talenta PUBLISHER

DATA SCIENCE
Journal of Computing and Applied Informatics

Experimenting Diabetic Retinopathy Classification Using Retinal Images

***Muhammad Fermi Pasha*[1]*, Mark Dhruba Sikder*[1]*, Asif Rana*[1]*, Maya Silvi Lidya*[2]*, Ronsen Purba*[3]*, Rahmat Budiarto*[4]**

[1]*Malaysia School of Information Technology, Monash University, Malaysia*
[2]*Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Indonesia*
[3]*Information Technology Postgraduate Dept. STMIK-STIE Mikroskil Medan, Indonesia*
[4]*College of Computer Science and IT, Albaha University, Saudi Arabia*

Abstract. Along with many complications, diabetic patients have a high chance to suffer from critical level vision loss and in worst case permanent blindness due to Diabetic Retinopathy (DR). Detecting DR in the early stages is a challenge, since it has no visual indication of this disease in its preliminary stage, thus becomes an important task to accomplish in the health sector. Currently, there have been many proposed DR classifier models but there is a lot of room to improve in terms of efficiency and accuracy. Despite having strong computational power, current deep learning algorithm is not able to gain the trust of the medical experts in classifying DR. In this work, we investigate the possibility of classifying DR using deep learning with Convolutional Neural Network (CNN). We implement preprocessing combined with a widely-used image recognition model: InceptionV3, and very deep convolutional networks model: VGG16. Our preliminary experimental results show that InceptionV3 outperforms VGG16. InceptionV3 model achieves an average training accuracy of 73.5 % with a validation accuracy of 68.7%. VGG16 model achieves an average training accuracy of 66.4% with a validation accuracy of 63.13%. The highest training accuracy for InceptionV3 and VGG16 is 79% and 81.2%, respectively. Overall, we achieve an accuracy of 66.6% on 52 images from 3 different classes.

Keywords: Diabetic Retinopathy, Image Processing, Deep Learning, CNN

Received 23 December 2020 | Revised 24 January 2021 | Accepted 31 January 2021

1 Introduction

It is predicted that by the year 2035, the count of diabetic patient will rise to 562 million [1] with Diabetic Retinopathy being one of the many diseases associated with diabetics [2]. Along with many complications diabetic patients have a high chance to suffer from critical level vision loss and in worst case permanent blindness due to DR. Almost half of the global population is in the treat of becoming a victim of DR [3]. This catastrophic disease is claimed to be the major cause of blindness and with time the number of patients losing their eyesight is increasing rapidly [3]. Rate of DR can be reduced significantly if the disease can be addressed in the early stages; detecting DR in the early stages is a challenge to modern science. Since, it has no visual indication

*Corresponding author at: King Abdulaziz Rd, Al Bahah 65731, Arab Saudi

E-mail address: rahmat@bu.edu.sa

ISSN: 2580-6769 | e-ISSN: 2580-829X | DOI: https://doi.org/10.32734/jocai.v5.i1-5232
Journal Homepage: https://jocai.usu.ac.id

of this disease in its preliminary stage. The necessity of detecting (DR) in early stage becomes an important task to accomplish in the health sector. In the past DR was classified as a disease which cannot be cured [3]. Currently, there have been some proposed DR classifier models but there is a lot of room to improve in terms of efficiency and accuracy. Deep learning has shown some promising outcomes for image feature extraction and classification in the medical domain ([4]. Despite having strong computational power, current deep learning algorithm is yet not able to gain the trust of the medical experts in classifying DR [2].

In the field of medical image analysis, deep machine learning is playing a vital role [4]. Some research articles suggest that the detection of Diabetic Retinopathy can be made by applying the CNN (Convolutional Neural Network) combined with image fusion [5]. The use of the CNN model is recognized as a better alternative to the conventional methods used for visual learning [6].

The main goal of this research work was to show the importance of detecting DR in the primary stage and that machine learning can play a significant role in providing a solution to detecting the presence of DR in an early stage, thus providing the medical experts valuable time to cure the disease. This is just a pinnacle of how machine learning can classify DR.

The rest of the paper is structured as follows. Section 2 provides literature review. Section 3 discusses the proposed method, followed by the experimental set up, results and discussion in Section 4. Lastly, Section 5 gives conclusion.

