Classification of diabetic retinopathy using transfer learning on pre-trained inceptionV3 model

MSc Research Project
Data Analytics

Saigirish Palavarapu
Student ID: X17170401

School of Computing
National College of Ireland

Supervisor: Theo Mendonca

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Saigirish Palavarapu
Student ID:	X17170401
Programme:	Data Analytics
Year:	2018
Module:	MSc Research Project
Supervisor:	Theo Mendonca
Submission Due Date:	20/12/2018
Project Title:	Classification of diabetic retinopathy using transfer learning on pre-trained inceptionV3 model
Word Count:	9450
Page Count:	24

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	12th December 2019

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Classification of diabetic retinopathy using transfer learning on pre-trained inceptionV3 model

Saigirish Palavarapu
X17170401

Abstract

Diabetic Retinopathy shortly called as DR is the leading disease for chronic diabetes person. It can lead to blindness in its final stages. According to world health organization reports it is mainly seen in working-age adults. Early detection of this condition is very important for prognosis as well as diagnosis. Due to the damage caused to the retina blood vessels by high blood glucose levels, micro-aneurysms, hard exudates, and neovascularization could occupy the retina area. In this research, it is demonstrated that the categorization of 5 stages of DR is done using convolution neural networks pre-trained model called Inception V3. With few modifications in its final stage of classification inception V3 is used as a training model in this research. This classification has been done based on the number of features extracted by the pre-trained model and these features are optimized using RMSprop and Adam optimization techniques.

Using of pre-trained convolution neural networks in building machine learning models is proved to be one of the best ways to detect many of the biomedical image analysis. Greyscale and ben graham's which are state-of-the-art techniques for pre-processing of images where used in this research to provide the best results by the algorithm. Using Transfer learning technology on pre-trained Inception v3 model from publicly available Kaggle data given an optimistic results. The results obtained using RMSprop, ADAM optimizers is, accuracy of 60.3%, 45%, sensitivity of 57%, 50% and specificity of 89%, 85% . This research shows that RMSprop optimization works better in detecting DR stages.

1 Introduction

Human's eyes are one of the most important and sensitive parts in their body and it has many infections than can be cured easily. Few of them can be easily solvable but few take a considerable amount of time, energy and money to take them back to original and few of them . One of the types of eye disease that takes huge care to be solved is diabetic retinopathy (DR), which can lead to vision loss in its final stage of infection. This is caused due to the less care taken by the people who are suffering from diabetics. DR is a very serious illness that does not have any symptoms in its initial stages to the person. Because of this reason, detection of DR for the doctor became a challenging task. DR slowly affects the eye sight of the person and then it can cause a serious threat to the eye. One of the serious stage is leakage of blood in the retina, if the person (DR positive) does not take care or recognize the issue and this situation continues then, it may lead to vision loss to the person in its final stage. It is very important to know the stages of DR for the evaluation of the illness. Particularly advanced stage detection called diabetes mellitus (DM) detection and Severe NPDR also to evaluate the other stages of retinopathy.

Every ophthalmologist cannot detect this infection and its stages, it required a huge amount of observation, time, equipment and also with a knowledge on both diabetatology and ophthalmology. This increases the cost of detection and even treatment for sometimes. DR has 5 stages they are No DR, mild DR, NPDR (Non-proliferative DR), sever NPDR and PDR (Proliferative DR). These five stages can be differentiated based on the microvascular lesions found onto the retina. Symptoms like microaneurysms (MAs), haemorrhages (HMs) and exudates (EXs) is noticed in retina, then it is called NPDR which is an early stage of DR. Along with these symptoms, if the abnormal increase of blood vessels and lesions is detected, then it is called PDR which is an advanced stage of DR. This can be explained more clearly in coming sections 3 with images. Due to these different reasons like time, cost in detection, expertise in the relevant areas made DR one of the major issue needs to be focused.It is one of the major issues where people need to be replaced with technology to some extent and this leads us to the problem statement.

1.1 Problem Statement

The goal is to increase the perfection in detecting DR with its stages. Accuracy in detecting the issue will play a major role in treating the infection. Sensitivity and specify is also considered as some of the important measures to be considered in classifying its stages Abràmoff et al. (2016). According to the reports of WHO, nearly 420 million people across the world are suffering from DR and most of them are 25-79 years old. The observed DR in people is in advanced stage which is called diabetes mellitus in medical terminology. The major reason behind the huge increase in the number of infected people is due to the lack of resources and sometimes people cannot go for frequent check-ups and also its impossible of many reasons for a part of people Afrin and Shill (2019). The reports also stated that from the past 30 years, DR has been recognized and occurred twice than before and it tends to be increasing day by day. DR is mostly observed in the region of Asian countries Garside et al. (2019).

This gap in treating the DR positive people can be fulfilled by the use of artificial intelligence and machine learning technology. Computer vision is one of the extensively used technology from the past few years by the researchers as well as industry to detect the objects Garside et al. (2019). In this research machine learning model introduced and developed by google called inception V3 is used to detect the infected areas of the retina and find weather the person is affected by DR. If algorithm detected as positive then to find out the stage of infection.

1.2 Research Question

By considering the above-stated issues following question is formulated

"Can the use of transfer learning method and hyperparameter tuning method with inception V3 architecture will improve sensitivity and specificity over the state-of-the-art accuracy in detection of DR?"

1.3 Scope of study

The specified research question is answered using publicly available images taken from two different sources[1]. These datasets consist of images of both effected and not affected retinas. The size of the data set is around 70,000 images. The images consist of microaneurysms, hemorrhages, exudates, cotton wool spots which are the important features to be extracted while classifying the DR and its stages.

2 Related Work

Machine learning and deep learning are some of the fields where the researchers are focusing on the past few years Gargeya and Leng (2017). Researchers have been using them to solve issues regarding the health care industry, sales, natural language processing and many more. At this point, image reorganization and classification have started showing its impact on the business world DLMIA et al. (2018). This improvement in technology has been helping the common person in some or the other way, for instance, recent technological research on the reorganization of malaria has to help many people around the world to detect the illness very fast and with a high degree of accuracy Joseph and Geetha (2019). To maintain the accuracy, a powerful machine learning model should be employed to do the work for us. Kipping this in mind there are many pre-trained machine learning models available in the present technological world..

Many of the machine learning models have been developed from the past few years to solve medical issues using different technologies. This section explains the past few works done by the researchers and technologies they have used to achieve their goals. This will also consists of comparisons between the results of their models and tries to expose some of the reasons behind the differences between their results.

2.1 Tensorflow

Tensorflow is developed by developers of google where it is said to be one of the important libraries to handle heterogeneous environments and large scale data Abadi et al. (2016). The TensorFlow works on the dataflow graphs to show the computations, operations, and flow of the data. The below image will explain the architecture of the TensorFlow.

The data flow graph is considered to be one of the important features of TensorFlow and also it plays a major role in understanding of how the library works. Wongsuphasawat et al. (2018) has elaborately explained the working of library with the graph visualizer. In TensorFlow the graph visualizes in its machine intelligence platform. models developed are then analyzed to know the working of model and it also have an advantage that, user can analyze the nested structure of the developed model. At the same time Joseph and Geetha (2019) paper has solver the puzzle of recognition of facial expressions using the TensorFlow library with good results. They used fuzzy classification in their research and initially the images are enhanced using wavelet transform before applying model to the images. The researchers have also used eye map and mouth map algorithms to find the exact outline of the face or the facial geometry. Based on that outline, expressions are classified by the algorithm. This researchers have also found out that the metrics like accuracy will increase or decrease with the increase of the number of iterations or epochs.

