Ocular disease recognition

Assignment for course:
Deep Learning for Image & Video Processing

Giorgos Vainterlis

December 2023

Contents

Abstract

This research project explores the application of computer vision and deep learning techniques with the goal of identifying the presence(or absence) of an ocular disease in digital retinal photographies. Within this scope, I have employed a modified version of the state-of-the-art image classification algorithm InceptionV3, to classify the ODIR dataset, which comprises over 5000 images of eyes, both healthy and with diseases. The project examines both the multi-label classification problem, effectively identifying the specific disease in the target image, and the binary classification problem, checking for the presence of each of the four examined diseases independently. Additionally, as the dataset was heavily imbalanced, downsampling was implemented for balancing. The best accuracies achieved for the binary classification task were 92.04% for the normal (N) versus cataract (C) class, 89.22% for normal (N) versus glaucoma (D), 91.73% for normal (N) versus glaucoma (G) and 95.28% for normal (N) versus AMD (A) while for multi-label classification, a respectable accuracy of 76.1% was reached. Furthermore, an effort is made to offer a degree of explainability to the model's decisions and explain the "blackbox" that CNN models are using filter, feature map and GradCAM visualizations. Lastly a brief inspection is made between the contrast of the input images and the accuracy of the model.

1 Introduction

The human eye is a sensory organ, part of the sensory nervous system, that reacts to visible light and allows humans to use visual information for various purposes including seeing things, keeping balance, and maintaining circadian rhythm. Ocular diseases represent a significant global health concern, necessitat-

ing efficient and accurate diagnostic tools for timely intervention. The manual diagnosis of these can often be time-consuming, error-prone, complicated and expensive. Therefore, an automated ocular disease detection system that is able to identify various eye disorders using pictures of the patients retina can be a very helpful application that helps both patients and doctors. In the context of this research project assignment for the master course Deep Learning for Image & Video Processing at Maastricht University, I delve into the realm of utilizing deep learning and computer vision techniques to enhance the diagnostic process for ocular diseases.

For this task at hand pretrained deep learning models such as InceptionV3, VGG, and ResNet can yield extremely impressive results. These models, have been honed on extensive image datasets such as ImageNet and excel at extracting intricate features. Thus, instead of opting to try to implement a model from scratch I will use the pretrained machine learning model InceptionV3.

2 Literature Review

Meng et al. [7] introduced a two-stage process employing convolutional neural networks (CNN) for Optic Disc (OD) localization in fundus images. He et al. [3] proposed automatic ocular disease classification models based on knowledge distillation, utilizing a system built through the sequential training and optimization of two deep networks. Roy et al. [9] recommended ReLayNet, a fully convolutional deep architecture, for segmenting retinal layers and fluids from Optical Coherence Tomography (OCT) scans. This technique employs an encoder-decoder network for semantic segmentation on OCT scans.

Liefers et al. [5] utilized a fully convolutional neural network with dilated convolution filters for pixel-wise classification on Optical Coherence Tomography (OCT) scans. The model's performance was assessed on a dataset comprising 400 OCT scans of patients with varying stages of age-related macular degeneration. Lee et al. introduced a CNN-based model for detecting intra-retinal fluid on OCT images. Trained on 1,289 OCT scans, the CNN model achieved a cross-validated Dice score of 0.911 for segmented images.

Playout et al. [8] proposed a novel convolutional multi-task architecture employing a supervised learning approach. The model simultaneously performed three tasks: segmentation of bright lesions, segmentation of red lesions, and lesion detection, with an area under the ROC curve of 0.839. Hu et al. [4] presented a retinal vessel segmentation technique using a convolutional neural network and fully connected conditional random fields (CRFs). The model's accuracy and effectiveness were evaluated on color fundus images from STARE and DRIVE [11] datasets.

Gulshan et al. [2] developed a deep learning-based algorithm for automating diabetic macular edema and diabetic retinopathy detection, utilizing an optimized neural network-based image classification model. Li et al. [6] proposed a deep learning-based system for detecting Glaucomatous Optic Neuropathy (GON) in color fundus photographs. The classification model, trained on 8000 color fundus images, achieved a sensitivity score of 95.6%, specificity of 92.00%, and an AUC score of 0.986.

