

MASTERS DEGREE FINAL PROJECT

Master's degree in Interdisciplinary and Innovative Engineering

**AUTOMATED CLASSIFICATION OF RETINOPATHY OF
PREMATURITY IN NEWBORNS**



Report and Annexes

Autor: Valentino Asole

Director: Raul Benitez

Ponent: Arnau Valls Esteve

Convocatòria: September 2021

Abstract

Retinopathy of Prematurity (ROP) is a disease in preterm babies with underdevelopment in retinal vessels. Early diagnosis of the disease is challenging and requires skilled professionals with very specific knowledge. Currently, in Spain, only a few hospitals have departments specialized in this pathology and, therefore, are able to diagnose and treat it accordingly.

This master project aims to develop the first preliminary instrument for the classification of the extent of Retinopathy disease. This tool has been built to be integrated into a diagnostic support platform to detect the presence of retinopathy and evaluate the sickness, providing insightful information regarding the specific image. This project also lays the base for the comparison between the clinical approach that the doctors use and the “black box” approach the Artificial Neural Network uses to predict the extent of the disease.

The developed algorithm is able to: segment ocular vessels using a U-Net Convolutional Neural Network; extract the critical features from the segmentation; and classify those features into ROP cases and ROP Plus cases by employing a range of different classifiers. The main features analyzed by the related specialists and thus selected are tortuosity and thickness of the vessels.

The segmentation Network achieved a global accuracy of 96.15%. The results of the different classifiers indicate a trade-off between accuracy and the volume of computed images. An accuracy of 100% was achieved with a Double Threshold classifier on 12.5% of the images. Instead, by using a Decision tree classifier, an accuracy of 70.8% was achieved when computing 100% of the images.

Resum

La Retinopatia de l'Prematur (ROP) és una malaltia que afecta els nadons prematurs mostrant-se com un subdesenvolupament dels vasos retinians. El diagnòstic precoç d'aquesta malaltia és un tot un repte ja que requereix de professionals altament qualificats amb coneixements molt específics. Actualment a Espanya, només uns pocs hospitals compten amb els equipaments especialitzats per al tractament i diagnòstic d'aquesta patologia.

Aquest projecte final de màster, té com a objectiu final desenvolupar una eina preliminar per a la classificació de l'extensió aquesta malaltia. Aquesta aplicació, ha estat dissenyada per a ser integrada en una plataforma de suport a la diagnosi de la Retinopatia i poder evaluar la malaltia, proporcionant informació detallada sobre les imatges analitzades. Aquest projecte, també estableix les bases per a la comparació entre l'enfocament clínic, que utilitzen els metges, i la naturalesa "Black-Box" natural de la Xarxa Neuronal Artificial per classificar l'extensió de la malaltia.

L'algoritme desenvolupat és capaç de: segmentar els vasos oculars utilitzant una xarxa neuronal convolucional U-Net; extreure les característiques representatives de la malaltia a partir de la segmentació; i classificar aquestes característiques en casos ROP i casos ROP Plus, mitjançant l'ús d'una gamma de classificadors. Les principals característiques analitzades són la tortuositat i el gruix dels vasos, indicadors de la malaltia emprats pels patòlegs experts.

La xarxa de segmentació ha obtingut una precisió global de l'96,15%. Els resultats dels diferents classificadors indiquen un trade-off entre la precisió i el volum d'imatges analitzades. S'ha obtingut una precisió de l'100% emprant un classificador de doble threshold en el anàlisi de l'12,5% de les imatges. En canvi, mitjançant l'ús d'un classificador "decision tree", s'ha obtingut una precisió del 70,8% analitzant el 100% de les imatges.

Resumen

La Retinopatía del Prematuro (ROP) es una enfermedad que afecta a los bebés prematuros mostrándose como el subdesarrollo de los vasos retinianos. El diagnóstico precoz de dicha enfermedad es un desafío ya que requiere de profesionales altamente capacitados con conocimientos muy específicos. Actualmente en España, solo unos pocos hospitales están dotados con los equipamientos especializados para el tratamiento y diagnóstico de esta patología

Este proyecto final de master, tiene como objetivo final desarrollar una herramienta preliminar para la clasificación de la extensión dicha enfermedad. Esta aplicación, ha sido diseñada para ser integrada en una plataforma de soporte al diagnóstico de la Retinopatía y evaluar la enfermedad, proporcionando información detallada sobre las imágenes analizadas. Este proyecto también sienta las bases para la comparación entre el enfoque clínico, que utilizan los médicos, y la naturaleza "Black-Box" natural de la Red Neuronal Artificial para clasificar la extensión de la enfermedad.

El algoritmo desarrollado es capaz de: segmentar los vasos oculares utilizando una red neuronal convolucional U-Net; extraer las características representativas de la enfermedad a partir de la segmentación; y clasificar estas características en casos ROP y casos ROP Plus, mediante el empleo de una gama de clasificadores. Las principales características analizadas son la tortuosidad y el grosor de los vasos, indicadores cauterizantes de la enfermedad empleados por los patólogos expertos.

La red de segmentación ha logrado una precisión global del 96,15%. Los resultados de los diferentes clasificadores indican un trade-off entre la precisión y el volumen de imágenes analizadas. Se ha obtenido una precisión del 100% empleando un clasificador de doble threshold en el análisis del 12,5% de las imágenes. En cambio, mediante el uso de un clasificador "decision tree", se ha obtenido una precisión del 70,8% analizando el 100% de las imágenes.

Acknowledgment

The first acknowledgment goes to Arnau Valls Esteve and Raul Benitez for not only providing the chance to develop this master thesis but also for the restless help provided during all those months.

The second goes to my family that helps me and supports me from Italy, making me always feel at home.

The third one, but no less important, goes to all my master mates that helped me and motivated me during all this time; Orlando, Edgar, Pata, Moha... You helped me really a lot. Special mention to the other colleague who has the “pleasure” to share such a relaxing and enjoyable task with me till the very last second, Steve.

Finally, I want to thanks all the people of the university and hospital who helped me through this epopee.