[1]https://www.kaggle.com/c/aptos2019-blindness-detection/data
https://www.kaggle.com/c/diabetic-retinopathy-detection/data

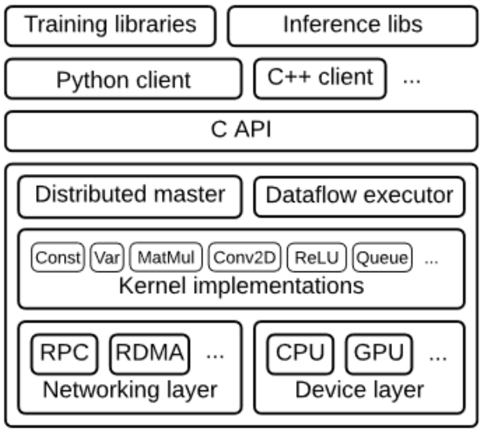

Figure 1: Tensorflow dataflow graph

Using TensorFlow as back end and inception v3 algorithm as a model in classification Xia et al. (2017), has classified flowers of two different data-sets. They have used oxford-17 and oxford-102 data for classification. This model has given greater results in recognizing the flowers with their types and got an accuracy of 98% and 95% for two different data-sets. They have used the transfer learning method to train the final layers of new categories of the data instead of training the entire model. On the other side Gogul and Kumar (2017) has done research on the same topic but on the different data-sets. The model used by these researchers is Inception v3 but the results are lower when compared with the Xia et al. (2017) paper. Gogul and Kumar (2017) has got only 82% accuracy in classification. Which is nearly 15% lesser than that of Xia et al. (2017). This is because of the size of the dataset or the features extracted by the algorithm or different pre-processing techniques.

2.2 Keras on Tensorflow

Even though TensorFlow provides many of the machine learning development opportunities, it is very hard to build a machine learning model because of its complexity in coding. To solve the issue of complexity for Tensorflow, Keras was developed. It's a library that works on the top of the TensorFlow which makes the coding easy to develop and also to understand the developed models. Keras is a library that is compatible with not only TensorFlow but also many of the machine learning platforms like Theano and CNTK. Interesting research was done by Yin et al. (2018) using Keras on TensorFlow. This research main aim is to collect the eye tracked data and this is done using a device called Tobii X60.

With the use of data collected by the device the researchers can able to find out whether the user's eye is looking at the Google News or the News map. The information collected by this device was GazePointX and GazePointY with also some other fields. Researchers used Mongo DB to store the data of the GazePoints in the form of a 2D array and also this array is in the size of screen resolution. They have used the gradient descent optimizations to optimize the features and got the best results. The below image 2 will show the difference between the two types of news.

Researchers like (Wagner et al. (2019)) has developed a model for classifying the type of forest based on the images. This researchers have used u-net architecture in

Figure 2: left is the Google news and the right is News map

classifying images and also used Keras on TensorFlow to build the model. Not only image recognization but also many other problems of classification has also been solved using this technology. Researchers like Garzón-Alfonso and Rodríguez-Martínez (2018) have taken twitter data to build a model on twitter health surveillance. Garzón-Alfonso and Rodríguez-Martínez (2018) have classified 56013 tweets into two categories based on the features using this library.

2.3 Image recognization using neural networks

For image classification and the segmentation of DR stages Abràmoff et al. (2016) has introduced an enhanced deep learning model. The researchers have used the fundus images to do the process and achieved good results of 96.8% of sensitivity and 87% specificity. But in this research, the size of the data is very small and according to researchers like (Meng et al. (2018)), data set size will also show influence on the results. The same research of classifying images of DR has been done by (Gargeya and Leng (2017)). Where they have classified into only two categories and the model they have used to classify was a noraml data-driven deep learning model. The researchers have used color fundus images, which also available in the public domain and free to construct models on them. The results they have got were good, but the number of categories are low. This is because of size of the data that is used for the analysis.

When using the CNN model in image classification, images need to be pre-processed to obtain the best results from the model. In this one of the pre-processing method was to mask the optical disk. Where it helps the algorithm to differentiate between the optical disk and a few other features. This step is necessary because sometimes they will be same in color. This is followed by most of the state of the art researchers before applying CNN models. A paper was published by Adem (2018) based on the same procedure to detect the exudate automatically by the algorithm. The researcher has used the optical disk (OD) masking to classify the DR using the CNN model. The CNN model used to train and classify the images whether the person is infected or not. The model performed well enough in binary classification. The reason behind the success of this model is its architecture, where it consists of 3 convolution layers and also a layer of max pooling. So, with this paper, it is clear that even the normal CNN architecture can perform well enough to detect DR in binary format.

The researcher (Xie et al. (2019)) has presented a paper on classifying DR images. The researcher has used randomly wired neural networks to do the process. In this research, authors have encapsulated the total network generation process using a stochastic network generator. The results are very good when comparing with the other processes to solve the same issue. The researchers have achieved 79% accuracy in classifying the images under the similar computational cost of ResNet-50. The results when compared with the Szegedy et al. (2016) paper, it is evident that the Xie's research process performed well in detecting the DR.

2.4 Inception V3 algorithm

Google developers has introduced a pre-trained model called inception Net which has 4 versions. Chowdhury et al. (2019) has used inception v3 pre-trained model on the publicly available data from kaggle. The researcher has trained only the last few layers of the model which is used for the classification task. The accuracy achieved by the researcher is 78.1% in binary classification. This research has got better results than most of the state of the art techniques, which are trained on the ImageNet dataset. While at the other hand Masood et al. (2017) have performed a classification of different stages of detecting DR. The researcher has also used the traditional inception v3 algorithm with the training of final layers of classification. There is also a pre-processing stage that has been performed by the researcher on the data which is taken from kaggle. The results are very less when compared with Chowdhury et al. (2019) paper. The results got for the researcher Sarfaraz were only 48.2% where this massive change in the results is because of the number of stages in the classification and mostly because of the pre-processing stage performed by the researchers.

Takahashi et al. (2017) have proposed a paper on detecting the stages of DR using a machine learning algorithm called modified Google Net architecture. The data used for this research is taken in a different maner. Researchers have taken four images of each person's eyes at non mydriatic 45 field color fundus photographs every year from 2011 to 2015 in a medical university. In this research, another training tool has also been used that is PABAK (Prevalence and bias-adjusted kappa) for 5% of training and the remaining images are trained using pre-trained google net architecture. The accuracy got by the researchers for PABAK was 81%. The author also mentioned that it is comparatively less with the other state-of-the-art PABAK models but the training in their research is done in two-fold architecture, this is majorly useful in training single-field fundus images and also classification of DR stages. The alone google net architecture ofChowdhury et al. (2019) research was got less result than that of modified google Inception V3 architecture.This research results in thinking of new dimensions of the google net architecture.