2 Related Work

Quellec et al. [4] implemented a CNN classifier model to detect the lesion in the retina for DR prediction. The researcher use heat maps to eliminate the artifacts during feature extraction process. The model was implemented on the Kaggle dataset and obtained Area under the ROC Curve (AUC) value of 0.954 after training the model with 108,000 images. However, the validity of the model is questioned when we consider the fact that the model was created with a testing size of 89 images from DIARETDB1 dataset.

Pratt et al. [7] divide a complex CNN architecture into three sections and assigning each section an individual task of extracting features from the training image. This approach is efficient as training 3 separate CNNs to do the same task will require a lot of computational power and time. After the partition several of blocks are grouped together to result into several convolution blocks. Batch normalization with max pooling is being applied to each of the convolution block. CNN with a smaller training data size have a high tendency overfit the model. In order to prevent the model from overfitting, the convolution block is flattened to one-dimension. The final convolution layer is then associated with dropout dense layer and rectified linear unit in the end. They revealed that implementing real-time class weights in the CNN with back propagation can be considered as the solution. The authors reported that the trained model shows the model achieved an accuracy of 75%, specificity of 95% and sensitivity of 30%. Achieving a sensitivity of 30% and specificity of 95% suggests that the model proposed is biased towards classifying the output as not having DR.

Fine tuning was also used by Tajbaksh et al. [8] to build a CNN classifier model. In this paper the researchers went with a different approach to fine tuning rather than going with the common practices as in [7]. The common implementation of fine tuning CNN which only creates features from the training images and some fine tunes all of the layers of CNN [8]. AlexNet [9] was used as their preferred of CNN. The authors claimed that by incorporating the CNN model with handcrafted features will output an improved classifier model. The generated result shows that fine-tuned CNN had a slight better performance compared to the handcrafted features however fine tuning all the layers of CNN can improve the accuracy significantly. In this article, there was no specific method about the classification of DR. They use a single video and to source the images for training the classifier, then made us less confident in acknowledging their work as a better implementation to classify DR.

To obtain a good prediction, Pang et al. [14] use two fundus images of each patient. Hence, this required them to implement feature fusion for two eyes. While creating the classifier model for DR constructed a CNN model with two levels. InceptionV3 which is an implementation of CNN architecture is used for feature extraction and VGG model. Even though, models like CNN and deep Multiple instance learning (MIL) have been used in the model while building the classifier model for DR; the authors did not mention any specification of their model which we could take note of. In addition, they did not mention the data set being used neither report on any form of pre-processing steps before implementing the model.

Researchers in [10] also use AlexNet and come up with four different CNN models rather than sticking with a single model.

In image fusion the combination of modalities used to classify a model depends on the presence of noise in the training data set. There is direct relationship between the performance of classifier model and the quality of the data set in the model. The performance of a modality in medical domain depends on the structure of the organ and the tissues [11]. The authors gave an overview of the application of image fusion in the medical domain and a brief idea about how image fusion can be applied in the medical domain. Since it was a review article there were no specific proposed methodologies for applying image fusion. Table 1 summarizes the relevant work on DR detection.

Table 1. Summary of relevant works on DR detection

Ref. #	Method	Focus	Pros.	Cons.
[3]	VGG16	DR	Deal with small size dataset	No explanation on its preprocessing
[4]	CNN	Lesion in the retina	impressive Area under the ROC Curve (AUC) of 0.954	Weak validity, size of 89 images from DIARETDB1 dataset
[7]	CNN with preprocessing	DR	Overcome overfitting	Having bias results
[8]	Fine tuning CNN	DR	High accuracy	Use a single video (less confident)
[10], [16]	Alexnet	DR	High accuracy	n/a.
[15]	InceptionV3 + VGG	Fundus images	High accuracy	No dataset information

3 Methodology

The methodology to construct a perfect diabetic retinopathy classifier is considered to be one of the most critical aspects in current times. Several methodologies have been proposed during the past few years, yet no such methodology has proven to reach accuracy over 82%. In this work, we propose a promising approach that would produce only a slight less accuracy using small dataset as the first stage to achieving a higher accuracy model with larger dataset. We illustrate the ideology of pre-processing the images similar to the other proposed ideologies. However, the approach to making a prediction over a certain retinal image is slightly different. In this work, we first implement the preprocessing of the retinal images so that we can apply the learning model for building the classifier model using deep learning. Then we focus on the extraction of features from the retinal images by abiding some of the key features noted relating to extraction and learning of exudates from the training images followed by the principals of the features used in classification. In the process of implementation of the proposed classifier model we consider the problem statements identified by the researchers and provide a solution to those problems.

