Hacettepe University Department of Computer Engineering

Name & Surname	: Baturalp Furkan KARAMUS
Identity Number	: 21527137
Course	: BBM416 - Fundamentals Of Computer Vision
Project	: Diabetic Retinopathy Project
Advisors	: Assoc. Prof. Dr. Nazlı İKİZLER CİNBİŞ
Due	: 22.06.19
E-mail	: b21527137@cs.hacettepe.edu.tr

1 Abstract

Diabetic retinopathy (DR), one of today's major health problems, can cause permanent blindness if it is not diagnosed early. To train qualified personnel as the only means of early diagnosis is very long and takes a long time to make the diagnosis. This study applies convolutional neural networks (CNNs) to identify features of early-stage diabetic retinopathy. With the information we get from the trained datasets, we want to make a project where we can learn the status of both healthy and sick individuals.

Not only can we get good results using CNN data, we will also try to get the most accurate results by applying a heatmap process on suspicious images. This study demonstrates the potential to use regionally - trained CNNs to generate probability maps and also output predictions in retinal scans of subtle diabetic retinopathies.

2 Introduction

Approximately four hundred and twenty million people worldwide have been diagnosed with diabetes mellitus. The prevalence of this disease has doubled in the past 30 years and is only expected to increase, particularly in Asia. Of those with diabetes, approximately one-third are expected to be diagnosed with diabetic retinopathy (DR), a chronic eye disease that can progress to irreversible vision loss.

Early detection, which is critical for good prognosis, relies on skilled readers and is both labor and time-intensive. This poses a challenge in areas that traditionally lack access to skilled clinical facilities. Moreover, the manual nature of DR screening methods promotes widespread inconsistency among readers.

Finally, given an increase in prevalence of both diabetes and associated retinal complications throughout the world, manual methods of diagnosis may be unable to keep apace with demand for screening services.

Automated techniques for diabetic retinopathy diagnoses are essential to solving these problems. While deep learning for binary classification in general has achieved high validation accuracies, multi-stage classification results are less impressive, particularly for early-stage disease.

3 Diabetic Retinopathy

Diabetic retinopathy, also known as diabetic eye disease, is a medical condition in which damage occurs to the retina due to diabetes mellitus. It is a leading cause of blindness.

Diabetic retinopathy affects up to 80 percent of those who have had diabetes for 20 years or more. At least 90 percent of new cases could be reduced with proper treatment and monitoring of the eyes. The longer a person has diabetes, the higher his or her chances of developing diabetic retinopathy. Each year in the United States, diabetic retinopathy accounts for 12 percent of all new cases of blindness. It is also the leading cause of blindness in people aged 20 to 64.

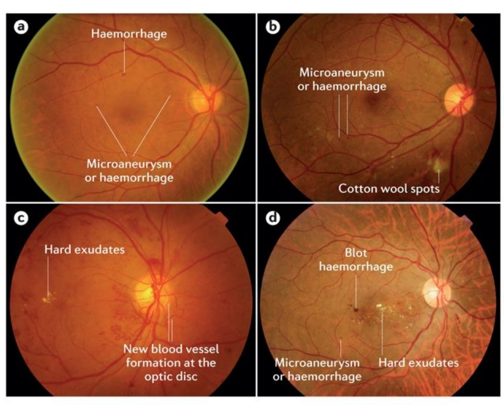


Figure 1: Basically Retinopathy Images Like That.

.

3.1 Early Detection

Early detection refers to measures that can be taken to diagnose disease as early as possible, when the disease is easiest to treat. As researchers discover more about what early stages of disease look like – under the microscope, in scans, or even on our own bodies – we can learn more about what signs or symptoms to look out for and what other measures we can take to be vigilant about recognizing diseases in its earliest forms.