A research is proposed by Lam, Yi, Guo and Lindsey (2018) to detect rare eye disease especially when there are fewer images for training. The researchers of this paper have taken a very small number of images for their research from the public domain. In this paper, to generate the probability map across all the images, researchers have used the sliding- window method. A pixel-wise detection has been followed by the authors to classify the images based on the feature which are detected by the algorithm. The CNN model like AlexNet, VGG16, GoogleNet, ResNet, Inception V3 have been followed and an interesting comparison of accuracy has been done between them in this paper. Among all the CNN models Inception V3 has given high results of accuracy of 96% and the least accuracy was 76% which is given by Alxnet. In this research, the data set considered was very low when compared with the other researchers and the results may vary when it is given with a high amount of data.

2.5 RMSprop optimization

A CNN model to detect diabetic retinopathy with its stage has been done by Xu et al. (2017). In this paper it is mentioned that to detect the stages of DR, a CNN model is developed which takes 224*224*3 sizes of data as an input. These convolution layers are then connected to a softmax layer for classification. RMSprop optimization has been performed in this research to optimize the number of features detected by the CNN model. The results obtained by the research are well enough to detect the stages of DR but not as accurate as of state-of-the-art researches. Thus the author also mentioned that there are many changes need to be taken for this method to detect with a higher degree of accuracy.

A research paper was published by Ardiyanto et al. (2017) in detecting diabetic retinopathy using a pre-trained Deep-DR-Net CNN model. In this research, the authors have classified the images of fundus into three categories. The authors have also tried different CNN models called ResNet and other optimization algorithms but they found out that the method of using deep DR net with the RMSprop optimization given good results than others. The results they obtained was 61% of accuracy 65.4% of sensitivity and 73.7% of specificity.

2.5.1 Adam Optimization

DR detection with its stages has been done by Kori et al. (2018) using an ensemble convolution neural networks. The authors of the paper have used transfer learning approach to find the solution for their problem using less number of images for training. They have used Adam optimizer from a library called Keras and tried to reduce the loss while training the model. The accuracy achieved by the authors was 84% for the data size of 56 images and for the size of 44 the accuracy was 95%. This shows that the performance of the algorithm is based on the size of the data. The model may be overfitted, if an oversize of data is given to process. In this research, Adam optimizer is used to solve the issue of loss and performed well for fewer images and this may or may not be possible for a large amount of dataset.

3 Methodology

In this research project, to answer the specified question 1.2 CRISP-DM methodology is followed. Which is also said to be one of the most popular methods to be followed when implementing a deep learning, descriptive analytics or a machine learning project Faure (2018). The following figure shows the different stages of the CRISP-DM methodology.

3.1 Business Understanding

It is a primary and necessary thing to understand the problem before building any of the machine learning or deep learning model. Keeping this at the highest priority.A knowledge base developed on the cause of DR with its stages and also the processes taken until now to find them. This knowledge base also consists of some of the methods to treat DR.

Figure 3: CRISP-DM methodology

3.2 Data understanding

The primary goal of this research project is to increase the accuracy rate as well as the rate of sensitivity and specificity. By keeping this in mind for this research, an openly available data is taken with the combination of two different data sets. Due to data available in the public domain and free to construct any experiments on the data, there will not be any GDPR issue. The dataset contains images with human retinas which includes both infected and uninfected retinas. The affected retinas contain slight differences when compared with others. They consist of one of the microaneurysms, hemorrhages, exudates, cotton wool spots or in some cases all of the defects can also be found in the image. A total of 70150 images have been downloaded from the websites mentioned below1[0].But after pre-processing only 9530 images are considered for training stage and 4663 images are considered for testing.

3.2.1 Categories of infection

According to the international clinical diabetic retinopathy severity scale(ICDR), there are majorly 4 levels of infection, they are level 0,1,2,3,4. The final or stage 4 is called macular edema (ME). The below section will explain elaborately about the stages by taking images as an example.

- **No DR:** This stage says that there is no DR for the patient. Level 0 is shown in figure 4(a).
- **Mild DR:** This stage is more difficult to identify for the patient as well as doctors, this stage do not have ME. There are few microaneurysm or haemorrhages are found in this stage as shown in 4(b).
- **NPDR:** It is a mild NPDR, that consists of microaneurysm or haemorrhages with the combination of hard exudates in retina as shown in 4(c). This is the stage all the symptoms will start showing their effect on eye sight.

- **Severe NPDR:** Which is considered to be one of the sever stages where the patient feels of decreasing power in eyesight. This stage eyes consist of a large number of microaneurysm or hemorrhages, hard exudates as well as inter retinal microvascular abnormality as shown in 4D. Inter retinal microvascular abnormality is defined as the abnormal growth of nerves in the retina and these nerves will obstruct the light that passes through the retina which helps in vision.

- **PDR:** this is the final stage of DR were there are huge chances of vision loss. This stage consists of all the above defects as well as the neovascularization and pre-retinal hemorrhage which means the blood loss through the vessels from the retina which is shown in 4e.

A new category has been evolved from a few years to evaluate the performance of the specific category of diseases is a vision-threatening stage of DR. The stages come under this category are ICDR 3 and 4. Where they are considered to be most important stages to be taken care of by patient.

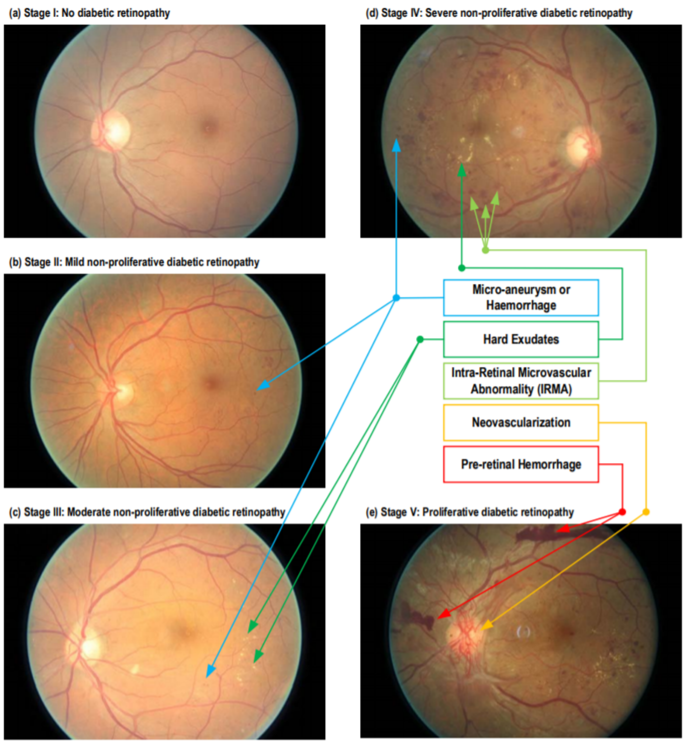

Figure 4: Different stages of DR

3.3 Data preparation

For the Neural network to work with a higher accuracy, the data need to be pre-processed. This pre-processing consists of few stages they are explained below.

3.3.1 EDA

The data used in this research is taken from a public domain, so it consists of data imbalance and due to the different lightning conditions, the data is inconsistent. This data consists of 5 categories of images where and due to the smaller number of images for some categories, it may lead to the poor performance of the algorithm. Model will not give an optimized result for the category which is not balanced in real-time. To avoid this problem, in this research the data is distributed to all categories by using data augmentation and oversampling methods.