3 The ODIR dataset

Ocular Disease Intelligent Recognition (ODIR) [1] is a structured ophthalmic database of 5,000 patients with age, color fundus photographs from left and right eyes and doctors' diagnostic keywords from specialized doctors.

This dataset is meant to represent "real-life" set of patient information collected by Shanggong Medical Technology Co., Ltd. from different hospitals/medical centers in China. In these institutions, fundus images are captured by various cameras in the market, such as Canon, Zeiss and Kowa, resulting into varied image resolutions. Annotations were labeled by trained human readers with quality control management. They classify patient into eight labels including:

1. Normal (N)
2. Diabetes (D)

3. Glaucoma (G)
4. Cataract (C)
5. Age related Macular Degeneration (A)
6. Hypertension (H)
7. Pathological Myopia (M)
8. Other diseases/abnormalities (O)

However, in the scope of this project I decided to focus on the four diseases that prove to be the most problematic for the patients, namely Diabetic Retinopathy, Glaucoma, Cataract and Age related Macular Degeneration (AMD). The methodology and model I employ can easily be expanded to accommodate the classification of the rest of the diseases in the dataset. Lastly, the dataset also contains information about the patients age and gender, features that can be used for other types of classification tasks like finding the correlation(if it exists) between age, gender and the presence of a specific disease.

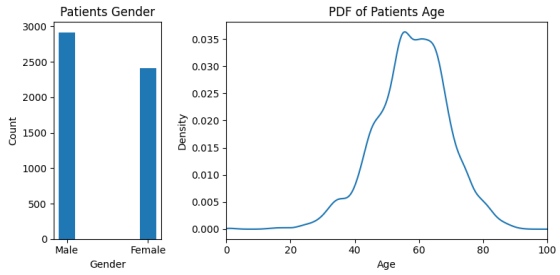

Figure 1: Gender and Age Information of Patients

3.1 Data preprocessing

As already mentioned the ODIR dataset includes a wide range of photographs with different resolutions. Before procceding to analyzing and using the image data someone would first need to choose a specific format for the images. Conveniently, the researchers of the dataset have offered their preprocessing procedure which includes an upscaling or downscaling to a 512x512 image. This procedure however has lead to some images looking a little ovalish or distorted, but this is acceptable since it will introduce some form of noise to our image data which can possible help in the robustness of our future model. Furthermore, I opted to further downsample the images to an 214x214 resolution, mostly to compensate for the limited computational resources I had in my disposal and reduce the dimensionality of the problem. Improvements of this project can include a more detailed configuration of the resolution quality and comparing the final results.

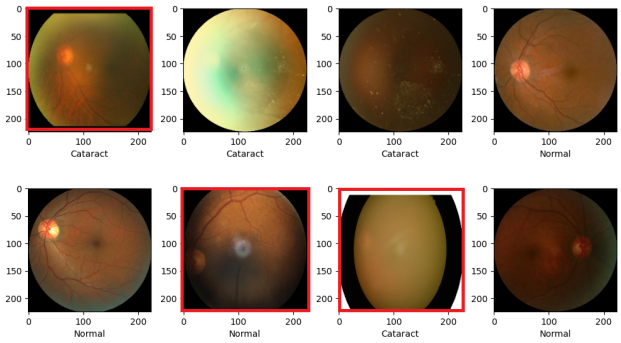

Figure 2: Cameras of different resolutions lead to images that are scaled differently

A not ideal characteristic of the ODIR dataset is that its classes are considerably unbalanced. As you can see on the figure below the number of images depicting Normal eyes and eyes suffering with Glaucoma are many times more than those of the other diseases. To counter this problem I decided to use the downsampling technique. In more detail, for the binary classification tasks I downsampled the Normal class to match the length of the compared class. For instance, since the number of AMD cases is equal to 500 I only chose 500 samples from the normal cases, I concatenated the two arrays, suffled them and worked with this minidataset. For the multi-label classification, comparatevely I downsampled the normal and Glaucoma classes to match the other three as shown in figure 1.