Summary

Abstract	I
Resum	II
Resumen	III
Acknowledgment	IV
Summary	V
List of Figures	IX
List of Tables	XI
Chapter 1: Introduction	1
1.1. Background	1
1.2. Problem statement	3
1.3. Objectives	4
1.4. Scope	5
1.5. Project Overview	6
1.5.1. Requirements and specifications	6
1.5.2. Data	8
1.5.3. Retinopathy of prematurity	11
1.5.4. Programming environment	16
1.6. Theoretical background	18
1.6.1. Image Processing	18
1.6.2. Medical Image Analysis	18
1.6.3. Artificial Neural Network	20
1.6.4. Deep Neural Network	22
1.6.5. Convolutional Neural Network	24

1.6.6. U-Net	25
1.6.7. Feature Extraction	26
1.6.8. Decision Tree	27
Chapter 2: State of Art	28
2.1. Segmentation of Retina Images	28
2.1.1. 2-D Gabor Wavelet and Supervised Classification	28
2.1.2. Trainable COSFIRE Filter	29
2.1.3. Major Vessel Extraction and Sub-Image Classification	30
2.1.4. Ensemble Classification-Based	30
2.1.5. Cross-Modality Learning	31
2.1.6. CNN – Deep Learning	32
2.1.7. Results obtained by previous methods	33
2.2. Plus and no Plus Discrimination	34
2.2.1. Automated Identification of Retinopathy of Prematurity by Image-Based Deep Learning	34
2.2.2. Plus disease in retinopathy of prematurity: pilot study of computer-based and expert diagnosis	35
2.2.3. Automated diagnosis of plus disease in retinopathy of prematurity using deep convolutional neural networks.	36
Chapter 3: Methodology	37
3.1. Main and Modules Structure	37
3.2. Connectivity and Modules	39
3.3. Main Challenges	40
3.4. Model Preparation	41
3.4.1. Data Preparation	41
3.4.2. Training	42
3.5. Prediction and Performance Evaluation	46
3.6. Prediction for Real Images	48

3.6.1. Mask Creation	48
3.6.2. Data Preparation Real	49
3.6.3. Prediction Real	49
3.7. Main Challenges	50
3.8. Feature Extraction (Classifiers Training)	51
3.8.1. Rumour Clean	52
3.8.2. Skeletonization	53
3.9. Main Challenges	56
3.10. Classifiers Training	56
3.11. Performance Evaluation	59
Chapter 4: Results	60
4.1. Data Augmentation by Patches	60
4.2. Pre-processing	61
4.3. U-NET (Segmentation)	62
4.4. Cleaning	64
4.5. Skeletonization	66
4.6. Standard Pool Distribution	68
4.7. Double Threshold	71
4.8. Mean Distance Threshold	72
4.9. Smart Threshold	73
4.10. % Threshold	74
4.11. Decision Tree	75
4.12. Different Pools Evaluation by the Threshold Proposed	76
Chapter 5: Discussion of the results	79
Chapter 6: Conclusions and Future Works	81
Chapter 7: References	83
Annex AI Specification RetCam	88



Annex All Images for Classifier Training 90

List of Figures

Figure 1, Retina Images Available and employed during the different stages (Own Production).....	8
Figure 2, Zone I ^o , II ^o , III ^o of the Retina (Gole, et al., 2005)	12
Figure 3,ROP stage 1 (A), stage 2 (B), stage 3 (C), stage 4 (D) and stage 5 (E) (Gole, et al., 2005)	12
Figure 4, Non-Plus ROP (A), Pre-Plus ROP (B), Plus ROP (C) (Own Production)	13
Figure 5, Aggressive Posterior ROP (Gole, et al., 2005)	14
Figure 6, Medical Image Analysis steps (Own Production)	19
Figure 7, ANN structure (Walczac & Cerpa, 2003)	20
Figure 8, Dropout implementation of the network, from standard (a) to dropout (b) (Srivastava, Hinton, krizhevsky, Sutskever, & Salakhutdinov, 2014).....	23
Figure 9, Receptive field between layers (Indolia, Goswami, Mishra, & Asopa, 2018)	24
Figure 10, U_Net structure (Ronneberger, Fischer, & Brox, 2015).....	26
Figure 11, Example of Decision Tree structure (Song & Lu, 2015).....	27
Figure 12, Block Diagram Code (Own Production).....	37
Figure 13, Colaboratory adaptation of canonical python structure	40
Figure 14, Training workflow (Own Production).....	42
Figure 15, images and relative histograms through the pre-processing, gray scale conversion, normalization, CLAHE filter, gamma adjustment filter (Own Production)	44
Figure 16, Patches example (Own Production).....	45
Figure 17, Test workflow (Own Production)	46
Figure 18, Predicted segmentation workflow (Own Production)	48
Figure 19, Mask creation by first and second approximation (Own Production).....	49
Figure 20, Standard creation workflow (Own Production)	51
Figure 21, Cleaning Process (Own Production)	52
Figure 22, Skeletonization process through the stages of segmentation, skeleton detected, skeleton region, segmentation region (Own Production)	53
Figure 23, Two stages segmentation region definition: Segmentation overlapping, Flood filling (Own Production).....	54
Figure 24, Standard classifier evaluation workflow (Own Production)	55
Figure 25, Double Threshold (Own Production)	56
Figure 26, Mean Distance Threshold (Own Production).....	57
Figure 27, Smart Threshold (Own Production)	57
Figure 28, % Threshold (Own Production)	58

Figure 29, Decision Tree (Own Production)	58
Figure 30, Image examples of the Low quality (A), Medium quality (B), and High quality (C) pools (Own Production)	59
Figure 31, Image and Ground Truth Patches Databases Creation (Own Production)	60
Figure 32, Raw and Pre-processed Images	61
Figure 33, U-Net Segmentation (Own Production)	Error! Bookmark not defined.
Figure 34, segmentation of retina image provided by the SJD hospital (Own Production)	63
Figure 35, Segmentation Cleaning (Own Production)	64
Figure 36, Zoom of the top border of segmentation frames (Own Production)	64
Figure 37, Field of View Border correction	65
Figure 38, Process of skeletonization (Own Production)	66
Figure 39, Examples of Thickness and Tortuosity branches distribution for three retina images (Own Production)	68
Figure 40, Standard Pool distribution of ROP and NO ROP Mean Values (Own Production)	70
Figure 41, Decision Tree Classifier Flowchart (Own Production)	75
Figure 42, Double Threshold, Mean Threshold, Smart Threshold and % Threshold evaluation by the Match-Reliability Ratio	77
Figure 43, Global Evaluation by the Match - Reliability Ratio	77