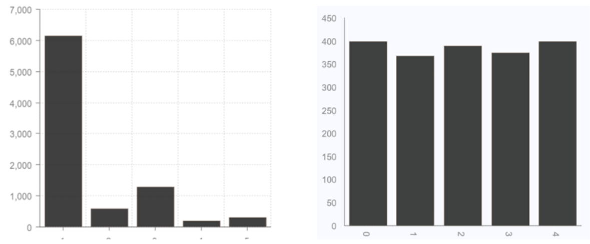

Figure 5: Data before and after EDA

As part of pre-processing in the research an oversampling technique is used. This is because the stages 1,2,3,4 consists of a very smaller number of images when compared with category 0 . All the images are then resized to a smaller number to have an improvement in the algorithm.

3.3.2 Data pre-processing

In this research, a sample of 14,193 images are used and among them, 9530 images for training and 4663 images are used for validation purpose. The data has been downloaded from a public domain, so they are in raw format and have different lighting conditions. For instance, if a dark image is considered, it is very difficult to visualize. According to Kumar et al. (2019) to solve this issue. All the images are initially converted to greyscale and then visualized. But there is an issue raised while doing the process which is, the model cannot differentiate between cotton wool spots and hard exudates Considering the

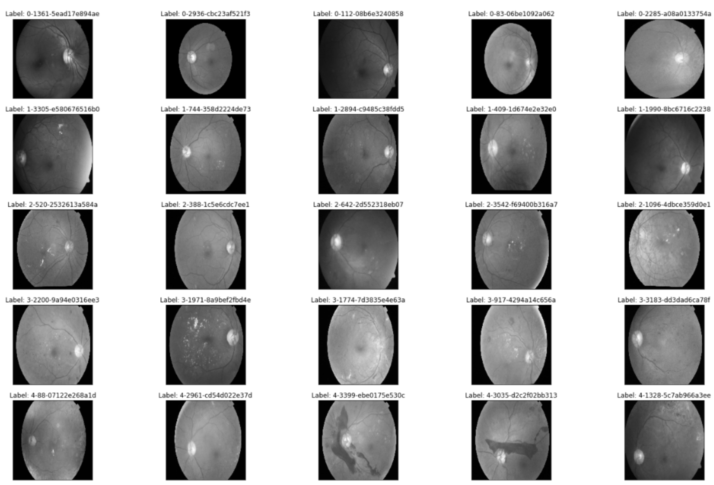

Figure 6: Grey scale of images

above figure6 the image (4,1) and (4,4) the blood cells are visible clearly, but the cotton wool spots and hard exudates cannot. This issue is solved using ben graham's process, which has secured first place in pre-processing in a kaggle competition [2]. This image pre-processing is followed in this research because it has given the best results while analyzing other dataset. This pre-processing is done using the OpenCV package of python. In this process, every image has undergone three stages. Firstly, cropping of images or resizing the images, where by doing this process all images will be in the same size and shape of 255 pixels (as an example). Next is to remove the local average color to 50% grey for each pixel of the image, this is due to the reduction of difference in lightning and finally, the images are clipped to 90% of its size. This is to remove the boundary effect, where this effect causes incomplete information at the boundary areas.

The below images will show the difference between the processed and raw images with grey shades and ben graham's process.

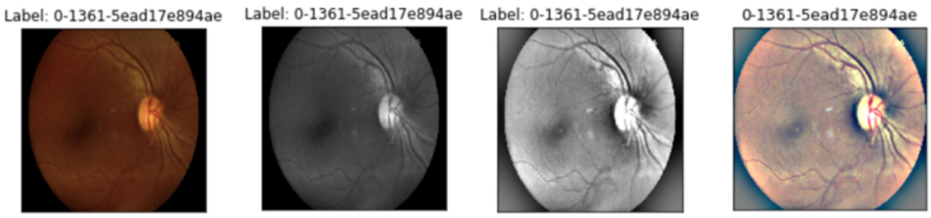

Figure 7: stages of pre-processing

From the above figure 7, the first image is the original image. The second image shows the greyscale of the original image where all the particles are not clear in detection. Then to make it clearer the sigma vale is decreased. So that, it became as shown in the third image. Even this is also not clear at detection of data at boundaries and given poor results. The fourth images show that the ben graham's shade of coloring images. After following this process there is an unwanted extra space for a large number of images. These will increase the processing time of the algorithm. To remove this unwanted area, a crop function is used and also sigma X is given as 30 for this function in this research, as shown in the below figure 8.

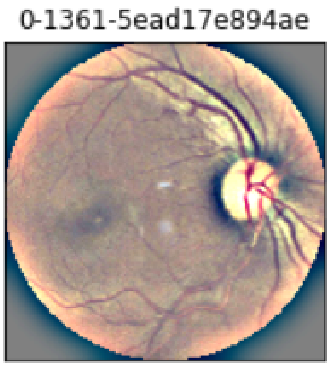

Figure 8: croped and pre-processed image

[2]http://blog.kaggle.com/2015/09/09/diabetic-retinopathy-winners-interview-1st-place-ben-graham/

3.4 Data Modeling

The immediate step after preparing data is to apply the pre-trained model to the data. While doing related work for the project, a pre-trained inception v3 model have been chosen based on many of the papers like Lam, Yi, Guo and Lindsey (2018) Wang et al. (2018) Yip et al. (2019).

After carefully considering these papers. Taken a decision that a CNN model with using of transfer learning technology, a pre-trained weights has been made good results when comparing with the other machine learning models. But the main problem with this pre-trained model is to train entire model is not possible because of time and computing power. It is possible to train only the few layers of any of the pre-trained models. with keeping this in mind, the design of this research is made in such a way that, after passing the data to the inception v3 model, the final layers are retrained and then it is attached to an external CNN network for further classification, which is discussed in the further sections.

Later on, after getting the results they are not good enough to detect all the classes of DR. To solve this issue many of the researchers have done the parameter optimization of different types. After careful consideration, as mentioned in the related work two optimization techniques are taken into consideration which are RMSprop and Adam models.Then the output of the model is optimized with both the RMSprop and Adam optimization models and the results are compared based on evaluation metrics.

3.5 Evaluation

For any machine learning model evaluation is as important as implementation. The evaluation has many metrics that can be considered. But in this research, according to the specified problem only 3 metrics have been taken into consideration they are, sensitivity, specificity, and accuracy. The description about these metrics is explained in the further sections according to the problem statement. The process to find these matrics can be done using confusion metrics which is mentioned below 1.

	Diesease status		
		Positive	Negative
Classifier Results	Positive	TP	FN
	Negative	FP	TN

Table 1: Confusion matrix

$$Sensitivity = \frac{TP}{TP + FN} \tag{1}$$

$$Specificity = \frac{TN}{TN + FP} \tag{2}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad (Afrinand\ Shill\ (2019) \tag{3}$$

3.6 Deployment

This section of CRISP-DM methodology is not applicable for this research because, this project not deploying on any other platforms.