Lastly, another matter that had to be addressed was the non-exclusivity of the labels and a slight ambiguosity in the diagnoses for the images. To counter

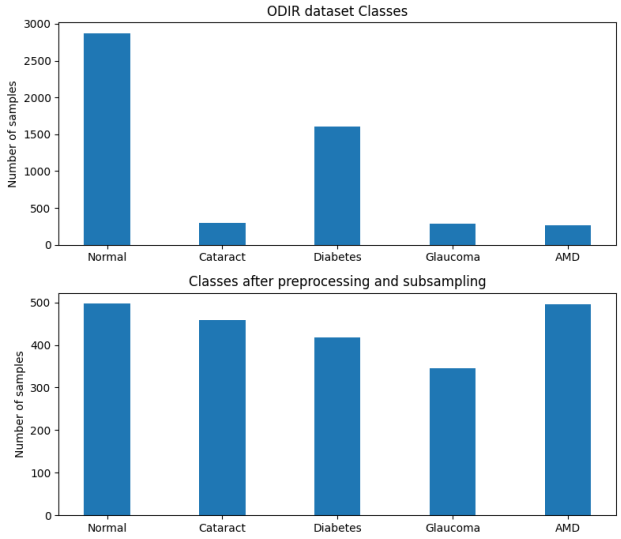

Figure 3: Solving the imbalance between the classes of the dataset

this problem I decided to keep only the images with a singular diagnosis that agreed both with the description of the diagnosis and the label that it was already assigned. All the remaining images were not used in this assignment.

4 Methodology

As mentioned before, I chose to work with the pretrained state-of-the-art model InceptionV3 [12]. Next, I will briefly explain its architecture and the changes I had to implement to make it usable for my classification problems.

4.1 The InceptionV3 model

InceptionV3, a refined iteration of the Inception architecture, was introduced to strike a balance between computational efficiency and high performance. Originating from the 2015 paper "Rethinking the Inception Architecture for Computer Vision," this model incorporates several strategies to maintain computational advantages while enhancing adaptability for diverse applications.

The guiding principles of Inception v3 include:

1. Factorized Convolutions: By employing factorized convolutions, the model trims down computational demands, optimizing efficiency without compromising overall performance.

2. Smaller Convolutions: The substitution of larger convolutions with smaller ones speeds up training. For instance, using two 3×3 filters instead of a single 5×5 filter reduces parameters, promoting efficient training.

3. Asymmetric Convolutions: Inception v3 introduces asymmetric convolutions, allowing a 3×3 convolution to be replaced by a combination of 1×3 and 3×1 convolutions. This technique effectively manages the network's parameter count.

4. Auxiliary Classifier: Unlike GoogLeNet, where auxiliary classifiers were employed for deepening the network, Inception v3 leverages an auxiliary classifier as a regularizer. This small CNN, inserted between layers during training, contributes its loss to the main network loss.

5. Grid Size Reduction: Addressing computational bottlenecks, grid size reduction is achieved through efficient pooling operations, maintaining computational efficiency while reducing the grid size.

A high-level diagram of the model is shown in the following screenshot: InceptionV3 has proven itself

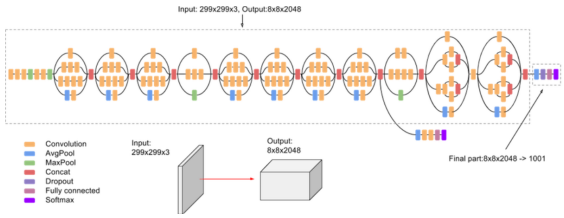

Figure 4: InceptionV3 Architecture

in image recognition by achieving 78.1% accuracy on the ImageNet dataset solidifying its position as one of the most comprehensive and efficient solutions in the realm of computer vision.