The following subsections illustrate an overview of the implementation techniques and ideologies used to classify a diabetic retinopathy retinal image, how the issue of the data was handled and what is the outcome.

3.1 Data Split

The training and testing data provided was split before the model was trained. It is an industry norm to split the data set into a 70/30 split. 30% split of the data is kept for validating the data set after the model has been built. The rest 70% of the data is used for training the model.

While training the model, the training dataset was further split to 70/30 split. 30% of the training dataset was used for validating the learning process of the model. Exactly after this point is where we assured that data augmentation was a key aspect to achieve higher accuracy.

3.2 Image Pre-processing and Data Augmentation

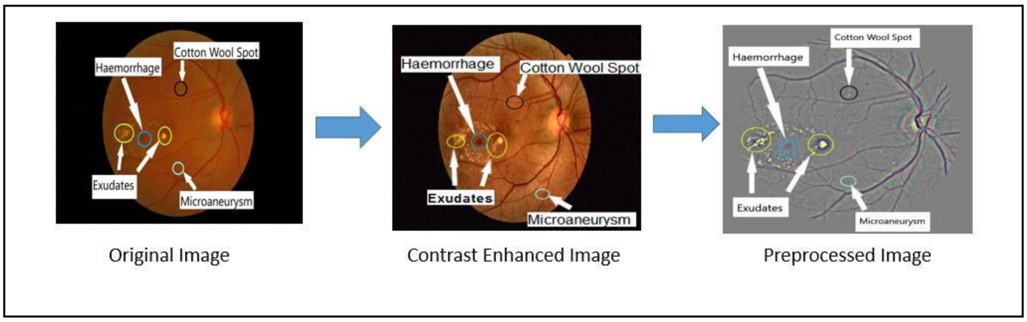

Figure 1. Pre-processing

After the split was made, while analyzing the dataset, the first thing we noticed is that the images provided does not have equal distribution for the classes. This means that we are required to provide a solution to this issue or else the model will return bias results. Second, the total number of images in our training set was only 650. During the stage of crating the literature review it was mentioned in several research papers that the size of the dataset for machine learning play a vital role in creating a reliable machine learning model.

To overcome the first obstacle of unbalanced images for training class we implemented batch normalization which was carried out by taking equal number of images for all of the classes in the training set for the classifier. This step for preprocessing will ensure that there is less chance for the classifier model to be biased towards a certain class.

In the next stage we figured out a solution for the small dataset size. Since, it would be very challenging to a significant model for classifying with a small dataset we had to come up with a solution which will deal with the constraint of small dataset size. The selected solution was to implement dataset augmentation. Augmenting a dataset is a common technique to deal with a small dataset. Many research works have provided some forms of dataset augmentation. During the implementation we have carried out several augmentation techniques which include rotation of the images into several degrees; 180 and 270 degrees.

The images were divided into three categories, where each category represents different variations of diabetic retinopathy. So, the most important stage before creating the classifier is to traverse the images through a series of some preprocessing techniques. The raw images provided were of very poor quality and so, configuring out a perfect processing technique which can fix the issue was strenuous. Images varied in resolutions, illuminations and contrast. A normalizing technique was performed which lead all the images to have the same resolution. Here, the images were resized to 512 pixels and cropped based on the radius of the retinal image which allowed the images to have the same radius of the field of view. In order to fix the illumination and contrast, we applied a robust contrast enhancement technique known as the histogram equalization which enhances the contrast of the image (middle image in Figure 1). This enhanced valuable features such as microaneurysms, exudates and thick blood vessels which are initiations of diabetic retinopathy. Furthermore, using the enhanced contrast resized image we applied Gaussian smoothing kernel, in contrast to standard deviation metrics in order to estimate the background illumination. This technique transforms the image in a way such that the obvious features such as the color of the retina were made translucent. Hence, only the important features such as the blood vessels, microaneurysms and the exudates were immensely visible. In contrary, we just made it easier for the model to learn as it is only going to learn the factors we want the model to learn. Lastly, using the contour technique the outer circular border was removed. The rightmost picture in Figure 1 displays the final pre-processed image of three different classes.