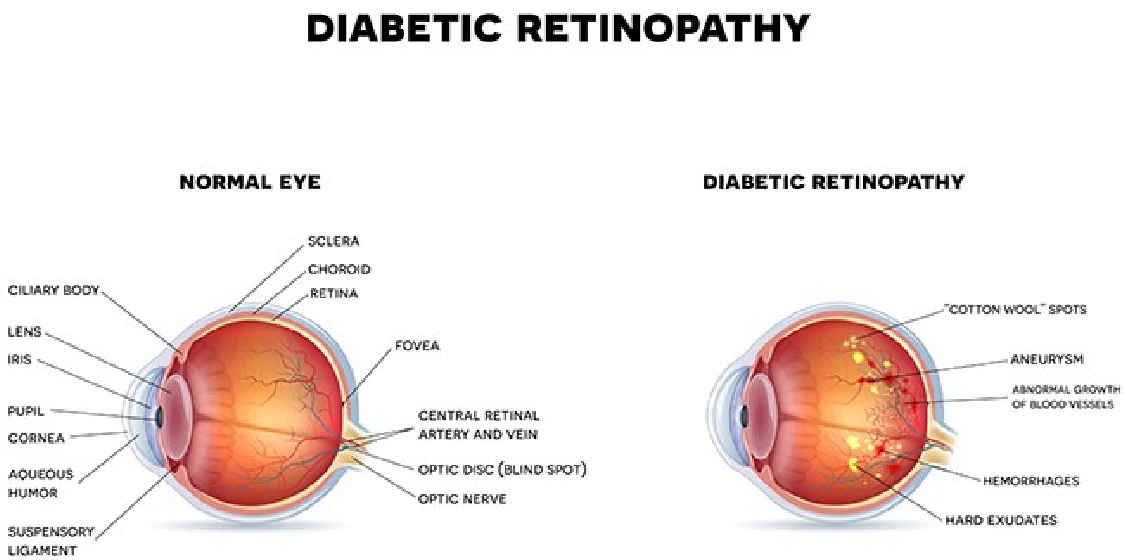


Figure 2: Differences between normal and diabetic eye.

4 Convolutional Neural Networks

Convolutional neural networks are deep artificial neural networks that are used primarily to classify images (e.g. name what they see), cluster them by similarity (photo search), and perform object recognition within scenes. They are algorithms that can identify faces, individuals, street signs, tumors, platypuses and many other aspects of visual data.

Convolutional networks perform optical character recognition (OCR) to digitize text and make natural-language processing possible on analog and hand-written documents, where the images are symbols to be transcribed. CNNs can also be applied to sound when it is represented visually as a spectrogram. More recently, convolutional networks have been applied directly to text analytics as well as graph data with graph convolutional networks.

The efficacy of convolutional nets (ConvNets or CNNs) in image recognition is one of the main reasons why the world has woken up to the efficacy of deep learning. They are powering major advances in computer vision (CV), which has obvious applications for self-driving cars, robotics, drones, security, medical diagnoses, and treatments for the visually impaired.

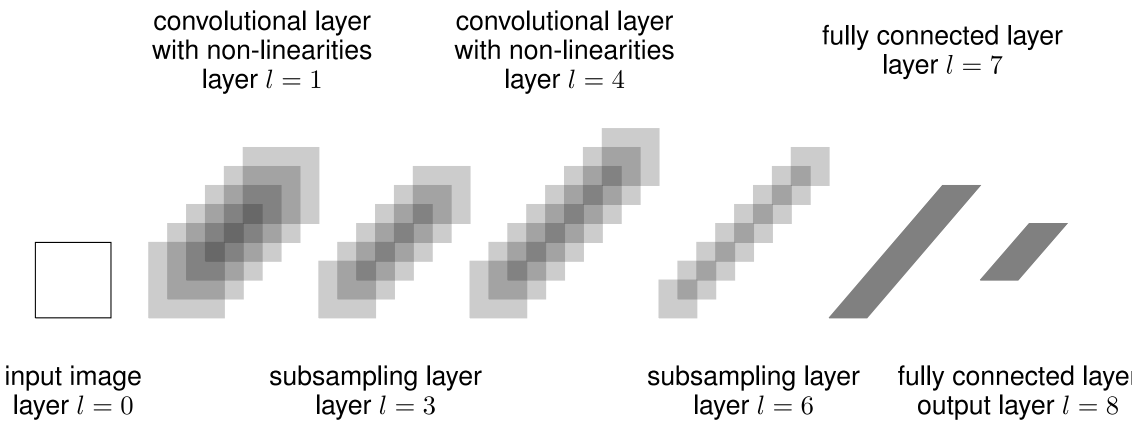


Figure 3: The architecture of the original convolutional neural network, as introduced by LeCun et al. (1989), alternates between convolutional layers including hyperbolic tangent non-linearities and subsampling layers. In this illustration, the convolutional layers already include non-linearities and, thus, a convolutional layer actually represents two layers. The feature maps of the final subsampling layer are then fed into the actual classifier consisting of an arbitrary number of fully connected layers. The output layer usually uses softmax activation functions.

