Travail d'Etude et de Recherche

Glaucoma diagnosis by eye fundus processing

Maximilien Wemaere
Encadré par **Nicolas Lermé**

octobre 2021-mai 2022

Abstract

Glaucoma is one of the most common causes of blindness in the world and it is important to detect it as early as possible to prevent its progression. Fundus image processing is a fast growing field because new machine learning methods are very promising to face this very complicated problem. Indeed these images are very difficult to study with conventional tools. This paper is the report of a study and research work within the framework of a first year master of electrical and computer engineering. We will attempt to code a program capable of diagnosing glaucoma using a fundus image dataset. We will first try a conventional method of feature extraction and then look at a method using a convolutional neural network

Contents

I Introduction

I.1 Medical Introduction

Strictly speaking, glaucoma is a group of disease which have in common that they damage the optic nerve. It all starts with a problem with the trabecular system, which allows the proper drainage of the aqueous humor, the liquid that fills the majority of the eyeball. Due to a partial occlusion of the trabecular system, the aqueous humor cannot be evacuated efficiently, and it condenses in the eyeball, increasing the pressure inside it. This pressure thus crushes the eyeball on the optic nerve, damaging it, which has for consequence a progressive reduction of the field of vision until the blindness.

Thus the diagnosis of glaucoma can be made using different data: pressure, field of vision, the eye-fundus. It is this last examination that interests us here: as its name indicates, it is a photo of the inside of the eye taken by ophthalmoscopy.

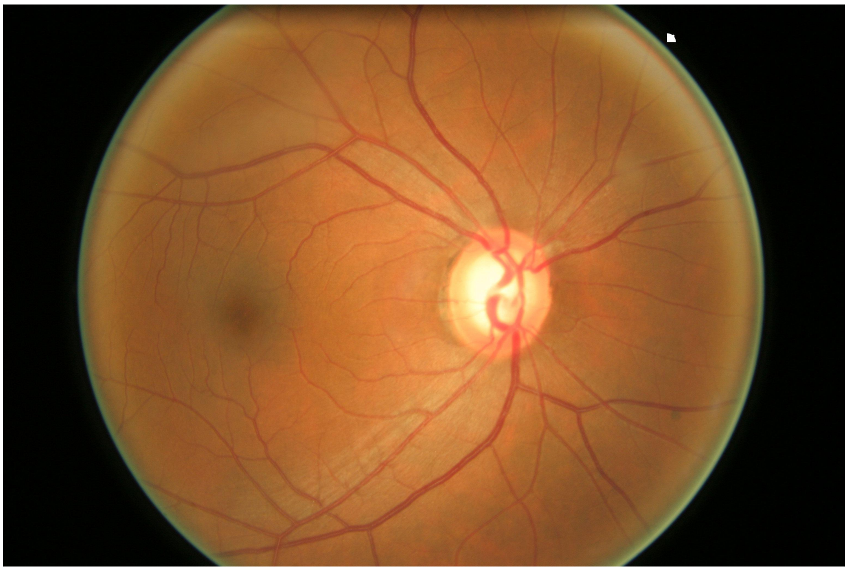


Figure 1: An eye fundus

What interests us here is the optic disc (OD) and the optic cup (OC). In fact, the ratio of one to the other gives us the: cup to disc ratio (CDR). And this number is very important because it shows the crushing of the optic nerve on the eyeball.

The closer the CDR is to 1, the more likely the patient is to have glaucoma. It is therefore a very important data for ophthalmologists who diagnose glaucoma. O.65 is usually the threshold value for suspecting glaucoma. However, this data is not sufficient to diagnose glaucoma: doctors use other information such as eye pressure.