4 Design Specification

This project completely built on the Google colab platform which is an open-source python development platform to build machine learning algorithms. Due to the huge amount of data that needs to be processed and evaluated, it requires GPU to make the process even faster. The project is built on python 3.7 and it used TensorFlow as its backend. In this model, the inception v3 model is downloaded from an external source [3] and then externally included it into the colab. This process of external storage of data is followed because colab will store all its files only for 12 hours and after its session expiry, all the data it is downloaded and uploaded will be deleted. If the process needs to run again it needs to download the data and model from the internet again. To avoid this, the pre-trained weights of the model is supplied externally to the program. All the data that includes testing, training and validation sets are uploaded to google drive and this google drive is mounted to google colab to transfer the data directly from server to server. This process of connecting server to server for the model has given positive results in terms of time in execution. The total number of images is processed, visualized and made available to model to calculate its transfer values using Google colab. For this purpose, a new page is created in colab and built entire model in the new file. Colab also has an advantage of extending its capacity of RAM to 25.51 GB and in our research, the maximum RAM used was 25GB to process all the images. After processing and modeling, the results are evaluated using the same tool and given optimistic results.

4.1 Transfer learning used in very deep inception-v3 net from keras

Transfer learning has many applications across different domains which made it a powerful deep learning technique. This technique is built on simple intuition, for instance, if a person wanted to learn a new language called French, then he can start with the words that he knows in other known languages. In the same way, transfer learning was also built and many of the pre-trained models are used by the researchers to represent new tasks DLMIA et al. (2018).

Convolution neural networks are one of the major ways that are used to solve the issue of DR with a greater degree of efficiency. Now in this section the major focus on the building of CNN using Tensorflow and Keras libraries. TensorFlow is one of the important libraries that many of the machine learning algorithms use nowadays. Keras Is also an important package of python where it consists of many of the pre-trained models like inceptionv3, Densnet, Resnet, etc. optimization functions like Rmsprop, Adam, adadelts. Metrics like accuracy. But there is a drawback in Keras that it does not have sensitivity and specificity as its metrics. To find these metrics in this research a helper function named sensitivity and specificity is used externally.

[3]https://www.kaggle.com/thophan/keraspretrainedmodel

4.2 Pre-trained Inception V3 model

As it is mentioned in the above literature review that inception v3 is one of the CNN models that perform well in image reorganization. The model is pre-trained using the ImageNet dataset and can be downloaded or can be included directly in the python environment from Keras. Inceptionv3 package. This inception algorithm consists of 48 layers and can classify up to 1000 categories of the data. The algorithm works on the images under 299*299 pixels and in this research, the images are 255*255 pixels. There are even advanced models than inception v3, but they required huge computing power and cost in evaluation. Even though it has 48 layers, in this research only the last three layers are pre-trained and then added a new neural network for processing and classification which explained in the further sections. Before the introduction of inception models, all the deep learning models work in the same drill-down methods. Where inception built using tricks to accelerate the performance of the model by taking less time in the evaluation.

Inception v3 net is introduced by google and can be implemented using Keras. Model architecture was described in the below image 9.

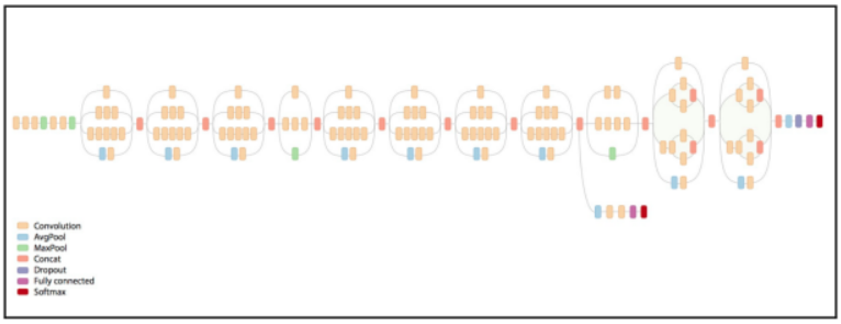

Figure 9: original inception V3 model architecture

Then a softmax layer is attached to the model to classify the newly trained model at the end of the process. This layer will classify according to the transfer values given by the model and train on the images given externally.

The below figure 10 shows some of the last few layers of the model in python environment. Though the model has huge number of parameters as shown in the above figure, but among them only 100 thousand parameters can be approximately trainable the remaining parameters cannot be trained.

5 Implementation

The implementation of this research is divided into 4 stages. They are explained in the below sections. The data for this research is taken from the public website 1[0].

5.1 Data gathering

As explained from the above section data is taken from the public domain into the local system initially. This downloading of data to a local server is done because to upload the data to google drive. Due to the size of the data is above 15GB, the researchers had

```
batch_normalization_93 (BatchNo (None, 3, 3, 384)    1152        conv2d_93[0][0]
conv2d_94 (Conv2D)              (None, 3, 3, 192)    393216      average_pooling2d_9[0][0]
batch_normalization_86 (BatchNo (None, 3, 3, 320)    960         conv2d_86[0][0]
activation_88 (Activation)      (None, 3, 3, 384)    0           batch_normalization_88[0][0]
activation_89 (Activation)      (None, 3, 3, 384)    0           batch_normalization_89[0][0]
activation_92 (Activation)      (None, 3, 3, 384)    0           batch_normalization_92[0][0]
activation_93 (Activation)      (None, 3, 3, 384)    0           batch_normalization_93[0][0]
batch_normalization_94 (BatchNo (None, 3, 3, 192)    576         conv2d_94[0][0]
activation_86 (Activation)      (None, 3, 3, 320)    0           batch_normalization_86[0][0]
mixed9_1 (Concatenate)          (None, 3, 3, 768)    0           activation_88[0][0]
                                                                 activation_89[0][0]
concatenate_2 (Concatenate)     (None, 3, 3, 768)    0           activation_92[0][0]
                                                                 activation_93[0][0]
activation_94 (Activation)      (None, 3, 3, 192)    0           batch_normalization_94[0][0]
mixed10 (Concatenate)           (None, 3, 3, 2048)   0           activation_86[0][0]
                                                                 mixed9_1[0][0]
                                                                 concatenate_2[0][0]
                                                                 activation_94[0][0]
==================================================================================================
Total params: 21,802,784
Trainable params: 1,048,768
Non-trainable params: 20,754,016
```

Figure 10: layers and hyper-parameters of inception V3

to buy additional storage from google. This data after uploading to google drive, it is then connected to google colab. It provides GPU with 25.5 GB RAM and 300 GB of storage. The connection of google drive and colab is done using two lines of code which is mentioned below.

from google.colab import drive
drive.mount('/content/drive')

5.2 Pre-processing and categorizing of data

Pre-processing has been explained more clearly in the pre-processing stage 3. The data which is used for this research is mixed in a folder and a CSV file contains all the file names and respective stage of DR . This data is then taken to colab and using python and joins concept, the data is separated according to the categories of DR. The code is designed in such a way that the file name and the rows of the CSV file are matched and then if the name matches then the corresponding image will store in the respective diagnosis category. In the same way, all the images are separated and stored as a NumPy array in the RAM. The separation of data according to the categories is essential because it is supervised learning model. In this type of learning the model feeds the network about the features of the images about a particular category.

5.3 Pre-trained model and its implementation in this research

The pre-trained model used in this research to classify the images is Google's inception v3. Where it is trained on ImageNet data, this is explained in the previous sections. Now in this section, it is explained that the changes made to the pre-trained model to make the model batter. The below image will explain the model underwent in this research.