There is some examples of pre trained convolutional networks from ImageNET.

4.1 AlexNet (2012)

In 2012, AlexNet significantly outperformed all the prior competitors and won the challenge by reducing the top-5 error from 26 percent to 15.3 percent. The second place top-5 error rate, which was not a CNN variation, was around 26.2 percent.

The network had a very similar architecture as LeNet by Yann LeCun et al but was deeper, with more filters per layer, and with stacked convolutional layers. It consisted 11x11, 5x5,3x3, convolutions, max pooling, dropout, data augmentation, ReLU activations, SGD with momentum. It attached ReLU activations after every convolutional and fully-connected layer.

AlexNet was trained for 6 days simultaneously on two Nvidia Geforce GTX 580 GPUs which is the reason for why their network is split into two pipelines. AlexNet was designed by the SuperVision group, consisting of Alex Krizhevsky, Geoffrey Hinton, and Ilya Sutskever.

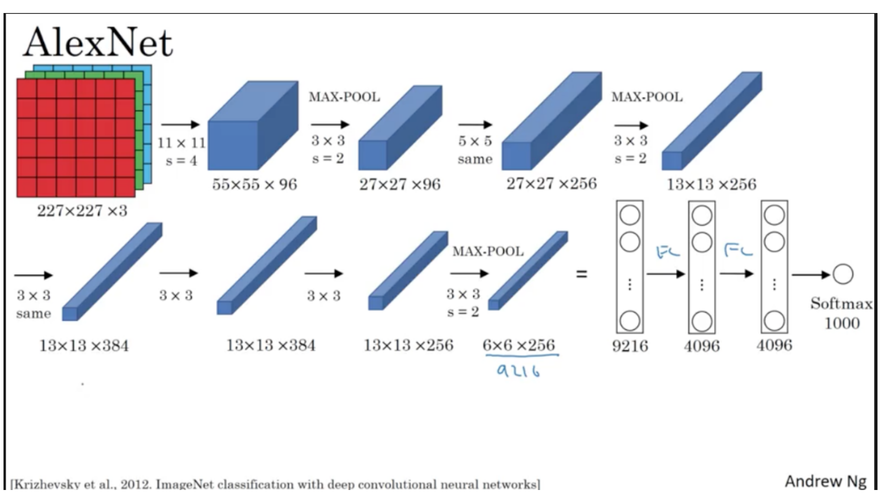


Figure 4: AlexNET Model

4.2 DenseNET(2017)

DenseNet (Dense Convolutional Network) is reviewed. This is the paper in 2017 CVPR which got Best Paper Award with over 2000 citations. It is jointly invented by Cornwell University, Tsinghua University and Facebook AI Research (FAIR).

With dense connection, fewer parameters and high accuracy are achieved compared with ResNet and Pre-Activation ResNet

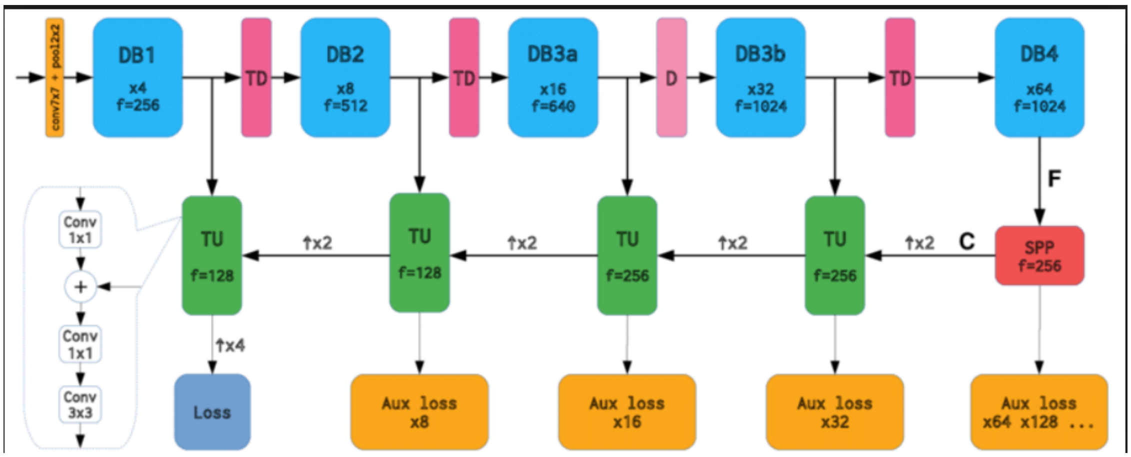


Figure 5: DenseNET Model

4.3 VGGNet (2014)

Runner-up at the ILSVRC 2014 competition is dubbed VGGNet by the community and was developed by Simonyan and Zisserman . VGGNet consists of 16 convolutional layers and is very appealing because of its very uniform architecture. Similar to AlexNet, only 3x3 convolutions, but lots of filters.