3.3 CNN Architecture

Several architectures have already been tested for DR classification. However, the difference lies in how the pre-trained model is fine-tuned which best fits the data specified. The most commonly used CNN so far is the VGG16 architecture which was originally designed to classify 1000 different classes. It consists of thirteen convolution layers coupled with ReLu. Hence, we used VGG16 as our prior model and alongside we used InceptionV3 which has a much deeper architecture than VGG16. They use parallel processing blocks in the layers to improve its performance. The weights however are pre-set as ImageNet and thus, for no complications in the model we used the weights of ImageNet.

The learning rate for Inception however was very critical to fix. Hence, we took the advantageous functionality of Keras, which allows us to use the Stochastic Gradient Descent (SGD) optimizer which changes the learning rate based on what it is learning during each epoch. To train the network we used a multinomial logistic loss function and implemented it in the same layer as the softmax classifier. The combination of the softmax layer and the log-loss is useful for numerical stability.

InceptionV3 known as has ability to perform better with images of size 299. Hence, when loading the images into the model we resize the image to 299. The model was tested for at most 60 epochs and a graph was plotted based on the training and validation accuracy for every epoch to analyze the training phase of the model. In order to reduce overfitting, dropout layers were added after each fully connected layer and lastly the functionality for the model to predict only 3 classes was initialized after the fully connected layer. InceptionV3 constitutes of 2 dense layers of 1024 filter size including two dropout layers of 0.5. Moreover, while tuning the model, some of the layers were frozen as those layers learn the basic features of an image such as shapes and objects. Hence, there was no specific requirement in tuning the first few layers of the model.

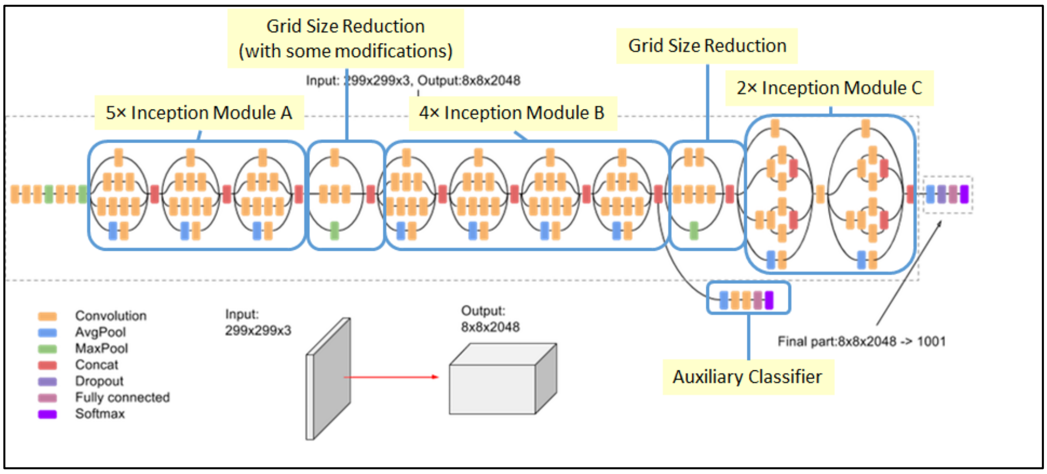

Figure 2. InceptionV3 Architecture [12]

VGG16 however does not have such issues in tuning as the internal architecture of the model is not as deep as InceptionV3. So, it is less likely to learn the complete deep features of the retinal image where we have to detect several features such as microaneurysms, hemorrhages and internal blood clots. Similar to InceptionV3, we provided same weights to VGG16 as initially this model was also trained using ImageNet weights. In contrary, one flatten layer was used which generalizes the input neurons into one specific shape. Followingly, two dense layers were used with filter size 4096 alongside with softmax function as the activation function and two dropout layers of 0.5 and 0.7 respectively; 20% and 70% of the neurons which have no contribution are dropped after training. Furthermore, during compilation, once again SGD optimizer was used as it is an incremental gradient descent which uses an iterative approach for optimizing a differentiate objective function. Individual optimizers tend to have different learning rate computations. However, SGD seems to outperform the other well-known optimizers such as ADAM and RMSPROP.