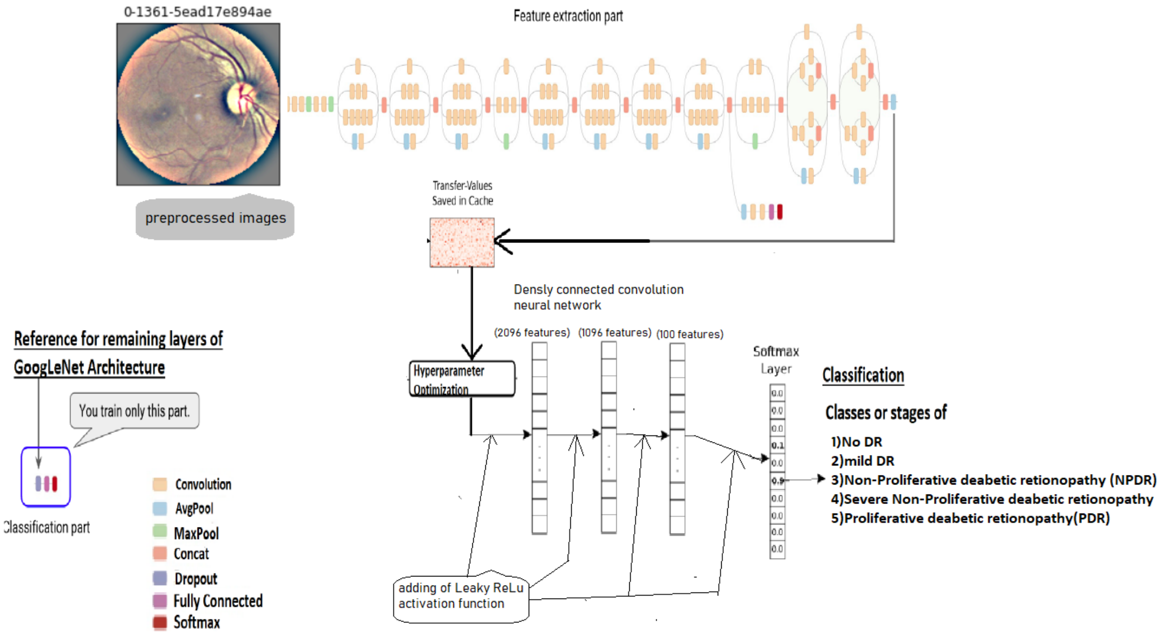

Figure 11: Process of Implementation

From the above figure11, it is stated clearly that how the model was implemented step by step. Initially, the images are pre-processed by the methods which are mentioned in the above sections and passed to the pre-trained model of Keras. Here the Keras gives only the pre-trained model but not its weights. The weights of the pre-trained model are taken from the Kaggle 3^0. After passing the data to the model the transfer-values will get as an output of the model. These are saved as a cache file for future use. If the model needs to be executed in the future again this program will not take much time in its execution. These transfer values have 1024 features for each image and all of them cannot be processed because it increases the execution time and decreases the quality of the algorithm. To solve this a hyperparameter tuning method called RMSprop and Adam methods are used and attached to a CNN layer for further process.

5.4 CNN and activation functions

As in the above literature review mentioned the reasons behind attaching the convolution layers. The output of the hyperparameter tuning methods is then attached to the 8 fully connected Dens layers, it consists of conv2d_86, conv2d_94, batch_normalization_94 layers. Where the arrangement is shown in the image 11. Leaky ReLU activation function is used in the middle of each layer and finally, a softmax layer is used to classify the results of the entire process based on the probablity given by the model. Where the drawbacks of ReLU like Dying ReLU problem are solved in the Leaky ReLU activation function.

6 Evaluation

This section will explain the results of the model and visualization of the results with suitable competitions. This section will also explain the results in terms of medical trials.

In this research, there are two hyperparameter tuning methods of Keras are followed, they are RMSprop model and Adam. Adam is an extension of the gradient decent with momentum model of optimization. Adam will update its network weights iterative based on its training data.

6.1 Evaluation of RMSprop optimization

This model of optimization is performed by most of the researchers who are working in different fields, but very few of them have been implemented in DR problem-solving. This optimization can be performed using Keras library. The results are explained below,

6.1.1 Accuracy

Accuracy is defined by the above formula 3. It is one of the important metrics to find the working condition of the algorithm. But in medical terms, accuracy will play a little role than the other metrics. In this research, the accuracy of the model is 60.3%. where 6 people out of 10 people's illnesses have been classified correctly. The reasons behind the remaining image's misconception are explained in the coming sections 6.3.

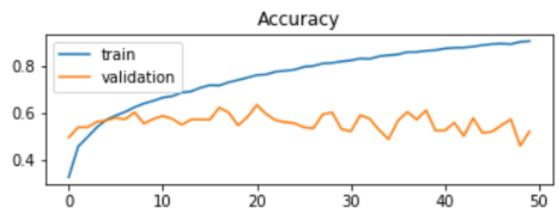

Figure 12: Accuracy of RMSprop optimization (x axis- epochs;y axis- Accuracy)

6.1.2 Sensitivity

The sensitivity is one of the important metrics that is used by most of the researches who are working in the medical trials. The sensitivity is defined from the above formula 1 from the confusion metrics 1. It is defined as the percentage of people who are DR positive and the people who are identified as DR positive by the algorithm (actual positive/predicted positive). In this metric the proposed algorithm has given an optimistic result, that it got only 57% of sensitivity. Almost 6 out of 10 people's illness has been categorised correctly. The reasons are discussed in the coming sections.

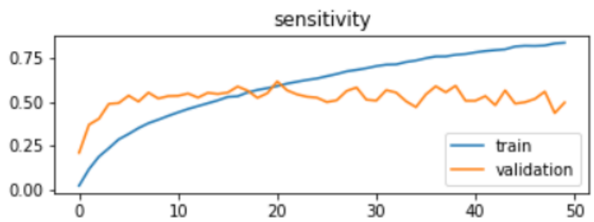

Figure 13: sensitivity of RMSprop optimization (x axis- epochs;y axis- sensitivity)

6.1.3 Specificity

Specificity is also one of the important metrics of the research to be followed while evaluating model's performance and necessarily performed for the medical problem's solving. The specificity is defined as the percentage of people who are not effected by DR and the algorithm predicted people who are not effected by DR (actual negative/predicted negative). The formula to find specificity is given above 2. In this research the specificity results got is 89%. This indicates that almost 89 out of 100 people who are not effected with DR is identified correctly by the algorithm. The below visualizations shows the results of 50 epochs.

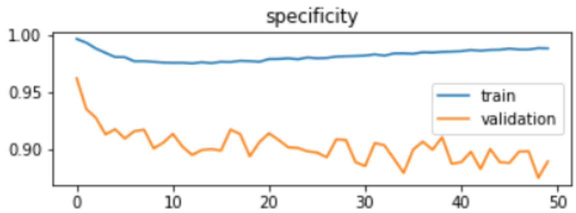

Figure 14: specificity of RMSprop optimization (x axis- epochs;y axis- specificity)

6.2 Evaluation of Adam optimization

This optimizer is performed using Keras library at the time of executing the model. The results are explained below,

6.2.1 Accuracy

As it is explained about the process of finding the accuracy and its importance. In this section, the results are explained clearly. The accuracy got when using Adam optimizer is 45%. That means only 4 people out of 10 people have been guessed correctly by the algorithm. This is a poor performance by the algorithm. The visualization below will explain 50 epochs of model execution.