Trained on 4 GPUs for 2–3 weeks. It is currently the most preferred choice in the community for extracting features from images. The weight configuration of the VGGNet is publicly available and has been used in many other applications and challenges as a baseline feature extractor. However, VGGNet consists of 138 million parameters, which can be a bit challenging to handle.

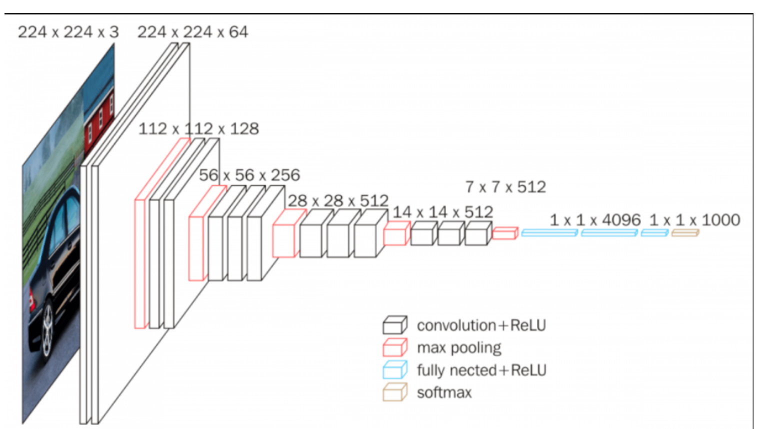


Figure 6: VGGNET Model

4.4 ResNet(2015)

At last, at the ILSVRC 2015, the so-called Residual Neural Network (ResNet) by Kaiming He et al introduced anovel architecture with "skip connections" and features heavy batch normalization. Such skip connections are also known as gated units or gated recurrent units and have a strong similarity to recent successful elements applied in RNNs. Thanks to this technique they were able to train a NN with 152 layers while still having lower complexity than VGGNet. It achieves a top-5 error rate of 3.57 percent which beats human-level performance on this dataset.

AlexNet has parallel two CNN line trained on two GPUs with cross-connections, GoogleNet has inception modules ,ResNet has residual connections.

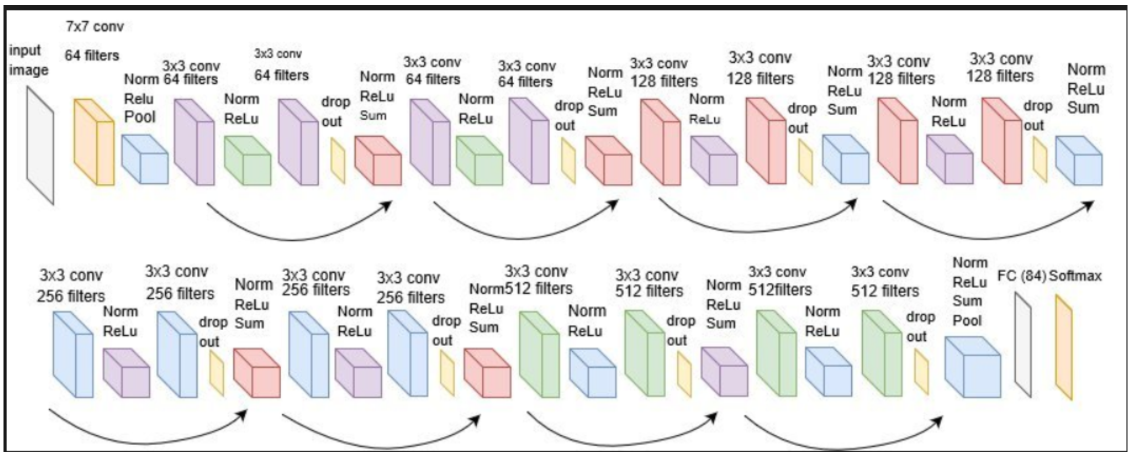


Figure 7: ResNET Model

5 Dataset

The dataset we are going to use is: https://ites.google.com//site//hosseinrabbanikhorasgani//datasets-1

Increase image numbers in this dataset with data augmentation.After that we got:

124 images under train

66 images under val

68 images under test

6 Methods

6.1 CNN Architectures

In order to assess the strengths and limitations of CNNs, several architectures were trained and tested with particular focus on a 4 different model AlexNET,VGG16,Resnet18,DenseNET. This very efficient way to achieve succesfull recognition and testing with different networks with each other. Increased convolution layers and improved utilization of internal network computing resources allow the network to learn deeper features.