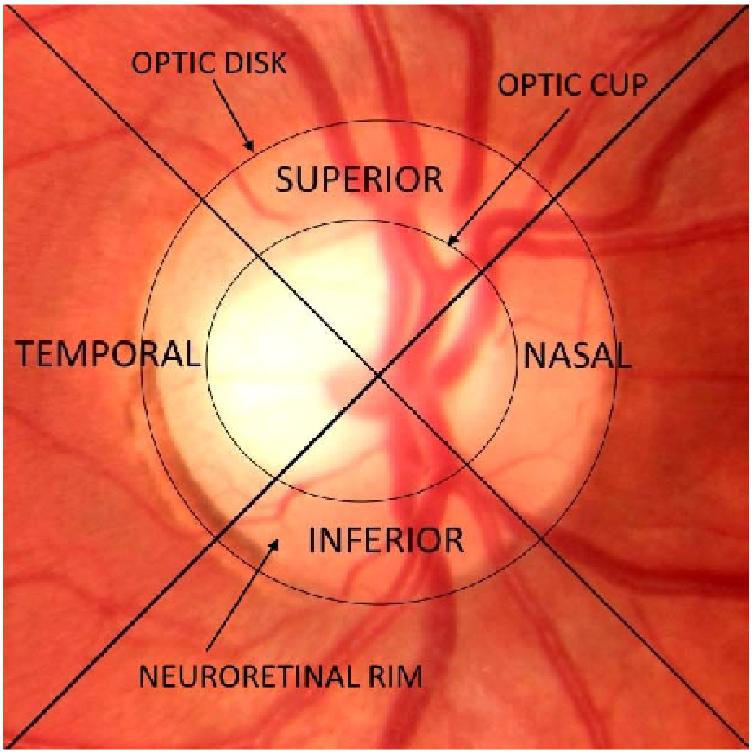


Figure 2: The optic nerve on a fundus

I.2 Dataset

The dataset used is called ORIGA (Online Retinal fundus Image database for Glaucoma Analysis). It contains 650 images of eye fundus divided into two classes: glaucoma positive (GP) and glaucoma negative (GN). There are 482 GN and 168 GP. These images were collected by the Singapore Eye Research Institute in the hospitals of Singapore between 2004 and 2007, the dataset was then published in 2010. The images have a size of 3072x2048. 420 images are used as a training set and 130 image are used as a test set. In the codes this second set is also called validation set because it is named as such in the dataset but we will use it as test set.

I.3 Objectives

The objectives of this TER are therefore: In part II to extract the OC and OD in order to calculate the CDR. Then to carry out a quick statistical study in order to verify that this data is not sufficient for the diagnosis of glaucoma. Then in part III, we will try to refine our results by proposing a diagnostic algorithm with a convolutive neural network (CNN).

I.4 Review of the state-of-the-art

First, [2] allows us to understand the mechanism of glaucoma and its observable effects on a fundus. As the crushing of the optic nerve convey by the interesting data that is the CDR.

Secondly, we used the paper provided by [3] in order to learn about the dataset. A method for detecting the region of interest (which will not be used) is described as well as the list of interesting features to extract.

[6] performs a statistical study of glaucoma diagnostic data to create a machine learning model for diagnosis. Many of the data used are not available to us (such as eye pressure), but the statistical study on the OC and OD is interesting.

Then, [1] provides a very detailed presentation of the problem by introducing the context very rigorously, the different methods already used and the results obtained.

[7] briefly explains the most common method used for this problem as well as a listing of other promising approaches

[4] details their approach to extracting OC and OD by K-means clustering. This paper will be used as a basis for part II

Another quite similar method is presented by [5] using a Gabor transform. This one will be tested but not presented in this paper because we did not succeed in obtaining satisfactory results.

[11] continues the explanations of [4] on the extraction of features by clustering by using this time an unsupervised method allowing not to have to choose beforehand the number of cluster

Finally [8], [9] and [10] each detail the use of a CNN to solve this problem of glaucoma diagnosis. While the architectures they use are similar, the hyperparameters studies they perform are complementary.

II Classification by features extraction

In this part we will try to calculate the CDR of the fundus by extracting the OC and OD

II.1 Plan of action

For the calculation of the CDR, we will use a similar procedure to that of [4] but adapted to the ORIGA dataset.

- Locate regions of interest (ROI) on all images and create a new dataset of reduced images
- Segmenting OC and OD with a K-means algorithm
- Recognize ellipses in order to identify OC and OD
- Compute the CDR

II.2 Data reduction

The first step is to locate the ROI of the images. Here it is simple since it is the optic nerve head, containing the OC and OD.

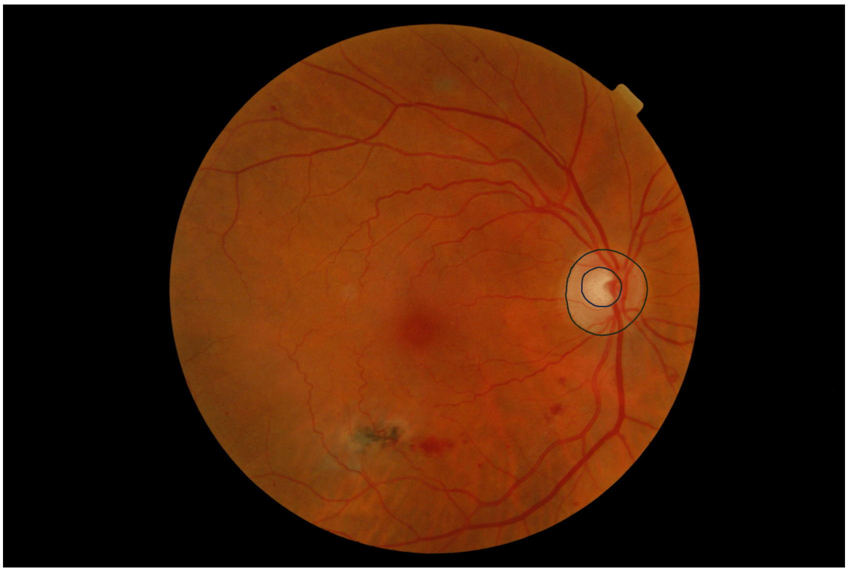


Figure 3: optic nerve highlight on a fundus

The first intuitive technique is after transforming the image into gray level, to make batches of 64x64 and keep the brightest. Then we keep only an area of 601 by 601 pixels centered on the brightest batch. The method works very well. On the 650 images, only 2 are not reduced on the ROI.