List of Tables

Table 1, AUC ROC on DRIVE Database with the different methods proposed in the state of the art ..	33
Table 2, Network Structure	43
Table 3, U-Net evaluation	63
Table 4, Skeleton Description with number of branches, type (single or multi-connected), start point, endpoint	67
Table 5, Standard Pool Mean Tortuosities and Thicknesses	69
Table 6, Min and Max Values of Tortuosity and Thickness for the ROP Plus and No Plus of the Standard Pool	71
Table 7, Double Threshold Results	71
Table 8, Double Threshold Score	71
Table 9, Mean Values of Tortuosity and Thickness for the ROP Plus and No Plus of the Standard Pool	72
Table 10, Mean Distance Threshold Results	72
Table 11, Mean Distance Threshold Score	72
Table 12, Smart Values of Tortuosity and Thickness for the ROP Plus and No Plus of the Standard Pool	73
Table 13, Smart Threshold Results	73
Table 14, Smart Threshold Score	73
Table 15, % Threshold Results	74
Table 16, % Threshold score	74
Table 17, Result of classification in terms of the different classifier using simple threshold applied on the six Pool analyzed	76
Table 18, Average Score Classifiers	76

Chapter 1: Introduction

1.1. Background

Artificial Intelligence (AI) is defined as any technique that enables a computer to mimic human intelligence (Chen & Wong, 2019) and, for many years, it has been used with a medical purpose in order to replicate or even improve human expertise, still questioning feasibility and viability of the use of such a powerful technology applied to a very sensitive field (Chandrasekaran, 1983).

Fundus imaging is well recognized as one of the principal methods used, more and more often in the last years, in order to establish a healthy condition of the retina and thereby diagnose and monitor the relative abnormalities; the morphological characteristics of retinal blood vessels detected have been associated to a wide range of disease, among which Retinopathy of prematurity in infancy. If properly exploited, it not only allows to spot the disease but also to categorize the relative level of gravity and, therefore, to obtain crucial information for determining the most appropriate approach to address the specific case (Gelman, Martinez-Perez, Vanderveen, Moskowitz, & Fulton, 2005).

In order to process the images, several approaches have been followed, depending on results obtained and available technologies; the use of a Convolutional Neural Network architecture (CNN) has been adopted for image processing and recognition problems since the 1970s (1); nevertheless, the dawn of alternative Machine Learning (ML) techniques, like Support Vector Machine (SVM) and Bayesian Network (BN), along with the difficulty of NN to scale well with task complexity, postponed the complete establishment of Neural Network as a reference targeting this task; only the arrival of Deep Learning (DL), brought it back into the spotlight (Liskowsky & Krawiec, 2016).

The reasons because DL is now leading the field are multiples, the possibility to train the network for notably wide ranges of classes, up to thousands, adequately increasing the fully connected layers and broadening the supervised training (Krizhevsky, Sutskever, & Hinton, 2017); the use of not saturating transfer function that does not vanish in the extremes and with sparse effect obtain substantial improvements (Glorot, Bordes, & Bengio, Deep Sparse Rectifier Networks, 2011); the introduction of regularization method such as Dropout that improves the results decreasing the overfitting (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014); the massive development of general-purpose computing on the graphical processing unit (GPUs) (Liskowsky & Krawiec, 2016).

The dawn of the Deep Neural Network effectively answer the requirements for proper characterization and categorization of the fundus imaging; providing a tool that is able to overcome many downsides met by the previous methods; nevertheless, being applied in a supervised way is characterized by some inefficiency caused by an external condition. The characterization of an image is manually performed by specialist doctors like ophthalmologists, though they are following a series of rules that are the result of experience and interpretation, not of an uninterpretable and neat standard; therefore, what is supposed to be the gold standard of the evaluation could vary and thereby, affect the network in either phase of training and testing. Furthermore, the pool of data used for that evaluation is often affected by noise, focusing, and heterogeneity of the overall quality; thus, the uniformity of evaluation is not perfectly maintained, and there are different studies diverging, even though in a minor way (Kareem, et al., 2007).

Finally, the introduction of limitations trying to ensure better transparency are strongly affecting not the potential but the use of the tool; the possibility of training with labeled data the network with large pool results in notable results but lacks in any explanation of the paths and pattern that are followed by the algorithm to come to a decision. If the use of the network as a "black box" represents on one side the chance of a deeper level of learning beyond the mere result; for particular fields, such as the diagnosis of an illness, it is ethically unacceptable to let pass without any system of control or revision; therefore, the great potential of the network is bound by the application on simple task or the study of the interpretability of the results obtained (Data Protection Act 2018, 2018).

1.2. Problem statement

Currently, the diagnosis of Retinopathy of prematurity is manually made by an expert that analyses a set of images of the retina that are taken from a different point of view and that altogether globally reconstruct its structure. Nevertheless, there is a series of factors that make this task uneasy about being carried out.

The expertise required for such an operation is extremely niche and complex in the first stand; presently, in Spain, only five hospitals claim a department with experts able to correctly spot the illness: Hospital Joan XXIII in Tarragona, Hospital De Granollers, Hospital de Sant Pau in Barcelona, Hospital Sant Joan de Deu in Barcelona, Hospital Trueta in Girona.

Furthermore, the diagnosis, despite following common guidelines, is complex and not standardized and, therefore, leaves room for interpretability and heterogeneity of evaluation that leads to a controversial reading of the images.

Moreover, the high level of complexity of the diagnosis is complicated by a further boundary, the time; time is a key factor in terms of prevention of an illness that can easily spread and worsen, enhancing the severity of the symptoms till producing irreversible outcome.

Finally, the evaluation is complicated by the quality of the retina images that is often compromised by obsolete as much as deteriorated equipment that not only results in a rise of the complexity level of assessment for the illness but also in heterogeneity of images, thereby, tragically affecting the robustness of possible system of evaluation.

In the complex frame described, the necessity of one tool able to deal with all those problems outstands; indeed, the target is a tool that could address the lack of expertise in an affordable and rapid way overcoming all the different challenges described and producing a consistent and reliable outcome. Moreover, it is expected that the solution provided could be customized in order to be managed by a doctor without any particular expertise, neither in the Retinopathy nor in the theoretical explanation of the tool.