The max pooling sample-based discretization process was performed and in some models have DropOut layers. The networks was then flattened to one dimension after the final convolutional block. Dropout of network layers was performed until reaching the dense five node output layer, which uses a softmax activation function to compute the probability of classification labels.Rectified Linear Unit applies gradient and improvise backpropagation. Effect on non-linearity netwroks. The network uses L2 regularization.

6.2 Preprocessing

All images were converted to a hierarchical data format for preprocessing, data augmentation, and training. Preprocessing involved several steps: images were cropped for fit to the models. Images were normalized by subtracting the minimum pixel intensity from each channel and dividing by the mean pixel intensity to represent pixels in the range 0 to 1.

6.3 Data Augmentation

We augmented the number of images in real-time to improve network localization capability and reduce overfitting. We implemented salt and pepper noise each image and create larger dataset for training and validation process. These transformations are particularly effective when applied to disease which are the most difficult to grade and fewest in number.

(a) Noisy Normal Image (b) Normal Image

Figure 8: Image with Salt Pepper noised....

6.4 Training And Testing Models

A deep learning model used with GPU. Google Collab used while implementation.Learning models and methods implementing with torch libraries.

The images were cropped to area size 224x224 and used as input data by Imagenet models previously trained for generic classification tasks. The test subset folder contained 68 images from disjoint from training data. This training system, which offered extensive hyperparameter selections, was then used to build model prototypes over 30 epochs requiring approximately 2-3 minutes each to complete.

6.5 Transfer Learning

Transfer learning based approaches were executed with pretrained AlexNet, ResNet, Vgg16, DenseNet architectures from ImageNet. The last fully connected layer was removed, then a transfer learning scenario was followed by treating the remaining network components as a fixed feature extractor for the new dataset. The transfer learning retains initial pretrained model weights and extracts image features via a final network layer.

7 Experiments

7.1 Software Usage

In this part i will use pre trained models and freeze all layers then start train network. Before starting train process we prepare network for training process. That means we have 2 classes and pre trained network has 1000 classes for last layer. We must change these last layers like 4096x2 dimension. After do that initilaze optimizer. We use ADAM and SGD optimizers in this example.

For this optimizer i decide best learning rate values that are 0.001 , 0.0001. This gives me best result.Assign criterion as Cross Entropy Loss and assign a scheduler. Then start training. After the training we change model to evaluation mode and evaluate test images response then calculate accuracy for best.

When i increase batch size program run faster but loss increase.

7.2 Implementation Details

In this part there is some explanation about test steps:

- Create new dataset for read images.
- Create dataloader objects.
- Create models which pre-tranined true.
- Use this models for training model with all training and validation images.
- Calculate average loss for each epoch and draw graph
- Compute test results with different hyperparameters.

7.3 Results:

The practicality of transfer learning was investigated by using a baseline prototype consisting of pretrained models obtained from the ImageNet visual object recognition database. The prototype was trained on the for 30 epochs using stochastic gradient descent and adam optimization with step decay learning rate initialized at 0.001 and 0,0001. The classification model validation achieved 98.03 percent as the best accuracy.

Model	Optimizer	Learning Rate	Validation Accuracy	Test Accuracy
AlexNET	SGD	0,0001	95.45	88.24
AlexNET	SGD	0,00001	96.25	90.42
AlexNET	Adam	0,0001	90.45	85.24
AlexNET	Adam	0,00001	93.94	97.06
VggNET	SGD	0,0001	94.12	92.95
VggNET	SGD	0,00001	95.26	89.06
VggNET	Adam	0,0001	95.45	95.59
VggNET	Adam	0,00001	96.97	98.53
ResNET	SGD	0,00001	93.24	92.43
ResNET	SGD	0,0001	94.25	90.25
ResNET	Adam	0,00001	92.42	94.12
ResNET	Adam	0,0001	93.94	94.12
DenseNET	SGD	0,00001	95.60	90.55
DenseNET	SGD	0,0001	88.55	85.35
DenseNET	Adam	0,00001	93.94	97.06
DenseNET	Adam	0,0001	96.97	94.12

7.4 Loss/Validation Graphs

In this part i will describe training process and test process seperately. In training process train/valid loss graph will be explained. Then comment about overfitting network.