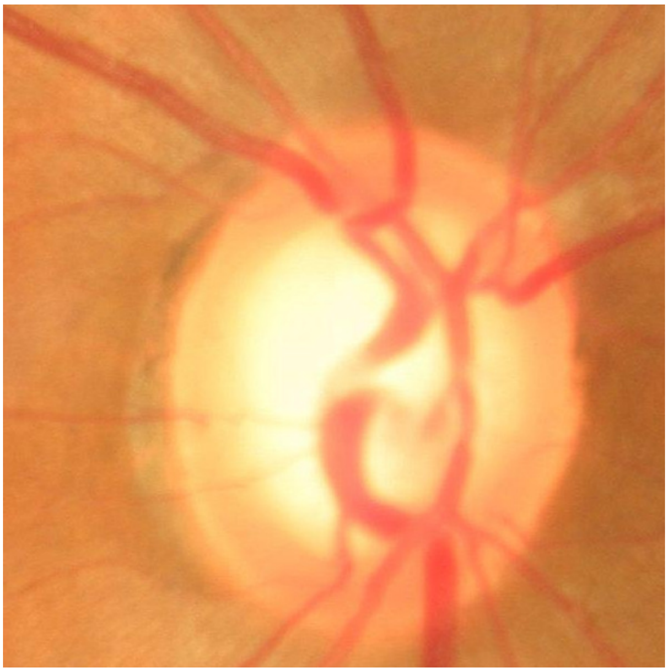


Figure 4: Reduced eye fundus

Concerning the computation time, for this step it is very short, because the algorithm is very simple. moreover the Julia language has been used here, it is a compiled language and not interpreted like python, which has therefore a much lower compilation time

In order not to perform this step every time, a new dataset is created with only the reduced images

II.3 Segmentation with K-means

In order to extract the OC and OD, several methods and algorithms were tested. We can not detect the features that interest us directly, because the border between the OC and OD areas is too blurred. Finally we had the most satisfactory results with the method used by [4]. It consists in the segmentation of the image into different classes. The classes are created by k-means clustering.

K-means is an algorithm that allows to classify data in unsupervised learning by clustering. For an image, it gathers the pixels in different classes, thus segmenting the image. However, the number of classes must be given as input to the algorithm. The k-means method is one of many clustering algorithms but it is quite common and efficient for a wide range of problems, so it is the one that has been used.

The indication that allows to separate the OC and OD is the brightness, that's why before clustering, the image is converted from RGB to gray level.

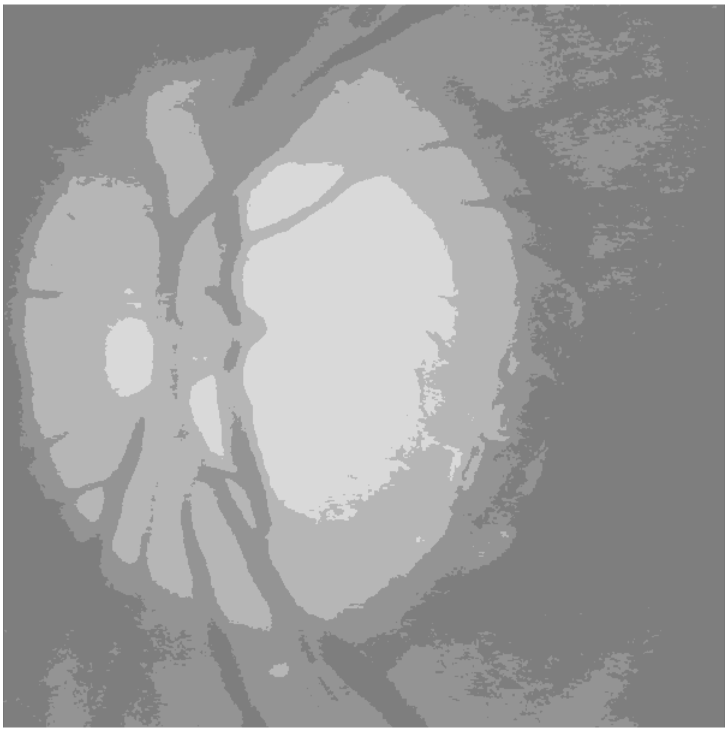


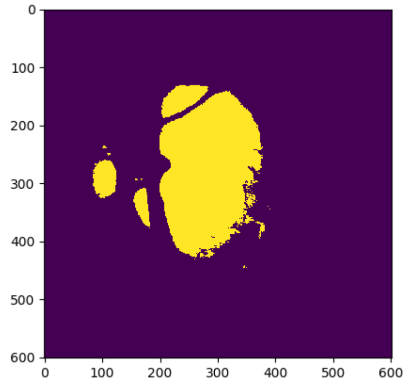
Figure 5: Eye fundus segmented

The algorithm has been tested with several numbers of classes, but the algorithm gives the best results for the majority of cases with 4 different classes as seen in [11].