1.3. Objectives

The objectives of this master thesis are the following:

- Define, through the use of appropriate features and parameters, different standards for Plus and no-Plus Retinopathy of prematurity.
- Define the standards with quantitative parameters that could be compared with the empirical parameters analyzed by the current medicine directives followed by the field experts.
- Elaborate an automatic system able to discern between images interested by ROP at a Plus and non-Plus stage using the standard defined.
- Build an algorithm able not only to discern between the categories but also to underline the factors taken into account for the choice.
- Use the code developed for implementing a dash application in order to build an effective tool exploitable from doctors with a different background.
- Generalize the process in order to be able to process data of different quality, formats, and origins.
- Develop the code in a functional way in order to leave the chance to easily implement, add or modify functions and script during the following stages of the project.

Through the collaboration with medical and engineering personnel of the Innovation Department HSJD.

1.4. Scope

The scope of this master thesis is double; first, it aims to build in Colab a script, thus written in Python, able to properly acquire labeled images, between ROP Plus and ROP, not Plus, provided with Field of View mask and to use them for training a model capable of segmenting the retina from the images, next, to develop a set of filters customized in order to effectively clean the image; second, it requires to use a set of segmented and labeled images in order to extrapolate all the important information, in terms of tortuosity and thickness of the vessels, and use them to create a standard apt to discern and categorize unlabelled images.

1.5. Project Overview

1.5.1. Requirements and specifications

The project aims to produce a tool that is going to be a starting point in order to target and effectively address the problem stated; indeed, to overcome the problems of expensive, high heterogeneity, scarcity of expertise, lack of standard parameters, and timeliness required.

Overall, the study aims to produce not only an automatic tool able to discern between images of ROP cases and no-ROP cases, but also to categorize them between the different stages of ROP I^o, II^o, III^o, IV^o and even to distinguish between Plus, pre-Plus and not Plus cases; nevertheless, this sub-project is focusing only on a particular stage, that is differentiating between Plus and no-Plus.

Therefore, the differentiation between the two stages is the goal of this project and has been addressed by the use of a Convolutional Neural Network; this is able to learn the different patterns and characteristics of the stages with a previous step of training carried out by labeled images, thus a supervised method.

One of the main strong points of this choice is the absolute portability of the solution; an application built with a script properly integrated would allow not only to overcome the tragic shortage of expertise creating an application suitable to be installed in any hospital, but also would tragically affect the speed of reaction producing a tool that on a theoretical level, could be installed anytime and anywhere on a laptop or even phone.

Nonetheless, an automatized cannot deal with such sensitive decisions without any restriction; indeed, it is subjected to strict ethical boundaries that preclude any solution that is not demonstrable and explainable. Indeed, the label of the CNN has been changed from the actual stage sought to the detailed segmentation of the retina that cleaned and characterized is giving enough information to define a standard for each stage and, thereby, solving the problem without incurring any possible problem of "black box" interpretability.

This initial part of the project is inspired on a GitHub (<https://github.com/orobix/retina-unet>) that provides basic code for the first stage of segmentation of the image, with the Network structure, and using labeled images from one of the most popular publicly available datasets (Fraz M. M., et al., 2012), DRIVE (<http://www.isi.uu.nl/Research/Databases/DRIVE/>), is able to reach an AUC ROC, tested on the same dataset images, of 97.90. This particular model is working not only by image and label

but also requires a mask in order to filter out the background that otherwise would challenge the network critically; indeed, the creation of a mask for each image is also essential.

Therefore, the project is logically split into the following steps:

- 1) Training of the model provided by previous studies with the set of images labeled and masked, provided by DRIVE.
- 2) Evaluation of the results obtained.
- 3) Creation of a set of masks for the images provided (vessel segmentation) by the hospitals' shared database.
- 4) Prediction of those images.
- 5) Cleaning/Filtering of the image.
- 6) Creation of two datasets of information, ROP plus and ROP no PLUS, and definition of standards.
- 7) Evaluation of the result obtained.

Moreover, the evaluation required are performed in different levels; due to the complexity of the problem, the cumbersome and bureaucratic procedure to access new data, and the time/computational demanding process developed; the evaluation is developed with several progressively wider pools of data; again, this sub-project is focusing on a small pool that is successively broadened.

1.5.2. Data

The data used are color fundus images of the retina used in order to evaluate the condition of the interior surface of the eyes.