4.2 Modification for binary model

In developing the four binary classification models for detecting cataract, diabetic retinopathy, AMD, and glaucoma, as discussed earlier, I extended the InceptionV3 architecture. To make binary classification possible, I omited to top layer of the pretrained model and introduced a flatten layer to simplify the multi-dimensional output from InceptionV3, followed by a Dense layer of one unit with a sigmoid activation function. The flatten layer helps in converting the complex hierarchical features extracted by the model into a one-dimensional array, ensuring compatibility with the subsequent Dense layer designed for binary classification. After experimenting with various configurations with even more layers, this simple addition proved the most effective in achieving the best performance. Furthermore, the weights of the InceptionV3 model that were pretrained on the ImageNet dataset were set as non-trainable, ensuring the model leverages pre-existing knowledge while fine-tuning for the specific task of identifying ocular diseases.

Compile Parameters	**Fit Parameters**
optimizer='adam'	Batch size=32
loss='binary crossentropy'	epochs=15
metrics='accuracy'	callbacks: val_accuracy

Table 1: Configuration of parameters of Binary Model

4.3 Modification for Multi-label model

For the multi-label classification model that was used to classify between pictures with singular diagnosis with labels; normal, cataract, diabetic retinopathy, AMD, and glaucoma, I, again, extended the InceptionV3 architecture by incorporating specific layers at the top. After removing the previous top layer, I added a Flatten layer followed by two distinct Dense layers. The addition of Flatten() serves to convert the 3D output from the convolutional layers into a 1D array, enabling the transition to the dense layers that will follow. The Dense(1000, activation='relu') layer introduces a deep and non-linear transformation, capturing intricate patterns in the data, while the final Dense(5, activation='softmax') layer outputs probability scores for each of the five classes. Picking Softmax as the final layer activation function of the top layer is of crucial importance for multi-label classification tasks. We expect the output of the model to be propabilistic scores and I will keep the argmax of that prediction vector as my final prediction label. As before the previous weights were kept non-trainable. To further optimize the training, early stopping and a checkpoint mechanism were implemented, checking for each epoch the validation set accuracy and ensuring model convergence while also preventing overfitting.

Compile Parameters	**Fit Parameters**
optimizer='adam'	Batch size=32
loss='categorical crossentropy'	epochs=15
metrics='accuracy'	callbacks: val_accuracy

Table 2: Configuration of parameters of Binary Model

5 Explainability through visualization

Neural network models are generally referred to as being opaque. This means that they are poor at explaining the reason why a specific decision or prediction was made. One of the most prominent neural networks that are exploited a lot in the InceptionV3 model are Convolutional Neural Network. Convolutional neural networks are designed to work with image data, and their structure and function suggest that should be even less inscrutable than other types of neural networks. In this section I will describe my effort to describe and visualize the complex procedures hidden inside those blackbox models.

Filters Convolutional filters, also known as kernels, play a very important role in detecting specific patterns or features within input data. In this application, where our data is in the form of RGB images, filters should hopefully identify edges, corners,

or textures. In deep learning, these filters' weights are learned automatically through training, with the process of backpropagation, on extensive datasets, in our case the imagenet dataset, and the outcomes of applying these filters are referred to as activation maps or feature maps.

Convolutional filters are often small linear filters of size usually between 1x1 and 5x5. In the case of InceptionV3, a noteworthy observation is that different sizes of filters are used throughout its architecture compared to other state of the art models that prefer to utilize only one shape of filters.

One important architectural consideration in CNNs is to ensure that the depth of a filter matches the depth of the input, corresponding to the number of channels. For instance, an input image with three channels (red, green, and blue) requires filters with a depth of three. In the next image I visualize the six first filters of the first convolution layer of the inceptionV3 model. Each layer is of depth 3 and represents the three color channels of the input image. Upon visualizing the filters of the first cond2d

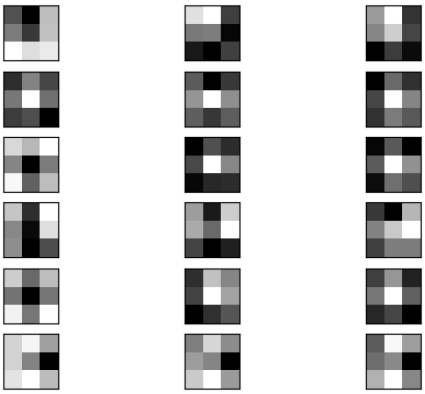

Figure 5: Some of the filters of the first convolution layer

layer, some patterns emerge. The filters exhibit some similarities across all three channels (rows 2 and 6) or similarities across two of the channels (rows 1, 3, 4 and 5). Dark squares indicate small weights and light squares represent large weights.