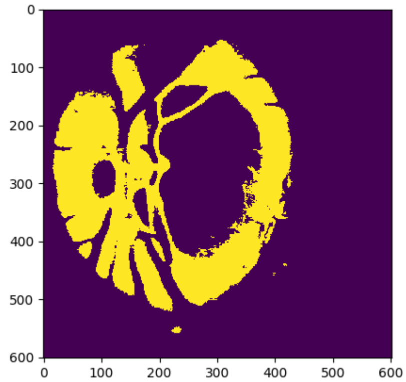
We therefore have results that separate the OC and OD quite well from the rest of the reduced image.

Finally, to know which class is associated with the OC and the OD, it is enough to associate the class that has the highest luminosity to the OC then the second to the OD.

We recover two binary images (recovered in .npy). One giving the segmentation of the OC and the other, those of the OD



(a) Optic cup segmented



(b) Optic disc segmented

II.4 Ellipse detection

This step is more delicate, indeed the majority of papers do not speak about it and those which speak about it quickly evoke a Hough transform.

First, the Hough transform to detect circles has been tested, spending time to test all parameters. However, the results are very dependent on the shape of the OC and OD. Which are most of the time not really spherical but rather elliptical, which makes the Hough circle detection inefficient.

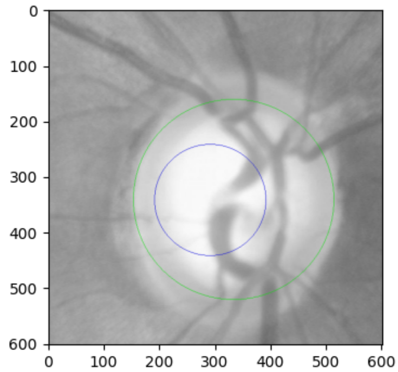


Figure 7: Wrong detection of circles

As most of the OC and OD borders look like ellipses, an ellipse detection by Hough was used. The CDR is thus the ratio of the great length of the ellipse of the OC to that of the OD.

The method gives very satisfactory results. The only small problem of this method is the computation time. With my laptop the algorithm run about 30h to detect the ellipses on the training set thus of 520 images (1040 ellipses to detect).

Indeed to detect an ellipse, it is necessary to identify 5 parameters: the two coordinates of the center, the two lengths of the half axes and the angle of the ellipse. With the Hough transform we have a 5 dimensional space, so the number of possible ellipses is very high, and so is the computation time. Nevertheless, the results obtained are those expected

In order to reduce the calculation time, a second, less computationally intensive method has been developed during this TER. The algorithm is the following:

- Keep only the edges of the image (OC or OD) by applying a Robert filter.
- Keep only the outer edges with a handmade function
- Collect the lengths of the chords of the ellipse by calculating the difference between the outer contours on each pixel line of the image
- With a polynomial regression on the lengths of the chords, estimate a function that links them
- Take the max of this function as the value of the long length of the ellipse

Thus this method does not really detect an ellipse but only its long length. Moreover it is necessary that the ellipse has its long length horizontal to estimate it well (small rotations do not disturb too much the estimation).

This is not a problem in our case because all the ellipses to be detected are vertical or have small angles to the vertical. There is only to transpose the images.

To generalize this method it would be necessary to perform different rotations of the image to find the one that maximizes the long length.

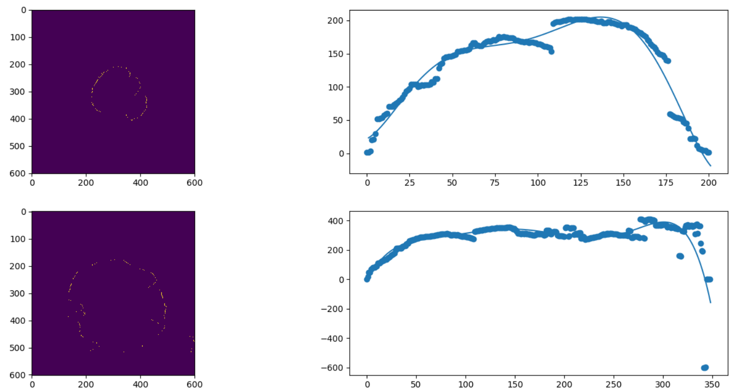


Figure 8: Length of the rope approximated by a function for the OC and the OD

The results of this method are less satisfactory, as explained in the following subsection.