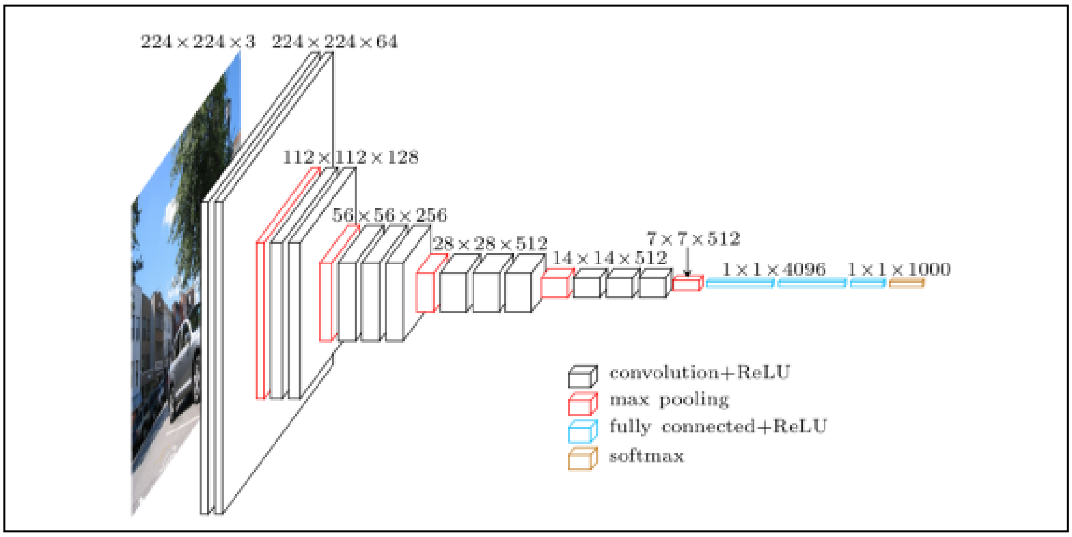

Figure 3. VGG16 Architecture [13].

Insufficient amount of data plays a vital role in the training phase of the model. No matter how detailed the images are or how many classes we have, due to a compact dataset the model cannot learn as many features as possible and hence there is an intensive requirement for data augmentation. Due to such compact data we decided to use 20 batches of images for each epoch during training after augmenting the data. In a sense, the training batch size depends on the number of total training images. Therefore, the training batch size should be a factor of the total training images so that the batches are evenly divided among every epoch. Similar to this concept the validation batch size and the test batch size was set using the total number of images [14].

As two models are being used for the classification of DR, we have two sets of predictions; one set for each model. But the final prediction has to base on one single set of predictions and in order to get one single set of predictions we introduced a fusion approach where we fused two predicted values and generated one single prediction for each class. This concept was taken from a technique known as fusion technique. Overall, we based our test accuracy on the final fused predicted values. This fusion was done via the concept of averaging technique which takes the average of the two sets of predictions for every image. However, this method does have limitations as well. One limitation can be the fact that if one classifier generates a very poor prediction and the other predicts a quite well prediction, then the overall prediction can turn out to be low and eventually lead to lower accuracy. However, in our case we do not aim for overall accuracy but aim for how much well the model predicts what class the DR image belongs to.

3.4 Validation

As mentioned above there are several standard ways for building a DR model. In terms of our case, the selection for the validation process for building the model was mainly centralized on the validation processes which can be visualized, which includes training vs validation graph, training vs validation loss and confusion matrix. All the mentioned methods for validation helped us to make sure that after each change is made to the model, the model was able to detect the features of the DR images appropriately with minimal loss.

4 Experimental Setup, Results and Discussion

4.1 Experimental Setup

The training was done on a virtual supercomputer provided by Information Technology Services of Monash University, constituting of 32GB RAM including a graphics processor of Intel(R) Xeon(R) of 2.20GHz.

4.2 Experimental Results and discussion

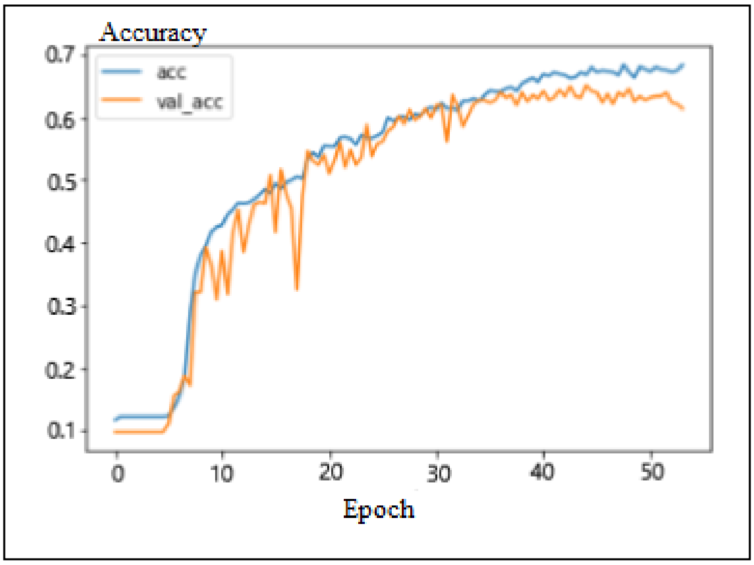

Figure 4. Training accuracy Vs. Validation accuracy (InceptionV3)