While visualizing a subset of filters can be a way to obtain some initial insights, examining all filters in a single image quickly becomes impractical as model complexity increases. For instance, if we want to visualize the filters of the second convolutional layer which contains 32 filters, each with 32 channels, it would require an overwhelming number of subplots. That is why in a model as complex as InceptionV3 visualization of the filters of its convolution layers is not a practical way to understand the way our model makes its decisions.

5.1 Activation Maps

Another way to provide insight into a model's internal representation can be gained from the feature maps resulting from applying filters to input images or feature maps from prior layers. These activation maps, also known as feature maps, encapsulate the outcomes of filter applications to inputs, be it the original image or another feature map.

Visualizing a feature map for a specific input image serves the purpose of understanding which features are detected or preserved in these maps. The expectation is that feature maps closer to the input reveal finer details, while those closer to the output capture more general features.

To explore feature map visualization, I chose a random image of the dataset as input for my multilabel model. The image below showcases all 32 feature maps captured in the first convolution layer.

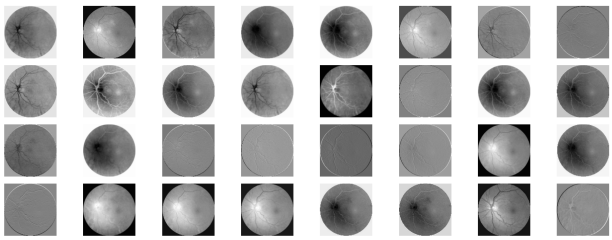

Figure 6: The feature maps of the first convolution layer for a given provided input image

Upon inspecting the result of applying filters in the first convolutional layer, numerous versions of the eye emerge, each emphasizing on different features. Some highlight the veins or the pupil while others

focus on the background so we can make an effort to understand what the ensemble of hidden units is learning each time.

5.2 GradCam

A third and possibly the most useful way to add explainability to our deep learning AI model is with the assistance of the Grad-CAM algorithm [10]. Grad-CAM, short for Gradient-weighted Class Activation Mapping, enhances interpretability in neural networks by producing a coarse localization map that highlights crucial regions for predicting a target concept. This can be particularly insightful for tasks such as this project where understanding the model's focus is valuable.

This technique improves upon the previous two methods of visualization. Despite its complexity, the output is intuitively comprehensible. At a high level, the process involves taking an image as input and creating a model truncated at the desired layer for Grad-CAM generation, with fully-connected layers for prediction attached. The input is then passed through the model, and the layer output and loss are captured. Subsequently, the gradient of the output with respect to the model loss is computed. The relevant sections of the gradient, contributing to the prediction, are adjusted, resized, and rescaled to create a heat-map that can be overlaid onto the original image.

In the gradient calculations, the importance of all successive feature maps leading up to the last convolutional layer is captured. The resulting heat-map already gives us some insight on the area that the last convolution layer focussed on to classify the image.

Nevertheless, I decided to further investigate the effect of all the previous convolution layers. By averaging Grad-CAM heat-maps from all model layers, a more nuanced and precise heat-map is generated. In the context of this research project focused on disease classification in eye images, as opposed to object detection in a generic picture, the ultimate heat map does not exhibit distinctly high or low values that explicitly indicate the presence or absence of the disease. Thus, to help the visualization process I slightly increased the contrast of the heatmap image. The