II.5 Computation of CDR and statistical study

The cdr is calculated by the ratio of the long length of the OC to that of the OD.

For the method using Hough, we obtain for 520 images, the following values:

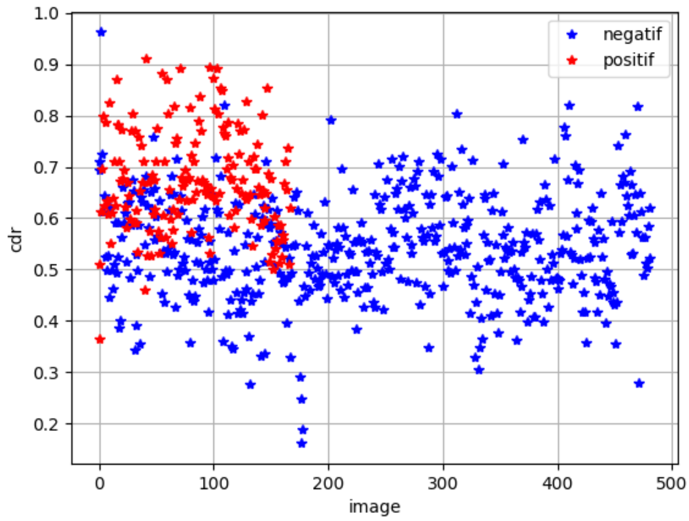


Figure 9: Values of the CDR

For more visibility we plot the probability density according to the CDR of glaucoma negative and positive images:

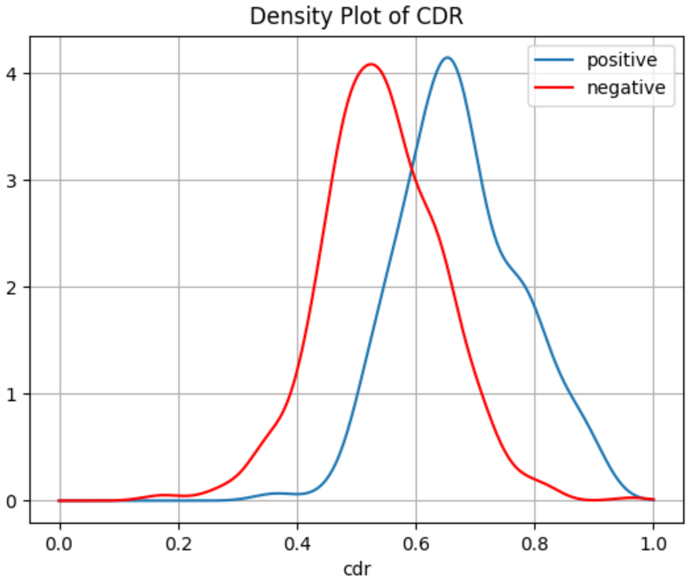


Figure 10: Density plot of CDR with Hough method

Therefore, there is a high error rate no matter which threshold value is chosen to classify an eye as negative or positive. However it is expected as in [6]. Moreover such results have been found with this dataset in [3].

In order to optimize the success rate of the diagnosis it is necessary to choose the best threshold. To do this, we calculate the false positive and false negative rates and we plot the success rate according to the chosen threshold. We obtain the following curve:

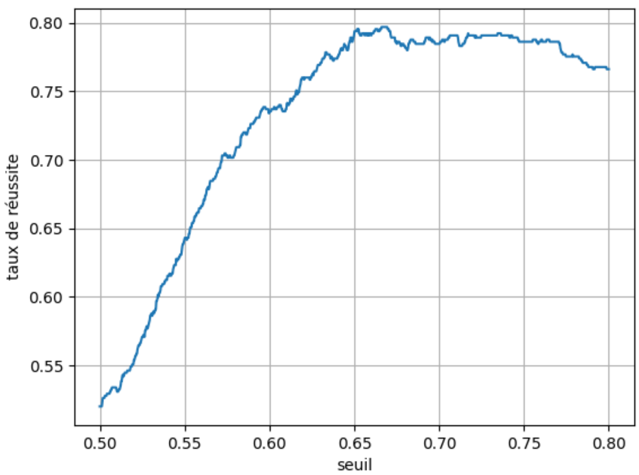


Figure 11: Evolution of the accuracy depending on the threshold

Thus, the curve is maximum around 0.65 which is effectively the value chosen by doctors to diagnose glaucoma (if not the most effective threshold value which is rather 0.67 but this is because the dataset is not very large).