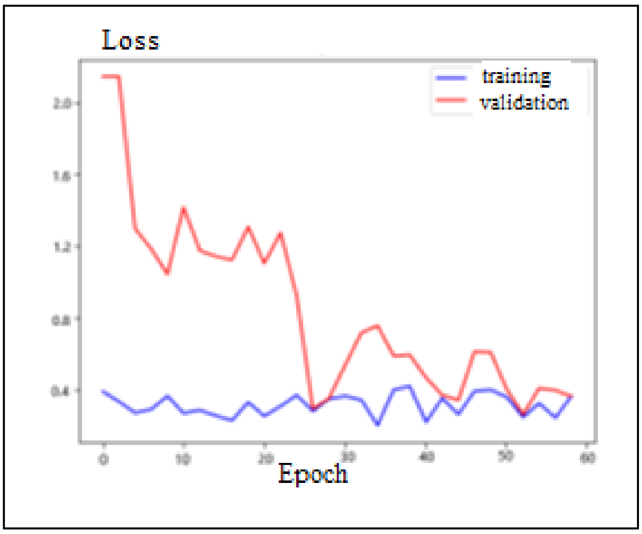

Figure 5. Training loss Vs. Validation loss (InceptionV3)

Among the two models we implemented, InceptionV3 which is under the domain of CNN architecture seemed to perform better. After multiple tuning and testing we achieved an average training accuracy of 73.5 % with a validation accuracy of 68.7% for InceptionV3 as seen in Figure 4. However, the highest we have achieved for training is 79%. In the case of VGG16 model, we were able to achieve an average training accuracy of 66.4% and validation accuracy of 63.13% as depicted in Figure 6. Highest ever reached training accuracy for VGG16 was 81.2%. By observing the training accuracy graph in Figure 5 very closely we found out that there was minor sign of overfitting for InceptionV3 which is an achievement on its own since with a small data set there was a high chance for the model to overfit. In addition, when running the model several times for further clarification about the reliability of the model we have noticed that the variance of the results for the training accuracy was less than 2%. Another metric that support this the performance analysis of the model is validation loss. Using the validation loss metric allow us to interpret how well the model performs during validation stage. From our preliminary experiment, the validation loss was less than 1.25% for InceptionV3 and on the other hand, the variance for validation accuracy for VGG16 is about 2.7% as shown in Figure 5 and 7. From the training and validation loss learning curve of VGG16 in Figure 7, we can observe that the model is a bit overfitting. However, reducing this overfit is very strenuous in a sense that finetuning the model in depth is challenging.

Considering the two validation accuracies, we can comment that due to the deeper architecture of InceptionV3, the model outperformed VGG16 and further tweaking can lead to higher accuracy. Overall after fusing the model's predictions, we achieved an accuracy of 66.6% in avarage for all 3 different classes on 52 images. The moderate DR class has the highest (100%) accuracy whereas mild and severe DR have the accuracy of 54% and 66.6% respectively.

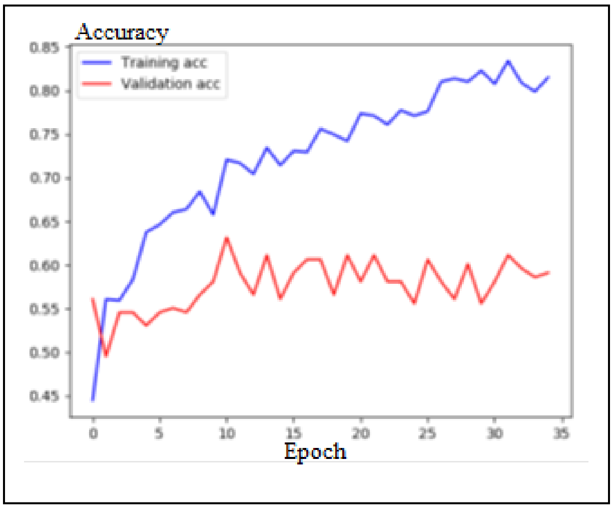

Figure 6. Training accuracy Vs. Validation accuracy (VGG16)