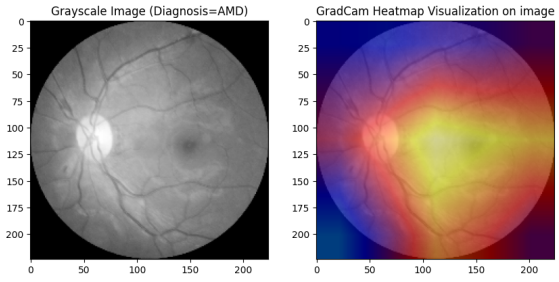

Figure 7: Grad-CAM heatmap of the final convolution layer

final image provides a clearer understanding of the model's decision-making process, shedding light on the specific areas that contributed to its predictions, namely the eye veins and the foggy and spotted middle part. A collaboration with a specialized ophthalmologist could provide helpful insights to validate if these areas are indeed the same that a doctor would focus on to diagnose the patient this eye belongs to.

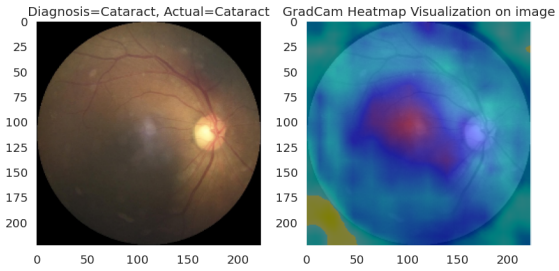

Figure 8: Average Grad-CAM heatmap

6 Results

This section will focus on presenting and commenting on the results of the different experiments implemented in this project assignment.

6.1 Binary Classification

As already mentioned, one of the main tasks executed in this assignment was the binary classification of the

target images. For each of the four main diseases (cataract, diabetic retinopathy, glaucoma and AMD) a separate training procedure was done to assess the accuracy of the InceptionV3 model to identify between images that show a normal eye and images that include that specific ocular disease.

Binary Classification	Accuracy	Loss
Normal-Cataract	0.9204	0.4470
Normal-Diabetes	0.8922	0.3813
Normal-Glaucoma	0.9173	0.2774
Normal-AMD	0.9528	0.1389

Table 3: Accuracy and loss metrics for the binary classification tasks

For the Normal-Cataract classification, the model achieved a loss of 0.4470 with an accuracy of 92.04%. The Normal-Diabetic Retinopathy classification yielded a loss of 0.3813 and an accuracy of 89.22%. Moreover, the model's capability in distinguishing between Normal and Glaucoma cases resulted in a loss of 0.2774 and an accuracy of 91.74%. Notably, in the instance of Normal-AMD classification, the model attained a minimal loss of 0.1390, showcasing an impressive accuracy of 95.28%.

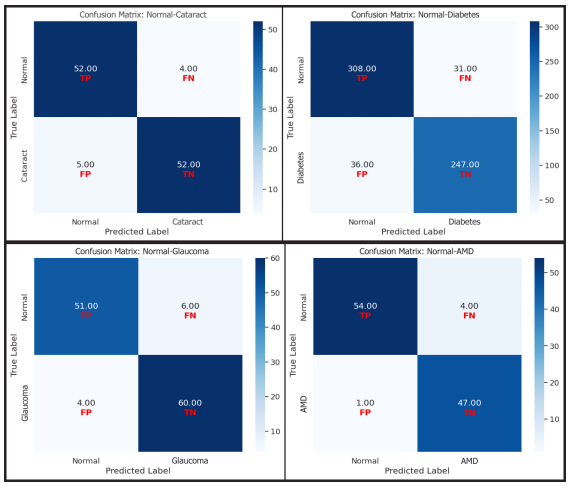

Figure 9: The Confusion matrices for the four binary classification tasks

Furthermore, the confusion matrices for each classification provide a detailed breakdown of the model's performance, depicting the number of true positives, true negatives, false positives, and false negatives. From these matrices we can see that the model in every binary classification task does not favor any of the diseases since the number of false positives and false negatives is always very close.