Thus with a threshold of 0.65 we have a false positive rate of 0.12863070539419086, a false negative rate of 0.43452380952380953. For an error rate of 0.2076923076923077 The false negative rate is very high and this is what is annoying for a real diagnosis.

For a threshold at 0.58 which is the value of CDR where the two density curves cross we have a worse error rate: 0.2692307692307692 as well as a lower false positive rate: 0.29045643153526973 but a much better false negative rate: 0.20833333333333334 which is preferable for a diagnosis.

the sources of errors come from all three steps:
-wrong location of the ROI (2 images only)
-poor segmentation (less than 10 images)
-wrong fitting of the ellipse (about 20 images)
These estimates are made by comparison with a list of the CDR measured by hand find on Kaggle.

For the chords method, the results are much less satisfactory with a false positive rate of 0.21243523316062177 a false negative rate of 0.5149253731343284, for a success rate of 0.7096153846153845

Nonetheless the method gives some hopes, because by looking at the density of probabilities, we can see that they are sufficiently separated to be distinguishable:

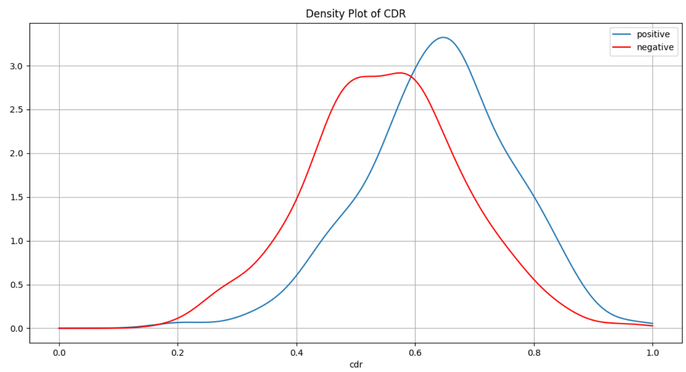


Figure 12: Density plot of CDR with the new method

As we expected, the CDR is not sufficient to obtain an accurate diagnosis, and the accuracy found (around 70) is what was expected.

III Classification by CNN

In this part we will try to solve this problem of glaucoma diagnosis with a fundus but using this time a convolution neural network (CNN).

The problem is thus completely different, because it is no longer a question of determining the best feature extraction strategy, but of finding the architecture and parameters of the neural network that will maximize its accuracy

The neural networks are realized with the TensorFlow library. The science of neural networks being a domain quite distant from conventional image processing, this part has been treated as a bonus at the end of the TER. The search for the best CNN was not done thanks to a total understanding of ANN, but rather by trial and error. Indeed, understanding the way of the CNN to answer the problem is very complex, especially for this very difficult problem, and would require a TER entirely devoted to this method.

Moreover the networks were coded trained on Google Colab, so only one network could run at a time and for a limited time

III.1 Second attempt: CNN with segmentation as input

First, we decided to continue our previous work by using the results of the OC and OD segmentations as input.

A contextualization CNN (C-CNN) was first realized inspired by [9], it consist in two CNN working in parallel and joining on a dense layer. But unlike [9] the first network took as input the segmentation of the OC and the second the segmentation of the OD.

However, because of the limited technical means available (google colab) it was not possible to train two CNN at the same time. So we continued our research with simpler methods.

We tried by joining the two images of the OC and OD segmentation and feeding the CNN with this images. But the input is too large and the CNN only converged to bad solutions whatever its layer combination or training parameters. We therefore returned to the paths traced by the entire state of the art reviewed, by feeding the CNN with the fundus directly.

III.2 Second attempt: CNN with ROI as input

We continue the tests with reduced images as input, i.e. only the ROI. The image is reduced from 600x600x3 to 224x224x3 in order not to have a too big CNN. The chosen architecture is classical: two convolution layers with two pooling layers. Then two dense layers, one of 100 neurons and the last of two neurons, one for each class.