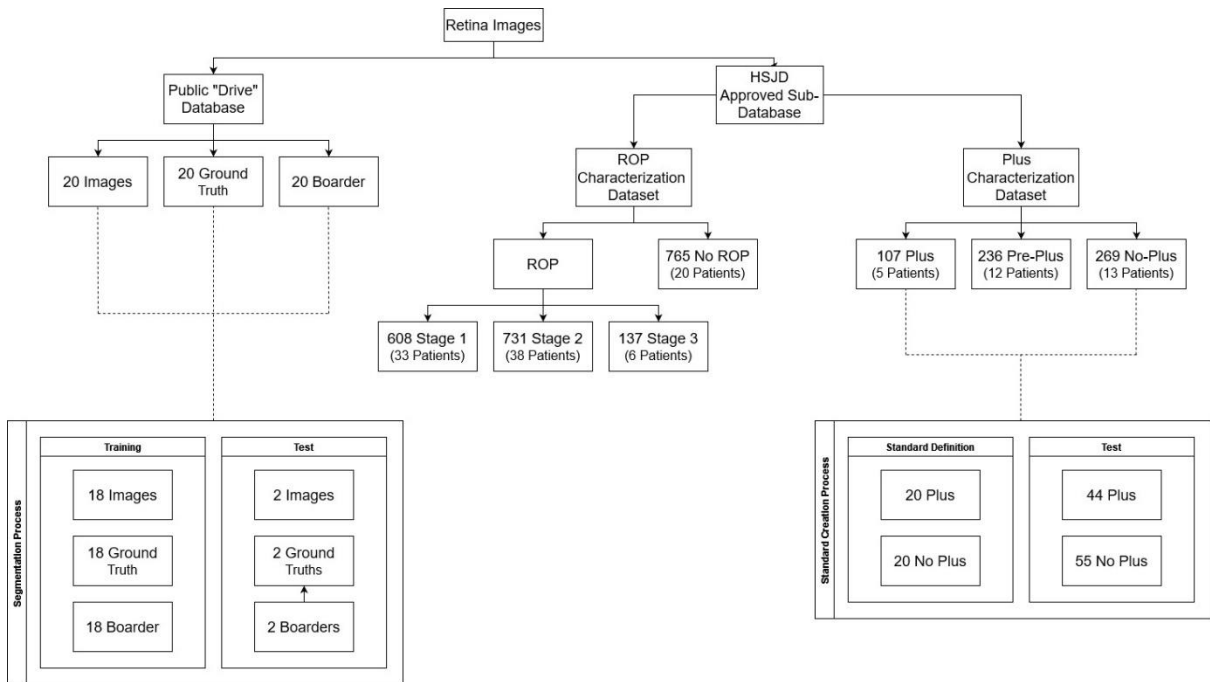


Figure 1, Retina Images Available and employed during the different stages

In order to carry out this project, different sources have been used (Figure 1, Retina Images Available and employed during the different stages); the first set of images used for the training of the network, with relative masks and labels, is from the DRIVE database (Staal, Abramoff, Niemeijer, Viergever, & van Ginneken, 2004).

The DRIVE database (publicly available at [http://www.isi.uu.nl/Research/Databases/DRIVE/.](http://www.isi.uu.nl/Research/Databases/DRIVE/)) has been built by a Netherlands screening study conducted on a pool of subjects aged between 25 and 90 years old. From the original number of 400 attendees, only 40 were selected and included in the database; they were composed of 7 showing mild signs of early diabetic Retinopathy and the other 33 representing a clinical norm. The images were captured in digital form from a canon CR5 nonmydriatic 3CCD camera at 45° field of view. The images size is 768 x 584 pixels, 8 bits per color channel, with an approximate field of view diameter of 540 pixels, and they are compressed in a JPEG

format. The dataset is split into training and test sets, and the labels of the training set were manually segmented by three observers, a computer science student, Meindert Niemeijer and Joes Staal, after been trained by the ophthalmologist Michael D. Abramoff, with a minimum classification certainty of 70%. The field of view selected in order to crop the images and focus is also provided in the form of a mask for each image of the whole dataset.

The second set of images, not labeled and thus, used only for prediction and analysis of data, is from a private drive shared for this project by the San Joan de Deu Fundacio' de Recerca.

This private dataset is shared between five hospitals that by "RetCam" are continuously gathering images in order to study and work on different illnesses with different projects; in order to access the data, a strict bureaucratic process has to be followed with an ethical board that has the final decision on granting the access with the purpose of consistently preserve the patients' privacy. Therefore, every time a new set of data is selected from this database, because recently gathered, selected by new criteria defined by the project or just required for a wider pool of images examined, the process has to be examined and approved by the board. Indeed, the number of data provided by this dataset is constantly increased; the last numbers reported are 765 images from NO ROP cases and 1476 from ROP cases, and 612 cases divided between PLUS, NON-PLUS, and PRE-PLUS.

The images are collected from 5 hospital part of the Cobsortia and joining the RTOC project (Red en Telemedicina Oftalmica en Catalunya):

- Hospital Joan XXIII in Tarragona
- Hospital De Granollers
- Hospital de Sant Pau in Barcelona
- Hospital Sant Joan de Deu in Barcelona
- Hospital Trueta in Girona

The product used and shared by the hospitals is "RetCam Envision ophthalmologic imaging system" (RetCam Envision™ ophthalmic imaging system, 2021); it provides:

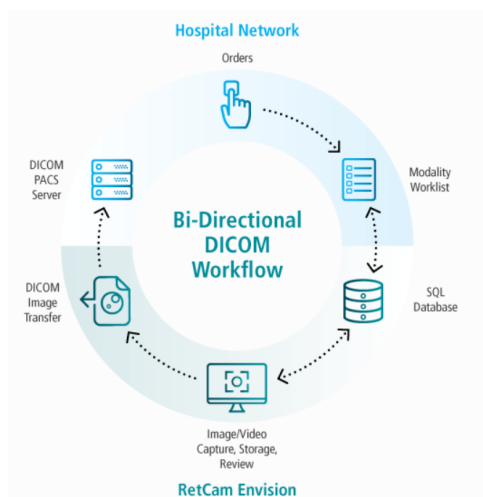


Figure 2, RetCam Seamless Connectivity (RetCam Envision™ ophthalmic imaging system, 2021)

- enhanced illumination and ultra-wide field of view for a better visualization of details and thus evaluation of the sickness
- seamless PACS integration and enhanced network capabilities that allow to decrease the time for transfer and archiving images (Figure 2, RetCam Seamless Connectivity).
- Detachable 130° lens and optional portrait lenses for a greater image visualization and an easier cleaning and disinfecting procedure.
- Fluorescent Angiography lenses for a more proficient examination of retinal blood circulation and thus examination of anatomical abnormalities.

Technical Specification (Annex AI)

1.5.3. Retinopathy of prematurity

Retinopathy of Prematurity (ROP) represents a disease of eyes in prematurely born infants who have not had enough time to properly develop the blood vessels of the retina during the gestational period; it is a biphasic condition starting with the first phase of vessel growth retardation and followed by vessel abnormal vessel proliferation in the retina that may induce scarring of the retina or detachment. The factors associated with this disease are multiple and complex; nevertheless, several studies underline how four factors play a key role in its characterization; the first three are related only to the status and history of the baby and are weight, when lower than 1500g; gestational age, when lower than 30 weeks; health of the baby, whenever the Apgar score is low; on the other side the fourth is influenced by an external factor and is oxygen therapy (Alajbegovic-Holimic, Zvizdic, Alimanovic-Halilovic, Dodik, & Duvnjak, 2015).

Oxygen therapy is a sensitive factor since a high saturation that positively affects babies, leads to an increase of number and gravity in term of ROP cases; a ripple effect is obtained by the hyperoxia that by occlusion produce hypoxia with the subsequent production of Vascular Endothelial Growth Factor (VEGF) and thereby, new vase-proliferation (Saugstad, 2006).

The thresholds defined are indicative, and each case has to be properly examined considering the particularly related background; patterns of slightly different behaviors are confirmed from different studies; indeed, discrimination can be outlined up to a genetic heritage that emphasizes a heterogeneity of severity strongly affected by the race (Tadesse, Dhanireddy, Mittal, & Higgins, 2002).