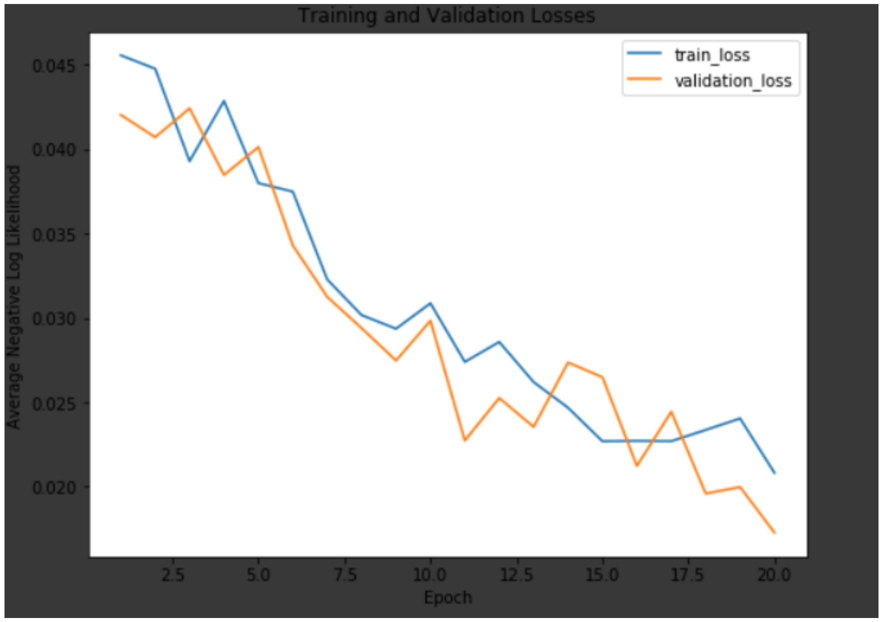


Figure 9: AlexNET

This graph shows while network training validation loss in decreasing in some part but not informaly. Some part valid loss decreasing and increasing so we can't stop immediately at any point. After the 20 epoch i choose best network checkpoint and save them if network getting overfitted it doesn't matter.This model also has dropout layer which value 0,5.

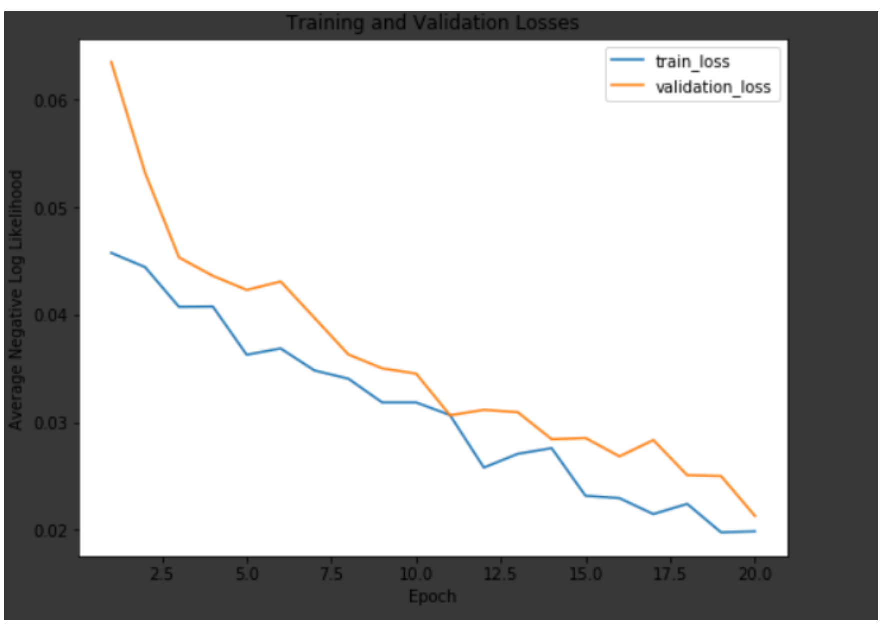


Figure 10: DenseNET Model

This graph shows more clear about process of training. In starting position we see that program underfitting. While train continue until 20 epoch training process going succesfully.