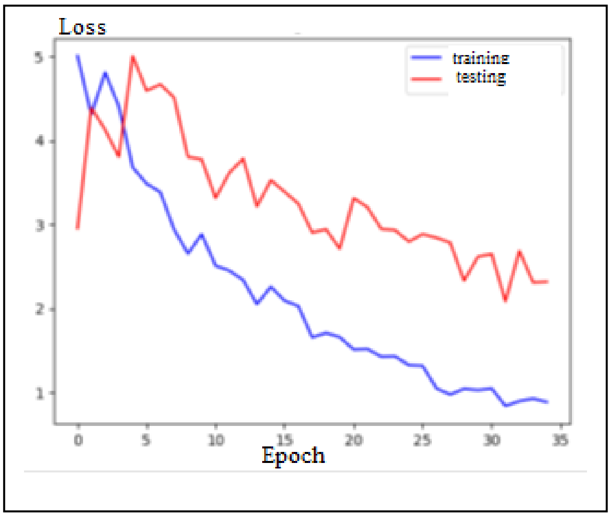

Figure 7. Training loss Vs. Validation loss (VGG16)

By further analyzing Figure 8 and based on the baseline provided in the Kaggle dataset, we can conclude that in overall, the model performed well as expected. Even though some of the predictions were incorrect, we can infer that the model however did not tend to produce uncanny results. Observing the model, we can visualize that most correctly predicted values are for Mild DR and alternatively we observed that the model predicted around 7 images which are supposed to be Moderate but predicted was Mild DR. This might be the issue that the difference among Moderate DR and Mild DR is very difficult to determine. Hence, the model performed poorly in such cases. Moreover, deeper analysis pointed out the fact that in some cases the model predicted Severe DR when it is supposed to predict Mild DR. This might be the case of data cleaning. Thus, we need medical doctor's opinion to clean the data appropriately.

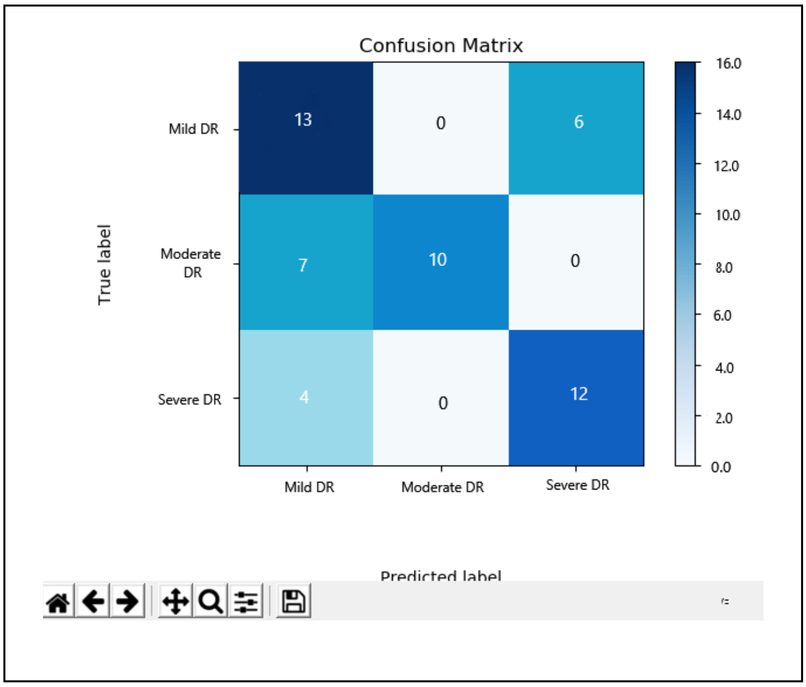

Figure 8. Confusion Matrix.

In current times DR can be classified into many different stages. However, we decided to categorize our data into 3 parts as our data is very limited and the classification of several classes can ultimately lead us to lower accuracy as the model then has less learning data compared to validation. In contrast, the dataset was densely imbalanced which caused a problem to split the data. This prevented us in achieving a much higher accuracy. The judgment for such transfer was made according to research finding in [14].