6.2 Multi-label Classification

In the multiclass classification task, the model exhibited overall promising results. The model achieved an average loss of 0.9209, indicating its ability to minimize predictive errors across all classes. The achieved overall accuracy of 76.13% shows that the model is adequately proficient in correctly classifying the diverse range of ocular diseases. Breaking down the performance metrics for individual classes, the precision, recall, and F1-score provide a more nuanced assessment. For the "Normal" class, the model achieved a precision of 66%, recall of 45%, and an F1-score of 53%. The "Cataract" class demonstrated high precision (94%), perfect recall (100%), and an impressive F1-score of 97%. In the "Diabetes" class, precision was 50%, recall was 59%, and the F1-score was 54%. For "Glaucoma," precision, recall, and F1-score were 84%, 92%, and 88%, respectively. Lastly, the "AMD" class showed a precision of 86%, recall of 94%, and an F1-score of 90%.

Class	Precision	Recall	F1-Score
Normal	0.66	0.45	0.53
Cataract	0.94	1.00	0.97
Diabetes	0.50	0.59	0.54
Glaucoma	0.84	0.92	0.88
AMD	0.86	0.94	0.90

Table 4: Evaluation metrics for each class for the multilabel classification task.

These detailed metrics provide valuable insights into the model's strengths and weaknesses in classifying different ocular conditions. The high precision and recall for certain classes, such as "Cataract" and "AMD," indicate the model's proficiency in accurately identifying these conditions. However, lower

precision and recall in other classes suggest potential areas for improvement, emphasizing the need for further optimization to enhance the model's performance across all categories.

Additionally, the training-validation loss and accuracy curves plotted over epochs demonstrate the learning dynamics of the model. Analyzing these curves helps to understand how well the model generalizes to new data and whether there are signs of overfitting or underfitting. These visualizations contribute to a comprehensive evaluation of the model's training process and performance across epochs.

Figure 10: Training and Validation Accuracy and Loss Graph

6.3 Multi-label Classification (Enhanced contrast)

Applying image augmentation techniques on an input image can sometimes enhance the classification model's accuracy. One of my plans was to experiment with changes on brightness and contrast on the eye pictures and document the corresponding metric changes. However, due to limited time and resources (I had utilized most of my computational units on Google Colab) I couldn't make an exhaustive grid search on those parameters. Nevertheless, one experiment that drew my focus involved tripling the contrast of the input images. The results revealed a noteworthy improvement in accuracy for the multi-label classification problem, with a recorded accuracy of 0.792, a noticeable upgrade from the previous 0.761. From the classification report, it is evident that while

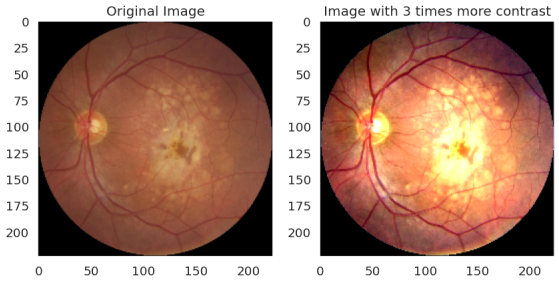

Figure 11: Original image and image with increased contrast

the evaluation metrics for correctly classifying AMD have decreased, the F1 Score for all the other diseases has shown a notable improvement.

Class	Precision	Recall	F1-Score
Normal	0.61	0.52	0.56
Cataract	0.98	1	0.99
Diabetes	0.63	0.59	0.61
Glaucoma	0.89	0.93	0.91
AMD	0.76	0.88	0.81

Table 5: Evaluation metrics for each class for the multilabel classification task.(Contrast=3)

Lastly, the training-validation loss and accuracy curves plotted over epochs show that the best results are reached in less epochs than for the previous non-augmented data. That is due to the early stopping that was mentioned before that deemed that the validation accuracy wont increase more further on, so it stopped the process without finishing all the 15 epochs.

7 Future work & Improvements

In future work and potential improvements, a comprehensive exploration could involve conducting a grid search to evaluate the impact of brightness and contrast and also other image augmentation techniques such as Frequency/Spatial Filtering or His-

Figure 12: Training and Validation Accuracy and Loss Graph for the high contrast experiment

togram transforms on accuracy, thereby fine-tuning this aspect for enhanced model performance. Additionally, considering the alternative approach of upsampling instead of downsampling and comparing the results could also provide valuable insights into the model's response to different data processing techniques.