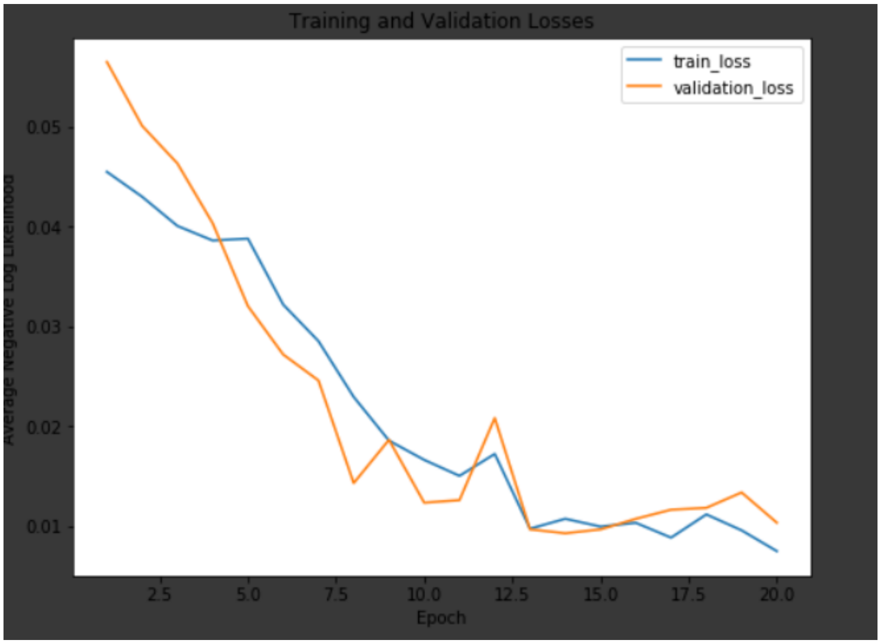


Figure 11: VGG16 Model

This graph shows more clear about process of training. In starting position we see that program underfitting. While train continue until 20 epoch training process going succesfully.

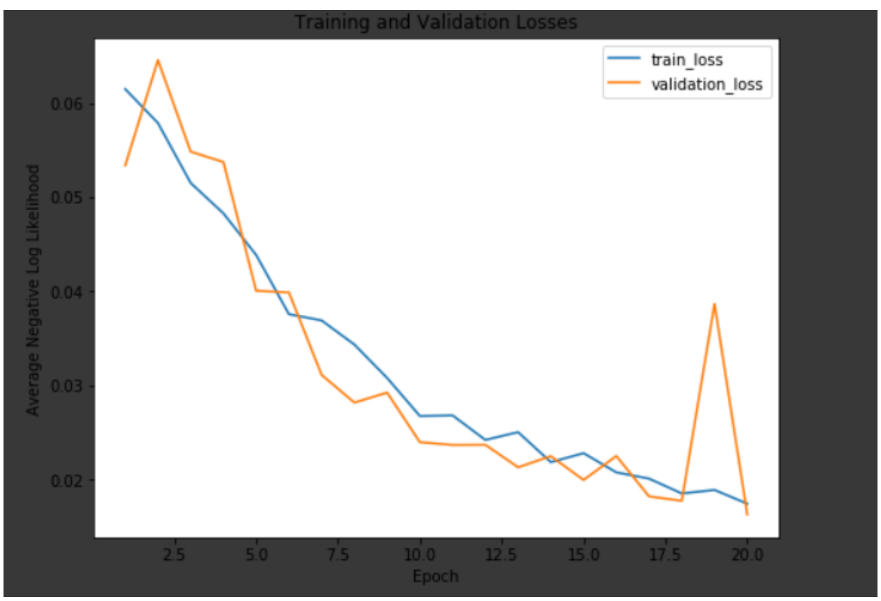
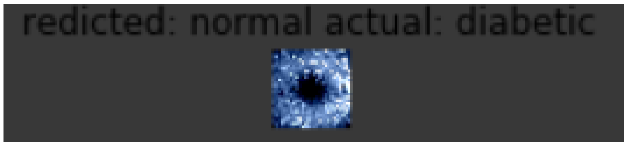


Figure 12: ResNET Model

7.5 Test Result

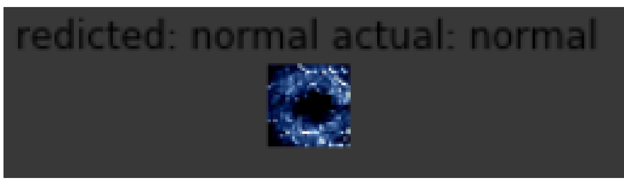
These are sample test results from random models. On the top of them predicted class and actual class is written.