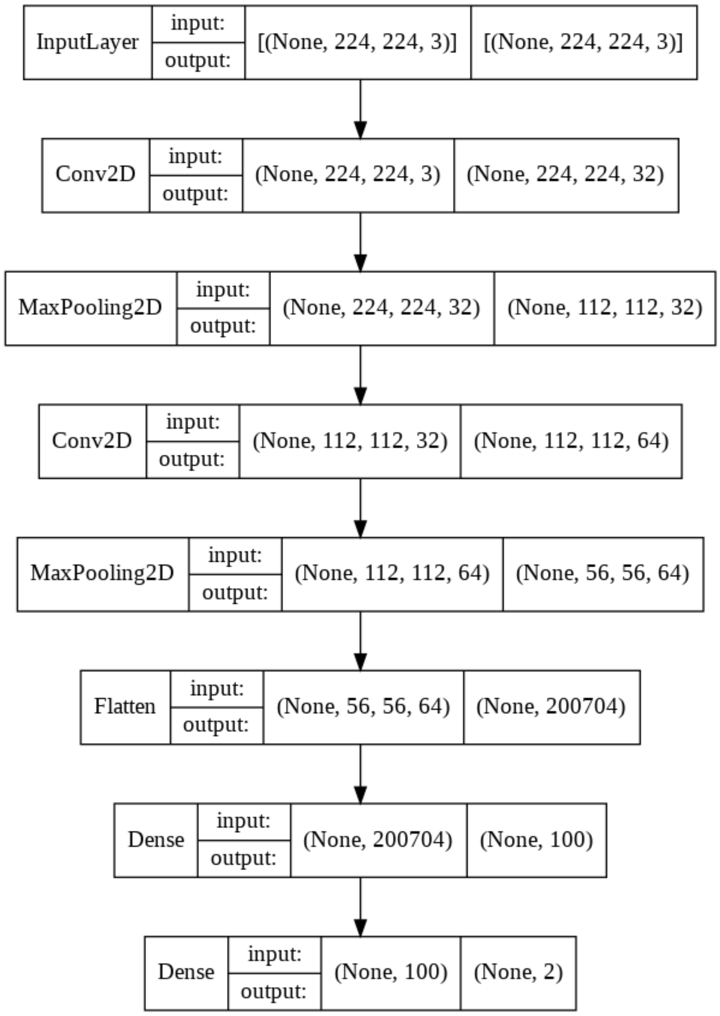


Figure 13: Architecture of the CNN

With a batch size of 10, an Adam optimizer with a learning rate of 0.0001 the first tests all converge to a situation where the network classifies all images in the NEGATIVE class, as shown below the confusion matrix.

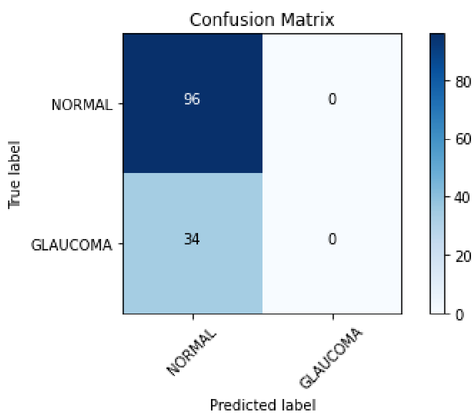


Figure 14

This situation is the point of convergence of all the first networks tested, and this by testing with a large number of different layer combinations. By changing the number of epochs or by performing a data augmentation of the images of the POSITIVE class

Thus, after invariance of the error despite all these modifications, the problem could only come from the back-propagation. This is why we now change the learning rate of the Adam optimizer to 0.00001 and with a batch size of 3 and 500 epochs, we have the following results: an accuracy of 0.6 but it is encouraging because not all images are diagnosed as negative, as shown by the confusion matrix. We must therefore continue in this direction.

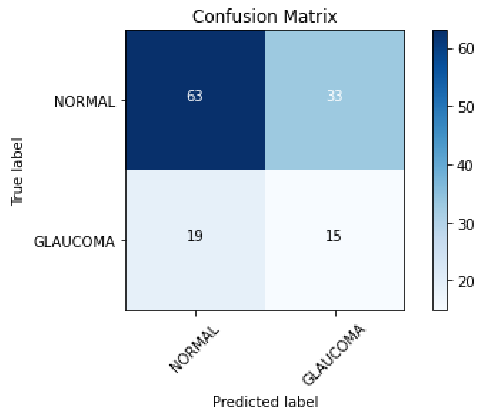


Figure 15

Thus with a longer learning period, with 3000 epochs we can obtain more satisfactory results: an accuracy of 0.8769231 , a false negative rate of 0.11764706 and a false positive rate of 0.21875. the low false negative rate is particularly encouraging.

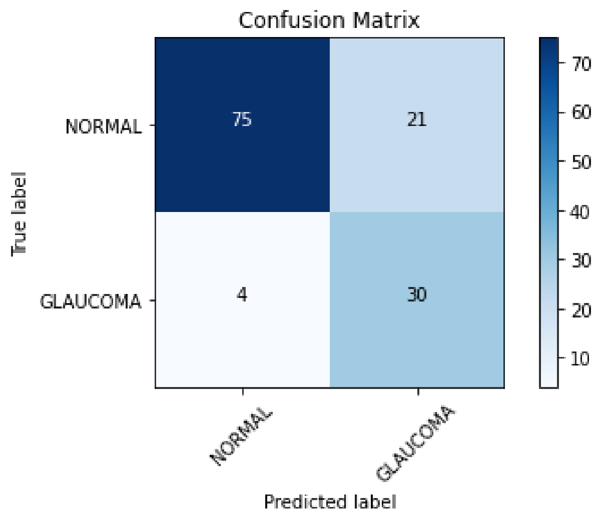


Figure 16

III.3 Conclusion

These results are therefore very satisfactory and we could think that we have reached our goal. However, in the practical context of medicine where diagnostic errors are very costly, this classifier is not at all precise enough. Research in this field is trying to achieve a diagnostic success rate close to 100

Moreover, if the classifier is efficient on this dataset, it would be interesting to test it on other datasets because given the high number of epochs that we realized during the training, it is very likely that there was overfitting and thus the network would be efficient on images very similar to the ORIGA dataset. Indeed in this dataset the images are taken in very similar conditions with the same devices.