The first case described in 1942, right next to the first pandemic registered and underlined the difficulties and hesitation encountered facing an infirmity not only for the first time spotted but also substantially rare (T., 1942).

The lack of any kind of standard procedure or even mere common knowledge was addressed by 23 ophthalmologists from 11 different countries who, in 1979, formed an international committee in order to devise the "International Classification for Retinopathy of Prematurity" (ICROP); in this first version of the document, from the inappropriate but traditional term "retrolental fibroplasia", the new terminology "retinopathy of prematurity" is adopted with a first rudimental definition of its features and characteristics; nevertheless, remarking the inefficiency of classification not yet mature or exhaustive (Ben-Sira, et al., 1984). Indeed, modification and extension followed in 1987 and 2005, the current version, with the purpose to integrate information about an illness not yet

comprehensively characterized and under study on one side, and to produce a definition as suitable and clear as possible in order to provide a valuable tool for any doctor facing the diagnosis.

In the current classification are defined location, stages, and particular categories with complications.

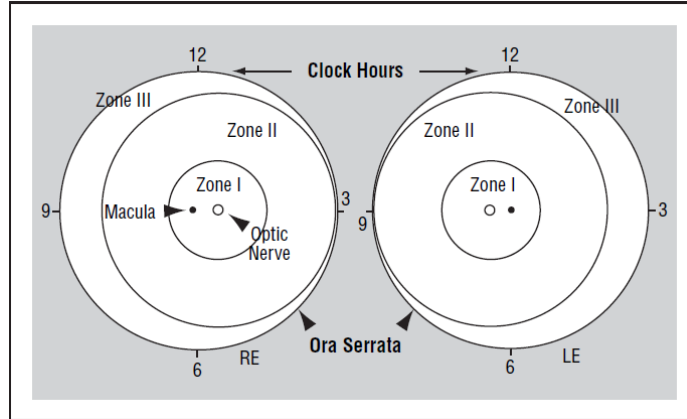


Figure 3, Zone I^o, II^o, III^o of the Retina (Gole, et al., 2005)

In order to properly assess the entity of the illness, the retina is partitioned into three concentric zones centered in the optic disc (Figure 3, Zone I^o, II^o, III^o of the Retin); the most internal one is Zone I^o that extends from the optic disc to twice the distance of the macula; followed by Zone II^o that goes from the edge of Zone I^o to the nasal ora Serrata; finally, Zone III^o is defined as the residual crescent of the retina. The extent and gravity of the disease are strongly influenced by the diffusion of the new vascularization in those areas, and thresholds, made of percentages of the zone interested in terms of clocks of 30^o, are used in order to properly diagnose the entity of the disease.

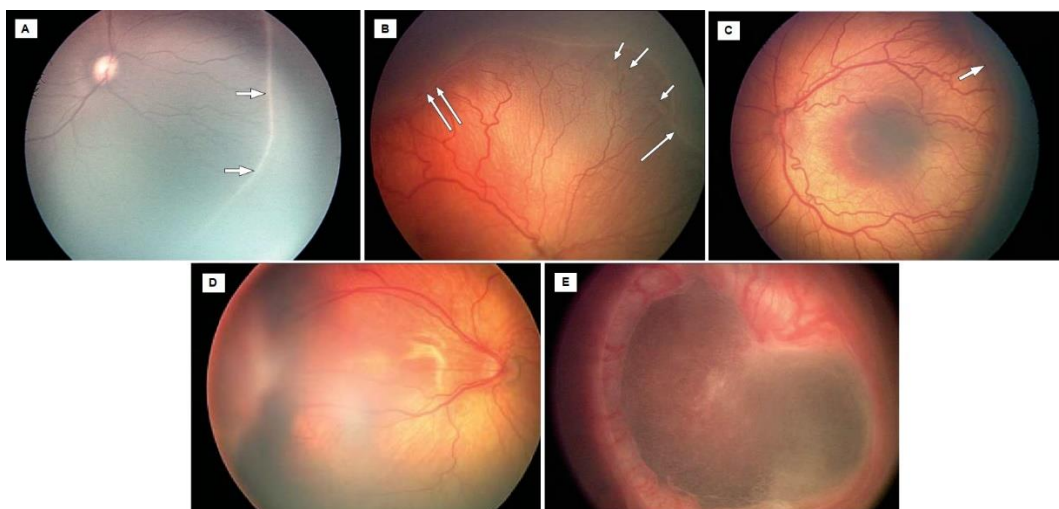


Figure 4, ROP stage 1 (A), stage 2 (B), stage 3 (C), stage 4 (D) and stage 5 (E) (Gole, et al., 2005)

Five stages are defined in order to characterize ROP and are focused on the structure of the area separating avascular retina anteriorly from vascularized retina posteriorly (Figure 4, ROP stage 1 (A), stage 2 (B), stage 3 (C), stage 4 (D) and stage 5 (E)); in Stage I^o, the structure is a white, thin and flat line with abnormal branching or arcading; in stage II^o, the line expands in term of height and width, forming a ridge that extends above the plane of the retina; in stage III^o, the neovascularization extends from the white structure, ridge, into the vitreous; in stage IV^o, there is partial detachment, normally concave and circumferentially oriented; finally in stage V^o, there is a total detachment generally tractional and funnel-shaped.

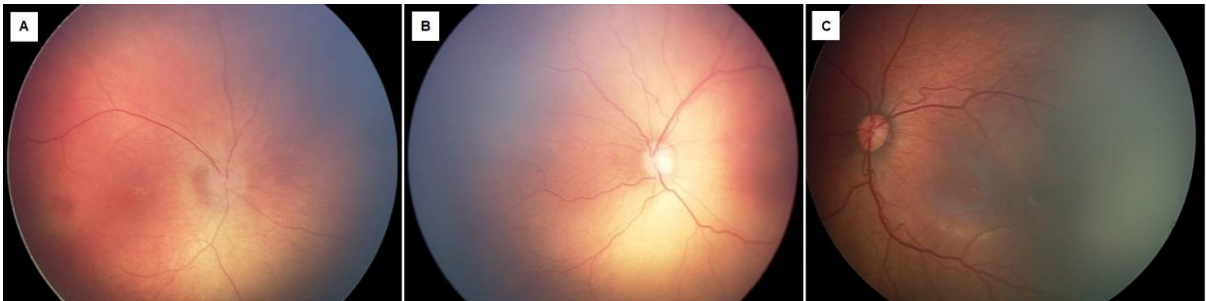


Figure 5, Non-Plus ROP (A), Pre-Plus ROP (B), Plus ROP (C)

Moreover, a further distinction is made between Plus, Non-Plus, and Pre-Plus disease in order to define the gravity of the Retinopathy (Figure 5, Non-Plus ROP (A), Pre-Plus ROP (B), Plus ROP (C)); Plus disease is defined as increased venous dilation and arteriolar tortuosity with possible vascular engorgement and pupillary dilation, in order to assess the level a "standard" photograph is used, and a minimum of 2 quadrants interested is required; furthermore, an intermediate level is recognized between high dangerous cases (Plus) and normal ones (Non-Plus), Pre-Plus, that is defined as a previous stage of Plus in which the illness could easily progress to Plus and indeed, is necessary cautious monitoring.