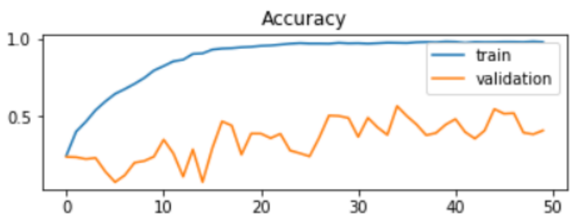

Figure 15: Accuracy of Adam optimization (x axis- epochs;y axis- Accuracy)

6.2.2 Sensitivity

These metrics are also very important in terms of medical experiments. The results are low when compared with other optimization techniques. The sensitivity got was 50%, which means only 50 people out of 100 people results are correct. The reasons behind this result are explained in the further sections.

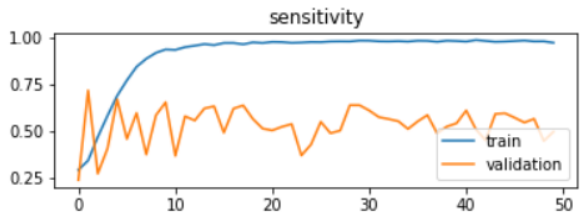

Figure 16: Sensitivity of Adam optimization (x axis- epochs;y axis- sensitivity)

6.2.3 Specificity

The sensitivity of this optimization technique is 85%. Where it has given some good results when compared with other metrics of the same optimization techniques.

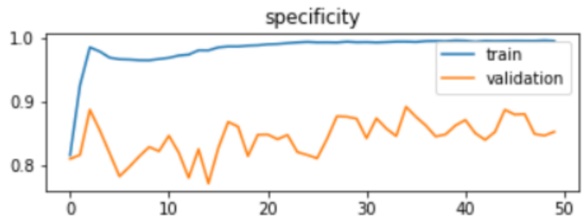

Figure 17: Specificity of Adam optimization (x axis- epochs;y axis- specificity)

6.3 Discussion

From this research, it is observed that machine learning problems can solve the issues faster with more accuracy and efficiency. This research is made in such a way that if any random image from the internet is given to the algorithm then this can detect whether the person has DR or not and if the results are positive then it can even detect the stages of the illness. This type of research helps people who can't afford to take treatments in detecting DR. The results may not be accurate at this point of time, but the state-of-the-art machine learning techniques can improve the current state of detection.

Although this research has given optimistic results, the performance needs to be improved in such a way that the machine should detect the stage of the illness within no matter of time with the highest degree of accuracy.

In this research, the total number of images used are 14,193. Among most of the images belongs to category '0'. So, this causes the data imbalance issues and solved using some machine learning techniques. After the issue of data is sorted the issue of computing power came into consideration and this is solved using google cloud platform to developing a machine learning algorithm. In this research, the unlimited size of GPU is used to process the images and the size of the 25GB RAM is used to process the data. A disk size of 300 GB is provided by the tool to store the required files.

When comparing both the optimization models which are performed in this research, the RMSprop model of optimization has been performed better than that of Adam optimizer. This is because even though the Adam optimizer is the extended version of the RMSprop optimizer, it does not work properly on this data. Many research papers suggest various types of optimizations models with a pre-trained model. which gives better results than inception v3 model and even other models give lesser results than the proposed algorithm and obtained results in this paper. So, it is evident that without experimenting with different models on different types of data, no one knows which model works best for which type of data.

While learning the model, it improves its accuracy by the number of epochs given to the model. The model can be stopped at the point when it reaches expected accuracy. In this research 6 out of 10 images are predicted correctly and the remaining 4 images are wrong in predicting because of many reasons. Some of the reasons that might affect this algorithm's accuracy were pre-processing methods, dataset size, feature optimizations and many more. But in this research, these all have been solved to some extent and need to be improved more in the future.

7 Conclusion and Future Work

The main intention of this research is to find,

- To differentiate between the people who are affected and not effected by DR.
- To find the accurate infected retinas using different pre-processing techniques.
- Considering these features, differentiating between categories of DR and classifying images according to the features.
- Increase the accuracy of the model using different feature optimization techniques.
- Visualizing the performance of the algorithm by using different evaluation metrics like sensitivity, specificity, and accuracy.

This research majorly focuses on categorize the stage of DR. Many of the external technological competitions are conducted by different companies and hospitals around the world on different platforms to solve this puzzle. An attempt has been made to solve this issue with this research.

Even though this research given an optimistic result but there are some improvements needs to be made to this research to obtain even better and reliable results. Currently, this research required a huge number of images and computing power to build a model, in the future, this problem needs to be solved. In the pre-processing stage cropping of images is performed in this research, this may lead to the loss of data like scabs/ wools around the images, which has been solved to little extent but needs to be improved. The pre-processing like changing the background color of the retina should be performed. so that, the blood cells, cotton wool balls, etc., will emboss clearly.

In this research the pretrained model used is Inception V3, this model is pre-trained on image new dataset. The dataset does not consist of images of the retina. The model uses the transfer values of the training data and extracts 2048 features from every image. This is a very long and time taking process. In the future many CNN models can be considered like vgg16 and 19, Resnet50, DenseNet169, DenseNet201 to solve this issue. These models can provide rapid, reliable, and needs less computing power than the inceptionV3 model which is used in this research.

Object detection on the retina to be made using CNN models, For instance, blood vessels, reorganization of the exact shape of the eyeball, etc. This is important because the extracted features of retina color and shapes are much similar to the features of optic disc. This similarity effects in detecting and classifying of DR. finally the research should focus on detecting the stages of DR with more degree of accuracy and also with high reliability. The cost in executing the data also matters(It includes the computing power as well). As in the above sections, it is specified that the inception V3 algorithm required much computation power which is a drawback of the algorithm, but it pawed way for most of the machine learning algorithms to develop ensemble algorithms with ResNet, DensNet, etc.

Last but not the least there is another improvement can be made in this research is the increase in the size of the dataset where Lam, Yu, Huang and Rubin (2018) said in his paper that about 93-96% of recall rate has been recorded for the images of 60000 in binary classification but it's not been proven when 120,000 is taken into consideration. So, it is evident that the increase in the size of the dataset may also affect the performance of the algorithm in either positive or negative ways.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y. and Zheng, X. (2016). TensorFlow: A System for Large-Scale Machine Learning, pp. 265–283.
URL: *https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi*

Abràmoff, M. D., Lou, Y., Erginay, A., Clarida, W., Amelon, R., Folk, J. C. and Niemeijer, M. (2016). Improved Automated Detection of Diabetic Retinopathy on a Publicly Available Dataset Through Integration of Deep Learning, *Investigative Ophthalmology & Visual Science* **57**(13): 5200–5206.
URL: *https://iovs.arvojournals.org/article.aspx?articleid=2565719*

Adem, K. (2018). Exudate detection for diabetic retinopathy with circular Hough transformation and convolutional neural networks, *Expert Systems with Applications* **114**: 289–295.
URL: *http://www.sciencedirect.com/science/article/pii/S0957417418304780*

Afrin, R. and Shill, P. C. (2019). Automatic Lesions Detection and Classification of Diabetic Retinopathy Using Fuzzy Logic, *2019 International Conference on Robotics,Electrical and Signal Processing Techniques (ICREST)*, pp. 527–532.

