simple. This shows the role of conventional architecture in DR analysis, and it emphasises the importance of conventional over advanced architectural design for the detection and classification of DR images.

In summary, the comparative analysis shows that our VGG 16-based model has a higher classification accuracy for detecting DR images. The effectiveness of deep learning techniques in DR is validated by the performance hierarchy. According to the performance accuracy, VGG 16 is at the top and is followed by VGG 19 and YOLO 11. These findings have the potential to revolutionise the field of DR diagnosis and offer advanced health care by utilising cutting-edge technological solutions.

5.4. Summary

The present section summarises the key findings of the results obtained and their implications for deep learning experimentation. In order to progress the field of deep learning experiments, future work is outlined and discussed about the conclusion.

CHAPTER-6

CONCLUSION

6.1. Conclusion

This study proposed a multi-class classification based on deep learning models to evaluate DR via VGG-16, VGG-19, and Yolo11 for the Messidor image dataset. The evaluation of these models is based on accuracy, recall, and F1-score. According to the experimental results, the VGG-16 architecture and data augmentation produced a successful model that could correctly identify DR cases with a 97.42% training accuracy. In addition to this, the study results demonstrate the superior performance of VGG-16 through a comparative analysis with other algorithms like VGG-19 and Yolo 11. This model is effective both in terms of training parameters and classification accuracy. This model automatically extracts deep features from the input image data. This removed the need for complicated feature extraction processes. The reason is that the combination of feature extraction and image processing and manipulation has produced precise DR detection with VGG-16. Traditional models works(VGG 16) well over new model (Yolo 11). As a result, this finding highlights that VGG-16 is the most optimal architecture for the classification of DR images.

Considering all these outputs, the study concludes that VGG-16 has proven effective in detecting DR, paving the way for future developments in medical image analysis and enhancing diagnostic capabilities in the fight against diabetes.

6.2. Limitations of the study

The present study evaluated the architectures using a small-sized dataset. It is difficult to generalise our result because of this challenge. Thus, this is the primary limitation of the study. It will be crucial to assess the performance of the model with a large amount of data. Another drawback is that the proposed method has not been evaluated using the real dataset (fundus images of actual patients).

6.3. Scope for further research

Future work needs to be done to enhance image augmentation techniques such as convolutional autoencoders or GANs to increase the amount of training data in order to improve the classification accuracy of test data. Ensembles can be used to enhance the precision and

accuracy of the algorithm. In addition to this, the model must reduce overfit issues and poor generalisation issues in subsequent studies. Future research will try to collect more datasets on DR and test the performance using hybrid models for better classification algorithms. Future studies must deploy a real-time medical diagnostic system to acquire DR images from different eye hospitals and diagnostic centres. The eye diagnostics can greatly benefit from our method. This offers a high-precision, innovative solution for medical imaging, particularly for more precise DR diagnosis.