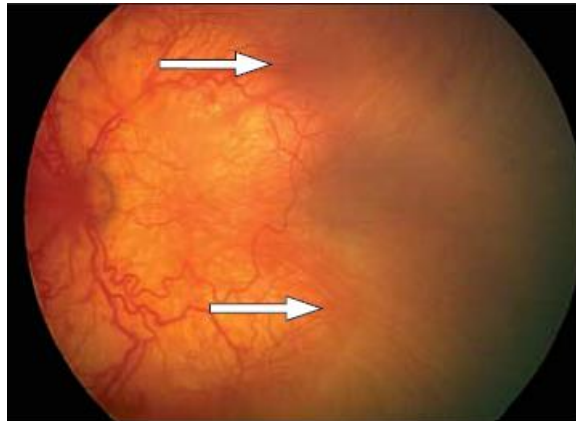


Figure 6, Aggressive Posterior ROP (Gole, et al., 2005)

Finally, a particularly severe form of ROP is defined, Aggressive Posterior ROP (AR-ROP) (Figure 6, Aggressive Posterior ROP); this form is normally localized in all four quadrants with notable dilation and tortuosity; furthermore, it is characterized by shunting that occurs within the whole retina, instead of only at the junction between vascularized and avascularized retina; the peculiarity of this form is not only its gravity but also the velocity in which it progress, skipping the classic initial stages from 1 to 3 (Gole, et al., 2005).

The classification is still not conclusive since not only the illness is still under active research, but also for the continuous strain of achieving a clear characterization of Retinopathy that could be effectively viable without leaving room for interpretation. Indeed, a new version of ICROP is under publication, focusing mainly on the deepening of regression, reactivation, and long-term sequelae (Chiang, et al., 2021 (article in press)).

Another possible classification is built using the Threshold ROP and Pre-Threshold ROP definition; Threshold ROP is defined as a condition with the 50% risk of retinal detachment if not properly treated and is defined with different combinations of clocks of the retina interested and stages developed, if more than five contiguous or eight cumulative clock hours present stage 3 PLUS ROP in zone 1 or 2, the Threshold ROP is reached; on the other side, if those clock hours present stage 1 ROP less than Threshold, stage 2 PLUS ROP in zone 2, stage 3 ROP in zone 2, the Pre-Threshold is achieved, furthermore, any stage 3 PLUS ROP in zone 2 is defined as Pre-Threshold (Komal & Subhadra, 2018).

The evaluation of the infirmity could be done analysing several factors; nevertheless, the different methodologies used are not focusing on the same parameters; "Low Weight Gain Proportion"

compares the weight in the first six weeks of life of the prematurely born child; "WINROP" algorithm takes into account body weight and IGF-1 level in the first 36 weeks of life of the baby; "ROP Score" analyses birth weight, gestational age, weight gain and blood transfusion during the first 6 weeks of life considering also the characteristics of the oxygen incubation; "IGF-1" by the determination of its serum level during the third week post-partum; "Plasma Soluble E-Selectin" scores gestational age of the preterm child and the E-Selectin plasma level; "Tele-screening" assesses the illness by the analysis of digital retinal imaging made by experienced ophthalmologists and collected by trained medical personnel; "Optical Coherence Tomography" uses cross-sectional images of the retina providing insight at a cellular and subcellular level into normal retinal development and its long term sequelae (Shah, et al., 2016).

In order to cure the ROP, it is necessary to contain the development and spreading of the neovascularization; indeed, it is necessary to inhibit the production of VEGF, responsible for it, or replace the IGF factors, able to mediate and control the VEGF production, whose production terminates after the premature conception. The first option is to eradicate the portion of the retina responsible for the production by two ablative methods, Cryotherapy and Indirect Laser Photocoagulation; both approaches are invasive and are likely to jeopardize the sight of the patient depending on the extent and timing of the intervention; nevertheless, the laser therapy, with more remarkable results, is considered the best evidence-based therapy for ROP. The second option is to inject intravitreal Anti-Vascular Endothelial Growth Factor drugs able to adequately reduce the VEGF production; nonetheless, this procedure could have side effects related to targeting indiscriminately all the VEGF isoforms lacking in focus; therefore, the economical and practical solution has to be evaluated considering the risk of possible poor timing and dosage in its usage. A possible alternative to this method, with a higher focus, is the use of Propranolol, whose effect is appropriately targeting the VEGF involved in the process without affecting the complete production of the protein but incurring many side effects such as Bradycardia and Hypotension. The third option is to supply IGF factors by the intravenous administration of recombinant IGF-1 (rhIGF-1) with its binding protein 3 (rhIGFBP-3). Alternative supplemental therapy is to treat borns prematurely by "Granulocyte Colony Stimulating Factors" (GCS-F); this solution shows positive results but is not statistically relevant. Finally, "Gene Therapy" associates the severity of ROP with mutation and polymorphism of various genes (Mutlu & Sarici, 2013).

1.5.4. Programming environment

This project has been developed with Python 3.6.9 (Van Rossum & Drake, Python 3 Reference Manual, 2009), an open-source, interpreted, high-level, general-purpose programming language.

The machine used to run the code is an Acer Aspire NC-A315-41-R8N8 with a processor AMD Ryzen 7 2700U with Radeon Vega Mobile Gfx 2.20 GHz with an installed RAM of 8 GB.