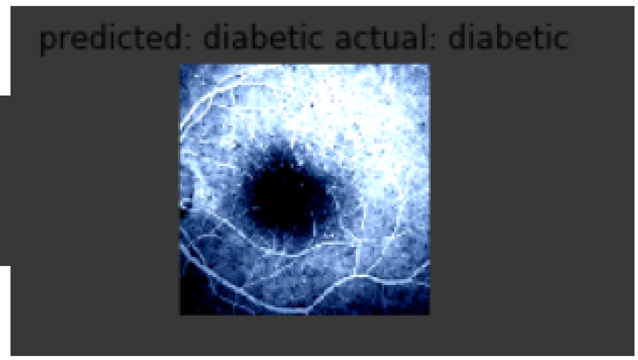
(a) 1a

(b) 1b

Figure 13: Test Results....

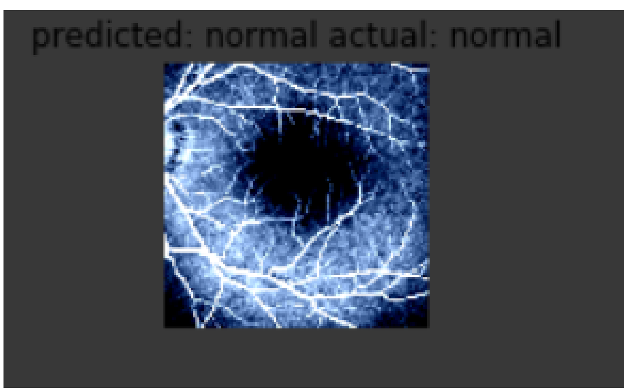


(a) 1a

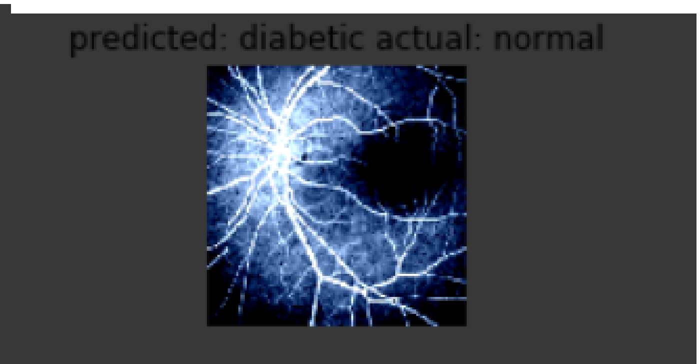


(b) 1b

Figure 14: Test Results...



(a) 1a



(b) 1b

Figure 15: Test Results....

8 Functions:

evalModel() Evaluate loss about test images on training network.

trainModel() Train on new created model.

For other process built-in functions used.

9 Conclusion:

Recognition and screening is an important thing to prevent a significant portion of vision loss in our population. In recent years, researchers have added CNNs to algorithms used to screen for diabetic diseases. CNNs promise that the physician will benefit from the large amount of images collected for the screening and learn from the pixels.

However, while we achieve state-of-the-art performance with CNNs using binary classifiers, the model performance degrades with increasing number of classes. Heat colour images and noisy image with ImageNET database incerase accuracy and give us 90-95 percent test result for each test image.According to my comment this experiment achieve recognition of diabetic retina images.

Together, it achieves the most advanced performance with CNNs using a binary classifier, decreasing with an increasing number of classes in the model. Although it is tempting to predict that more data may be better in previous studies in the field.ImageNET database gives good performance in this experiment.

References

[1] https://skymind.ai/wiki/convolutional-network

[2] https://cv-tricks.com/cnn/understand-resnet-alexnet-vgg-inception/

[3] https://en.wikipedia.org/wiki/Main$_{Page}$

https://medium.com/nanonets/how-to-use-deep-learning-when-you-have-limited-data-part-2-data-augmenta

[4] https://nei.nih.gov/health/diabetic/retinopathy

[5] https://medium.com/@sidereal/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-66609148

[6] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5961805/