Ardiyanto, I., Nugroho, H. A. and Buana, R. L. B. (2017). Deep learning-based Diabetic Retinopathy assessment on embedded system, *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 1760–1763. ISSN: 1557-170X.

Chowdhury, M. M. H., Meem, N. T. A. and Marium-E-Jannat (2019). A Machine Learning Approach to Detect Diabetic Retinopathy Using Convolutional Neural Network, *in* M. S. Uddin and J. C. Bansal (eds), *Proceedings of International Joint Conference on Computational Intelligence*, Algorithms for Intelligent Systems, Springer Singapore, pp. 255–264.

DLMIA, ML-CDS and MICCAI (2018). *Deep learning in medical image analysis and multimodal learning for clinical decision support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, held in conjunction with MICCAI 2018, Granada, Spain, September 20, 2018: proceedings*, number 11045 in *Lecture notes in computer science*, Springer, Cham.

Faure, s. (2018). Descriptive analytics, machine learning, and deep learning viewed via the lens of CRISP-DM.
URL: *https://www.kdnuggets.com/descriptive-analytics-machine-learning-and-deep-learning-viewed-via-the-lens-of-crisp-dm.html/*

Gargeya, R. and Leng, T. (2017). Automated Identification of Diabetic Retinopathy Using Deep Learning, *Ophthalmology* **124**(7): 962–969.

Garside, K., Henderson, R., Makarenko, I. and Masoller, C. (2019). Topological data analysis of high resolution diabetic retinopathy images, *PLOS ONE* **14**(5): e0217413.
URL: *https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0217413*

Garzón-Alfonso, C. C. and Rodríguez-Martínez, M. (2018). Twitter Health Surveillance (THS) System, *2018 IEEE International Conference on Big Data (Big Data)*, pp. 1647–1654. ISSN: null.

Gogul, I. and Kumar, V. S. (2017). Flower species recognition system using convolution neural networks and transfer learning, *2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN)*, pp. 1–6. ISSN: null.

Joseph, A. and Geetha, P. (2019). Facial emotion detection using modified eyemap–mouthmap algorithm on an enhanced image and classification with tensorflow, *The Visual Computer* .
URL: *https://doi.org/10.1007/s00371-019-01628-3*

Kori, A., Chennamsetty, S. S., P., M. S. K. and Alex, V. (2018). Ensemble of Convolutional Neural Networks for Automatic Grading of Diabetic Retinopathy and Macular Edema, *arXiv:1809.04228 [cs]* . arXiv: 1809.04228.
URL: *http://arxiv.org/abs/1809.04228*

Kumar, S., Adarsh, A., Kumar, B. and Singh, A. K. (2019). An automated early diabetic retinopathy detection through improved blood vessel and optic disc segmentation, *Optics & Laser Technology* **121**: 105815.
URL: *http://www.sciencedirect.com/science/article/pii/S0030399219311089*

Lam, C., Yi, D., Guo, M. and Lindsey, T. (2018). Automated Detection of Diabetic Retinopathy using Deep Learning, *AMIA Summits on Translational Science Proceedings* **2018**: 147–155.
URL: *https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5961805/*

Lam, C., Yu, C., Huang, L. and Rubin, D. (2018). Retinal Lesion Detection With Deep Learning Using Image Patches, *Investigative Ophthalmology & Visual Science* **59**(1): 590–596.
URL: *https://iovs.arvojournals.org/article.aspx?articleid=2670996*

Masood, S., Luthra, T., Sundriyal, H. and Ahmed, M. (2017). Identification of diabetic retinopathy in eye images using transfer learning, *2017 International Conference on Computing, Communication and Automation (ICCCA)*, pp. 1183–1187. ISSN: null.

Meng, T., Wu, C., Jia, T., Jiang, Y. and Jia, Z. (2018). Recombined Convolutional Neural Network for Recognition of Macular Disorders in SD-OCT Images, *2018 37th Chinese Control Conference (CCC)*, pp. 9362–9367.

Szegedy, C., Ioffe, S., Vanhoucke, V. and Alemi, A. (2016). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning, *arXiv:1602.07261 [cs]* . arXiv: 1602.07261.
URL: *http://arxiv.org/abs/1602.07261*

Takahashi, H., Tampo, H., Arai, Y., Inoue, Y. and Kawashima, H. (2017). Applying artificial intelligence to disease staging: Deep learning for improved staging of diabetic retinopathy, *PLOS ONE* **12**(6): e0179790.
URL: *https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0179790*

Wagner, F. H., Sanchez, A., Tarabalka, Y., Lotte, R. G., Ferreira, M. P., Aidar, M. P. M., Gloor, E., Phillips, O. L. and Aragão, L. E. O. C. (2019). Using the U-net convolutional network to map forest types and disturbance in the Atlantic rainforest with very high resolution images, *Remote Sensing in Ecology and Conservation* **n/a**(2019).
URL: *https://zslpublications.onlinelibrary.wiley.com/doi/abs/10.1002/rse2.111*

Wang, X., Lu, Y., Wang, Y. and Chen, W. (2018). Diabetic Retinopathy Stage Classification Using Convolutional Neural Networks, *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, pp. 465–471.

Wongsuphasawat, K., Smilkov, D., Wexler, J., Wilson, J., Mané, D., Fritz, D., Krishnan, D., Viégas, F. B. and Wattenberg, M. (2018). Visualizing Dataflow Graphs of Deep Learning Models in TensorFlow, *IEEE Transactions on Visualization and Computer Graphics* **24**(1): 1–12.

Xia, X., Xu, C. and Nan, B. (2017). Inception-v3 for flower classification, *2017 2nd International Conference on Image, Vision and Computing (ICIVC)*, pp. 783–787. ISSN: null.

Xie, S., Kirillov, A., Girshick, R. and He, K. (2019). Exploring Randomly Wired Neural Networks for Image Recognition, *arXiv:1904.01569 [cs]* . arXiv: 1904.01569.
URL: *http://arxiv.org/abs/1904.01569*

Xu, K., Feng, D. and Mi, H. (2017). Deep Convolutional Neural Network-Based Early Automated Detection of Diabetic Retinopathy Using Fundus Image, *Molecules* **22**(12): 2054.
URL: *https://www.mdpi.com/1420-3049/22/12/2054*

Yin, Y., Juan, C., Chakraborty, J. and McGuire, M. P. (2018). Classification of Eye Tracking Data Using a Convolutional Neural Network, *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 530–535. ISSN: null.

Yip, M. Y. T., Lim, Z. W., Lim, G., Quang, N. D., Hamzah, H., Ho, J., Bellemo, V., Xie, Y., Lee, X. Q., Lee, M. L., Hsu, W., Wong, T. Y. and Ting, D. S. W. (2019). Enhanced Detection of Referable Diabetic Retinopathy via DCNNs and Transfer Learning, *in* G. Carneiro and S. You (eds), *Computer Vision – ACCV 2018 Workshops*, Lecture Notes in Computer Science, Springer International Publishing, pp. 282–288.