RETINAL FUNDUS MULTI-DISEASE IMAGE CLASSIFICATION

Project Midway Presentation

Saksham Agarwal - 2011144
Deependra Singh - 2011058

Literature Review

This review covers three different research papers focused on the development of deep learning-based diagnostic tools for various retinal diseases using fundus images.

The first paper describes the development of a diagnostic tool using convolutional neural networks (CNNs) to classify retinal diseases based on fundus images. The researchers used three different CNN models (VGG19, ResNet50, and GoogLeNet Inception v3) to classify images into two and nine classes. They achieved high accuracy for both classifications, with VGG19 achieving the highest accuracy of 99.12% for the two-class classification and ResNet50 getting 87.42% accuracy for 9 class classification . However, the accuracy improvement using the AI tool was not significant compared to without the tool. Moreover, there were some limitations with the dataset, such as different resolutions and unmatched age of patients.

There was slightly higher accuracy of diagnosis by residents using AI tool but the significant difference was the reduction of time taken in diagnosing the disease (30% to 70%}.

The second paper used a transfer learning approach with the VGG19 model to classify fundus images into three classes: normal, glaucoma, and diabetic retinopathy. The researchers achieved an accuracy of 93.58% with a learning rate of 0.001 and 150 epochs.

The third paper focused on multi-label classification of retinal diseases using transformer-based architecture (C tran). The researchers prepared their custom-made dataset by combining multiple publicly available datasets and pre-processing them for quality images. The model achieved high accuracy (AUC score) for most of the unique labels (>90%}, but two labels showed lower accuracy.

Overall, all three papers indicate that deep learning-based diagnostic tools for retinal diseases can achieve high accuracy and reduce the time taken for diagnosis. However, the limitations of the datasets and some classes with low accuracy show that more work needs to be done to improve the performance of these models.

Introduction

We are making a multi-label classification model for a set of retinal diseases (total 45) using a dataset of 3200 images. Based on the literature review, we found that the best model to apply is convolutional neural networks. For this project, we have applied two specific CNN architectures, one based on a multi images classification model for 20 labels found here, and the other is a transfer leanrning approach using various pre trained models like VGG19, ResNet50, etc.

For midway, we had two goals which were implementation of baselines and finding new datasets. As of midway, we are done with the implementation of baselines, a CNN model and a VGG19 model, with their hyperparameter tunings still left to do. The first model have gave an accuracy of 96% while the VGG model with preset values got a 40% accuracy without fine-tuning.

Preprocessing

The original fundus images were in resolution 4, 288 × 2, 848 pixels in rgb format and the size was rescaled to 50 × 50 and 100 × 100. We then used keras ImageDataGenerator method for augmenting the data. Since ours in a medical data, we have to be very careful regarding augmentation since the alignment of images may be important.

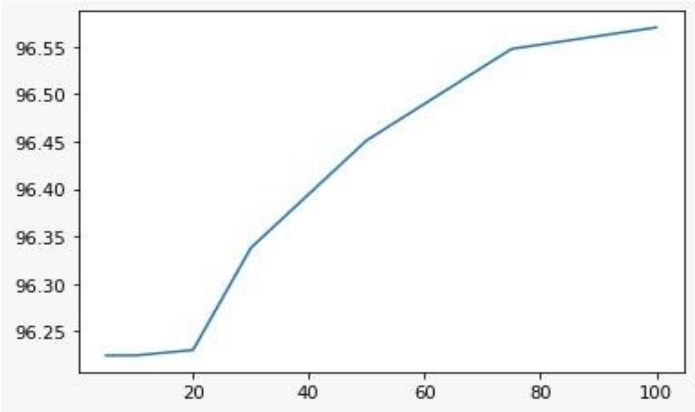
We inspected the dataset and find that few images were present with slight rotations and brightness. This motivated us to apply the various augmentation techniques on the dataset to a certain degree, to make the feature extractor invariant of orientation and position of the object in an image. The zooming of images was omitted. Tuning the augmentation parameters is yet to be done. Thereby we generated a dataset with more examples.

Model 1

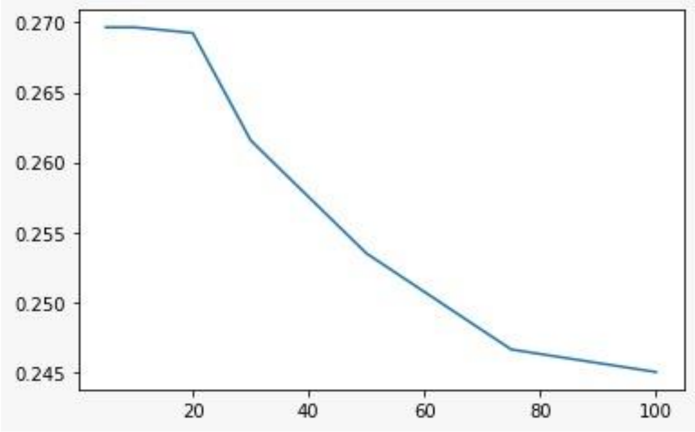
Model Training and Baseline Results

- We have tried running the algorithm for threshold 0.5 to 0.9, and the best accuracy was obtained for threshold greater than 0.7.
- We test the model by rescaling rescaling from 50 × 50 to 100 × 100 giving an improvement from 0.32 to 0.86 accuracy for 29 labels.
- However, only a few majority classes had good accuracy, none of the positive cases of minority classes was matched correctly thereby giving a high accuracy. So we changed the labels to 28 by removing disease risk.
- The model showed a slight improvement to 0.96 accuracy for 0.8 threshold. However, it is still had minority class problem.
- We plotted the variation of accuracy and hamming loss with epochs.