It is then complicated to draw conclusions about how the network was able to perform the diagnosis. It would however be interesting to observe the convolution filters performed by the network to try to discover which features are interesting but once again this information will not be easy to obtain.

IV General Conclusion

This TER was for me an opportunity to realize a first small research experience.
I was able to familiarize myself with many image and data processing techniques seen in class, such as the Hough transform or the K-means segmentation.
I was also able to confront myself for the first time with a really complicated deep learning problem.
On a practical level I was able to get more familiar with the python language for data processing. In particular with the OpenCV, SKLearn, SKImage, Keras and Tensorflow libraries.
Finally I have discovered what it was to work in a domain at the interface between two domains. Here image processing and medicine.

If I have not really brought any great novelty in the field of glaucoma diagnosis by fundus treatment, I have really been able to take hold of this subject and to dive deeply into it up to the most recent discoveries thanks to a long work of review of the state of the art.

Github repository:
`https://github.com/MxWmr/TER_eye_fundus_classification.git`

References

[1] Saba T, Bokhari STF, Sharif M, Yasmin M, Raza M. Fundus image classification methods for the detection of glaucoma: A review. Microsc Res Tech. 2018; 1–17. https://doi.org/10.1002/jemt.23094

[2] Item 82 (item 240) : Glaucome chronique,Collège des Ophtalmologistes Universitaires de France (COUF),2013

[3] Zhang Z, Yin FS, Liu J, Wong WK, Tan NM, Lee BH, Cheng J, Wong TY. ORIGA(-light): an online retinal fundus image database for glaucoma analysis and research. Annu Int Conf IEEE Eng Med Biol Soc. 2010;2010:3065-8. doi: 10.1109/IEMBS.2010.5626137. PMID: 21095735.

[4] Amed Mvoulana, Rostom Kachouri, Mohamed Akil. Fully automated method for glaucoma screening using robust optic nerve head detection and unsupervised segmentation based cup-to-disc ratio computation in retinal fundus images. Computerized Medical Imaging and Graphics, Elsevier, 2019, 77, pp.101643. 10.1016/j.compmedimag.2019.101643. hal-02295007

[5] Mr. Prasad N. Maldhure , Prof. V. V. Dixit "Glaucoma Detection Using Optic Cup and Optic Disc Segmentation", International Journal of Engineering Trends and Technology (IJETT), V20(2),52-55 Feb 2015. ISSN:2231-5381. www.ijettjournal.org. published by seventh sense research group

[6] Kim SJ, Cho KJ, Oh S (2017)Development of machine learning models for diagnosis of glaucoma. PLoS ONE 12(5): e0177726.

[7] A. Rebinth and S. M. Kumar, "A Deep Learning Approach To Computer Aided Glaucoma Diagnosis," 2019 International Conference on Recent Advances in Energy-efficient Computing and Communication (ICRAECC), 2019, pp. 1-6, doi: 10.1109/ICRAECC43874.2019.8994988.

[8] Diaz-Pinto, A., Morales, S., Naranjo, V. et al. CNNs for automatic glaucoma assessment using fundus images: an extensive validation. BioMed Eng OnLine 18, 29 (2019). https://doi.org/10.1186/s12938-019-0649-y

[9] Chen, X., Xu, Y., Yan, S., Wong, D.W.K., Wong, T.Y., Liu, J. (2015). Automatic Feature Learning for Glaucoma Detection Based on Deep Learning. In: Navab, N., Hornegger, J., Wells, W., Frangi, A. (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. MICCAI 2015. Lecture Notes in Computer Science(), vol 9351. Springer, Cham.

[10] Mahum, R.; Rehman, S.U.; Okon, O.D.; Alabrah, A.; Meraj, T.; Rauf, H.T. A Novel Hybrid Approach Based on Deep CNN to Detect Glaucoma Using Fundus Imaging. Electronics 2022, 11, 26. https:// doi.org/10.3390/electronics11010026

[11] N. Thulasi and Dr. K. Thirunadana Sikamani, Early Detection of Glaucoma Based on Local Binary Pattern and Unsupervised Clustering Approaches Using Fundus Images Research Scholar, St. Peter's University, Chennai, India1 Professor Head, Department of Computer Science Engineering, St.2 Peter's College of Engineering Technology, Chennai, India