Exploring more state-of-the-art models could also prove valuable insights as to which one is best suited for this specific task. Additionally, collaborating with a specialized medical professional to assess the importance and accuracy of explainability, particularly through the Grad-CAM visualizations could also prove extremely beneficial. This collaborative effort would validate whether the highlighted areas identified by the model align with a doctor's diagnostic observations, contributing to the interpretability and clinical relevance of the classification results. These approaches to future research endeavors could hold the potential to further elevate the robustness and practical applicability of this classification framework.

8 Conclusion

In conclusion, the InceptionV3 model showcased an impressive performance, achieving accuracies approaching 95% for Binary and 79% for Multi-label Classification on the ODIR dataset. Notably, optimizing image contrast has proven that it can be a valuable strategy for enhancing model results. Moreover, the utilization of visualizations such as filters, activation maps, and GradCAM has provided some insights, offering a glimpse into the intricate workings of the multi-layer CNN model and contributing to our understanding of its 'Black box' nature. These findings collectively underscore the effectiveness and interpretability of the InceptionV3 model in the context of diverse classification tasks.

References

[1] Ocular disease intelligent recognition odir-5k.

[2] Varun Gulshan, Lily Peng, Marc Coram, Martin C. Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, Ramasamy Kim, Rajiv Raman, Philip C. Nelson, Jessica L. Mega, and Dale R. Webster. Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs. *JAMA*, 316(22):2402–2410, 12 2016.

[3] Junjun He, Cheng Li, Jin Ye, Yu Qiao, and Lixu Gu. Self-speculation of clinical features based on knowledge distillation for accurate ocular disease classification. *Biomedical Signal Processing and Control*, 67:102491, 2021.

[4] Kai Hu, Zhenzhen Zhang, Xiaorui Niu, Yuan Zhang, Chunhong Cao, Fen Xiao, and Xieping Gao. Retinal vessel segmentation of color fundus images using multiscale convolutional neural network with an improved cross-entropy loss function. *Neurocomputing*, 309:179–191, 2018.

[5] Cecilia S. Lee, Ariel J. Tyring, Nicolaas P. Deruyter, Yue Wu, Ariel Rokem, and Aaron Y. Lee. Deep-learning based, automated segmentation of macular edema in optical coherence tomography. *Biomed. Opt. Express*, 8(7):3440–3448, Jul 2017.

[6] Zhixi Li, Yifan He, Stuart Keel, Wei Meng, Robert T. Chang, and Mingguang He. Efficacy

of a deep learning system for detecting glaucomatous optic neuropathy based on color fundus photographs. *Ophthalmology*, 125(8):1199–1206, 2018.

[7] Xianjing Meng, Xiaoming Xi, Lu Yang, Guang Zhang, Yilong Yin, and Xinjian Chen. Fast and effective optic disk localization based on convolutional neural network. *Neurocomputing*, 312:285–295, 2018.

[8] Clément Playout, Renaud Duval, and Farida Cheriet. A novel weakly supervised multitask architecture for retinal lesions segmentation on fundus images. *IEEE Transactions on Medical Imaging*, 38(10):2434–2444, 2019.

[9] Abhijit Guha Roy, Sailesh Conjeti, Sri Phani Krishna Karri, Debdoot Sheet, Amin Katouzian, Christian Wachinger, and Nassir Navab. Relaynet: retinal layer and fluid segmentation of macular optical coherence tomography using fully convolutional networks. *Biomed. Opt. Express*, 8(8):3627–3642, Aug 2017.

[10] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016.

[11] J. Staal, M.D. Abramoff, M. Niemeijer, M.A. Viergever, and B. van Ginneken. Ridge-based vessel segmentation in color images of the retina. *IEEE Transactions on Medical Imaging*, 23(4):501–509, 2004.

[12] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[12] [1] [7] [3] [9] [5] [8] [4] [11] [2] [6] [10]