Layer (type)	Output Shape	Param
conv2d (Conv2D)	(None, 96, 96, 16)	1216
max_pooling2d (MaxPooling2D)	(None, 48, 48, 16)	0
dropout (Dropout)	(None, 48, 48, 16)	0
conv2d_1 (Conv2D)	(None, 44, 44, 32)	12832
max_pooling2d_1 (MaxPooling 2D)	(None, 22, 22, 32)	0
dropout_1 (Dropout)	(None, 22, 22, 32)	0
conv2d_2 (Conv2D)	(None, 18, 18, 64)	51264
max_pooling2d_2 (MaxPooling 2D)	(None, 9, 9, 64)	0
dropout_2 (Dropout)	(None, 9, 9, 64)	0
conv2d_3 (Conv2D)	(None, 5, 5, 64)	102464
max_pooling2d_3 (MaxPooling 2D)	(None, 2, 2, 64)	0
dropout_3 (Dropout)	(None, 2, 2, 64)	0
flatten (Flatten)	(None, 256)	0
dense (Dense)	(None, 128)	32896
dropout_4 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 64)	8256
dropout_5 (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 28)	1820



Accuracy Vs epochs



Loss Vs epochs

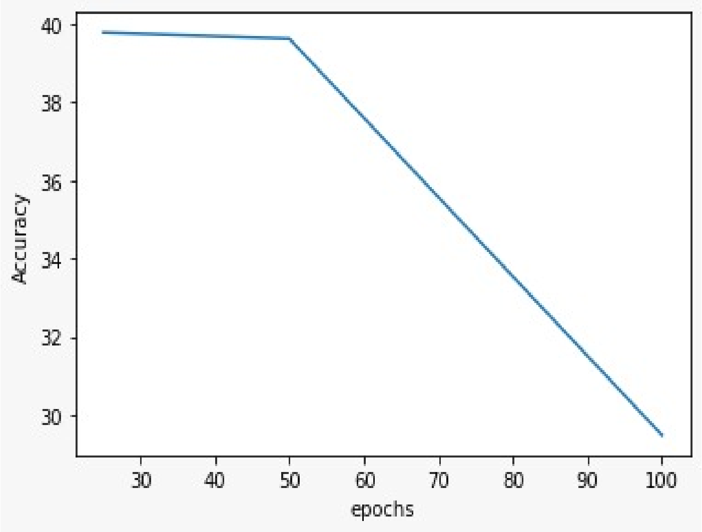
Model 2 VGG19

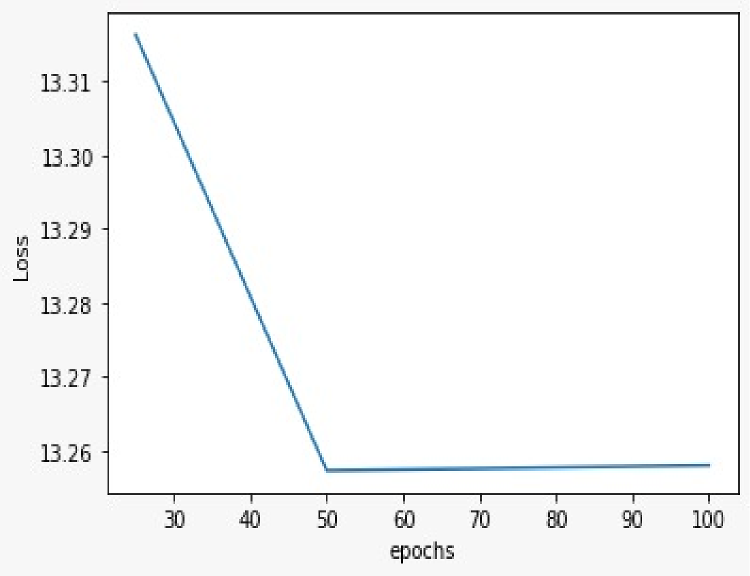
Model training and Baseline Results

- We have tried running the algorithm for threshold 0.5, batch size 32 and a learning rate of 0.001, as of now.

- The accuracy if around 40% for a threshold for 0.5. But we know from previous examples the threshold has to be greater. The majority class problem still remains.

- We plotted the variation of accuracy and loss with epochs for threshold .


40
38
36
34
32
30
Accuracy
30
40
50
60
70
80
90
100
epochs


13.31
13.30
13.29
13.28
13.27
13.26
Loss
30
40
50
60
70
80
90
100
epochs

Post Midway Work

- Using Sampling methods for handling the minority class problem.
- Find more evaluation metrics and plot corresponding graphs for the problem.
- Apply ResNet50 model for classification. And fine tune both ResNet and VGG for our data.
- Continue on with progressive transfer learning approach in both these models. • Alternatively use the MuRED data set available here. Will use during the implementation of transformers approach.
- Addition of Gluacoma label by combining two datasets using Label Superset Transform. And seeing if converting the problem to a multi class classification shows an improvement.
- Hyperparameter Tuning: The important ones are as follows-:
- Number of labels. We will try to sequentially increase the number of labels from majority class to minority class and see how trend for accuracy and loss function.
- Optimization for the augmentation parameters. We will use the first paper as reference.
- Optimization for image size, both resizing and cropping, followed by data cleaning.