There is no doubt that deep learning has the capability to classify DR. In the medical domain it is very common to get a skewed dataset for train the classifier; this is accompanied with variable resolution, brightness and contrast level. Common filtering approaches include normalization of the dataset, weighted sampling, eliminating skewed classifier.

5 Conclusion

After going through literature review and considering several modalities it can be concluded that using a CNN classifier with fine tuning and the incorporation of image fusion with the classifier is more like to output a better classifier model for DR. Some new features are added to the model with considering the state of the art in the first place. Overall, we achieve an accuracy of 66.6%

on 52 images from 3 different classes. The achievement was not so encouraging due to imbalance dataset.

In the future we aim to improve the overfitting problem of VGG16 and moreover, get a little bit of more knowledge about the exact signs of DR and using a much bigger and clean dataset, try to make the model more accurate in its predictions. We also plan to break the images into patches and then input it to the classifier where the classifier can learn the depth of the image patch by patch and afterwards join the image and make an overall prediction.

REFERENCES

[1] L. Guariguata, D. R. Whiting, I. Hambleton, J. Beagley, U. Linnenkamp, and J. E. Shaw, "Global estimates of diabetes prevalence for 2013 and projections for 2035," *Diabetes Res. Clin. Pract.*, vol. 103, no. 2, pp. 137–149, 2014.

[2] M. R. K. Mookiah, U. R. Acharya, C. K. Chua, C. M. Lim, E. Y. K. Ng, and A. Laude, "Computer-aided diagnosis of diabetic retinopathy: A review," *Comput. Biol. Med.*, vol. 43, no. 12, pp. 2136–2155, 2013.

[3] M. A. Bravo and P. A. Arbeláez, "Automatic diabetic retinopathy classification," in *13th International Conference on Medical Information Processing and Analysis*, 2017, vol. 10572, p. 105721E.

[4] G. Quellec, K. Charrière, Y. Boudi, B. Cochener, and M. Lamard, "Deep image mining for diabetic retinopathy screening," *Med. Image Anal.*, vol. 39, pp. 178–193, 2017.

[5] Y. Liu, Y. Guo, T. Georgiou, and M. S. Lew, "Fusion that matters: convolutional fusion networks for visual recognition," *Multimed. Tools Appl.*, vol. 77, no. 22, pp. 29407–29434, 2018.

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[7] H. Pratt, F. Coenen, D. M. Broadbent, S. P. Harding, and Y. Zheng, "Convolutional neural networks for diabetic retinopathy," *Procedia Comput. Sci.*, vol. 90, pp. 200–205, 2016.

[8] N. Tajbakhsh *et al.*, "Convolutional neural networks for medical image analysis: Full training or fine tuning?," *IEEE Trans. Med. Imaging*, vol. 35, no. 5, pp. 1299–1312, 2016.

[9] H. Gao, "A walk-through of AlexNet," *Mediu. Corp., Aug*, vol. 7, 2017.

[10] A. P. James and B. V Dasarathy, "Medical image fusion: A survey of the state of the art," *Inf. fusion*, vol. 19, pp. 4–19, 2014.

[11] R. F. Mansour, "Deep-learning-based automatic computer-aided diagnosis system for diabetic retinopathy," *Biomed. Eng. Lett.*, vol. 8, no. 1, pp. 41–57, 2018.

[12] V. Kurama, "A Review of Popular Deep Learning Architectures: resNet, InceptionV3, and SqueezeNet," *Consult. August*, vol. 30, 2020.

[13] D. Frossard, VGG in Tensor Flow, https://www.cs.toronto.edu/~frossard/post/vgg16/, 17 June 2016, retrieved December 17, 2020

[14] M. Mateen, J. Wen, M. Hassan, N. Nasrullah, S. Sun, and S. Hayat, "Automatic detection of diabetic retinopathy: a review on datasets, methods and evaluation metrics," *IEEE Access*, vol. 8, pp. 48784–48811, 2020.

[15] H. Pang, C. Luo, and C. Wang, "Improvement of the application of diabetic retinopathy detection model," *Wirel. Pers. Commun.*, vol. 103, no. 1, pp. 611–624, 2018.

[16] L. Zhou, Y. Zhao, J. Yang, Q. Yu, and X. Xu, "Deep multiple instance learning for automatic detection of diabetic retinopathy in retinal images," *IET Image Process.*, vol. 12, no. 4, pp. 563–571, 2017.