In order to overcome the high computational power required by this project, Google Colaboratory Pro has been used (Bisong, 2019); Colab is a research project for prototyping machine learning models on powerful hardware options such as GPUs and TPUs. It provides a serverless Jupyter notebook environment for interactive development. The standard version of Colab is available with all the other free Google suit products; nevertheless, the stage of training is imposing the use on the only GPU with a considerable computational power necessary of about 20 Gb; Indeed, the standard version, providing 12 Gb of GPU, is not meeting the power necessary and the upgrade to the PRO version with about 25 Gb fixed the problem.

A notable number of functions is used from a wide range of libraries; therefore, it is complex to link one single library to a specific use; nonetheless, the generic aim of each of them can be summarized in the following description:

- "ConfigParser" (Lundth, 2001): management of the configuration file.
- "cv2" (Bradski, 2000): range of different filters.
- "h5py" (Colette, 2013): change of more compact and adaptable format.
- "Keras" (Chollet & others, 2015): building and modeling network.
- "matplotlib" (Hunter, 2007): plot images.
- "numpy" (Harris, et al., 2020): arrays definition and transformation.
- "os" (Lundth, 2001): help with path directions.
- "pandas" (McKinney, 2010): data frame creation and modification.
- "PIL" (Umesh, 2012): images importation and exportation.
- "random" (Van Rossum, The Python Library Reference, 2020): creation random variables.
- "scipy" (Virtanen, et al., 2020): the creation of centroids.
- "seaborn" (Waskom, 2021): creation and visualization of histograms.
- "skan" (Nunez-Iglesias, Blanch, Looker, Dixon, & Tilley, 2018): skeletonization of the image.
- "skimage" (Van der Walt, et al., 2014): change of format, color, morphology, segmentation, study of region and filters.

- "sklearn" (Blondel, et al., 2011): evaluation of the results.
- "sys" (Lundth, 2001): automatization of the process of folder creation.

1.6. Theoretical background

1.6.1. Image Processing

Image processing (Da Silva & Mendonca, 2005) consists of the process of manipulation of digital images using computers; in the last decades, this process has been assuming more and more importance extending the possible application to a wide range of fields among which biomedical. In order to be stored and used, an image has first to be sampled and quantized; thereby, it is represented by a matrix of finite numbers. There are different typologies of image processing that can be summarized by:

- Image enhancement: class of procedures that allows increasing the quality of the image in terms of specific characteristics targeted for specific analysis; the process is not performed by addition of information but instead by better use of the data for the particular objective as, for instead, using filter or histogram manipulation.
- Image restoration: class of procedures that do not target the actual features and quality of the image but that aims to properly spot and describe the system, linear or not, that is degrading the image in order to remove its repercussion from the image.
- Image compression: class of procedures that aims to mediate the common problem of processing images represented by a demanding number of bits; this process is parted in "transformation" that reduces the correlations between pixels, "quantization" that allow mapping the image to a finite number of symbols, "coding" that allow passing to a string of bit more manageable.
- Image analysis: class of procedures that automatically extract information from one or a sequence of images following different levels of knowledge: features detection, segmentation, region labeling; it is widely used in many fields such as computer vision.

1.6.2. Medical Image Analysis

A wide variety of factors played a fundamental role in the recent ascent of image analysis as an effective and well-spread medical technology: increased access to digital data, data set size, pressure for an improvement of the healthcare system, increased use of image data; this trend was reflected

in a gradual evolution of the technology that passed from "native digital methods" such as Computer Tomography (CT) and Magnetic Resonance Imaging (MRI); through analogic methods as for instead Endoscopy and Radiography; till to arrive at Computer-Aided Diagnostic (CAD). The goal of this technology is to provide valuable second opinions to problems that require high time, expertise or whose accuracy and consistency are notably low. Three aspects are fundamental considering medical image analysis; first, biological structures are subject to -inter and intraindividual alterability; second, biological structures cannot be separated and analyzed from the entire picture; third, high levels of reliability and robustness are necessary. After the first run-up step, split between the acquisition, digitization, pre-processing, and segmentation, the actual image analysis takes place, passing through feature extraction, classification, and interpretation (Figure 7, Medical Image Analysis steps).

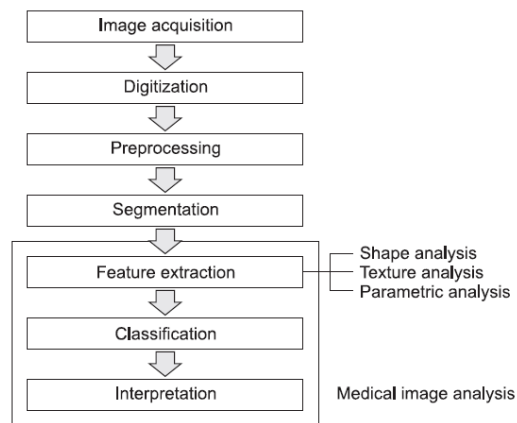


Figure 7, Medical Image Analysis steps

There are three typologies of feature extraction, shape analysis, texture analysis, and parametric analysis. Shape analysis refers to the profile or physical structure of the object and is represented by boundary, regions, moment, and structural representation; the descriptors selected have to follow four particular qualities, uniqueness, calculation independency, rotation invariance, and scale invariance. Texture analysis refers to the spatial relationship of pixel values in an image region and is characterized by the local spectral or frequency. Parametric mapping analysis entails the construction of spatially extended statistical processes in order to test hypotheses about regionally specific effects; the resultant maps are image descriptors with voxel values, under the null hypothesis and distributed following a probability density function.

The classification is a stage in which different patterns are analyzed or detected in order to categorize pixels or features; there is a wide variety of classifiers that goes from "Nearest Neighborhood" and "Decision Tree" to more complex systems such as "Linear Vector Machine" and

"Artificial Neural Network." Generally, the classification is a supervised method, and thus, it needs a set of supervised data in order to be properly trained and then tested (Kim, Son, & Kim, 2011).

1.6.3. Artificial Neural Network

Artificial Neural Network is one of the foundations of Computational Intelligence (CI) and provides a method for dealing with categorization and time-series problems characterized by high pattern-oriented complexity; its non-parametric nature enables a development despite lack of prior knowledge of the distribution of the data and avoids possible interaction effects between variables. Furthermore, its ability to create linear as much as non-linear models makes it competitive in terms of versatility.

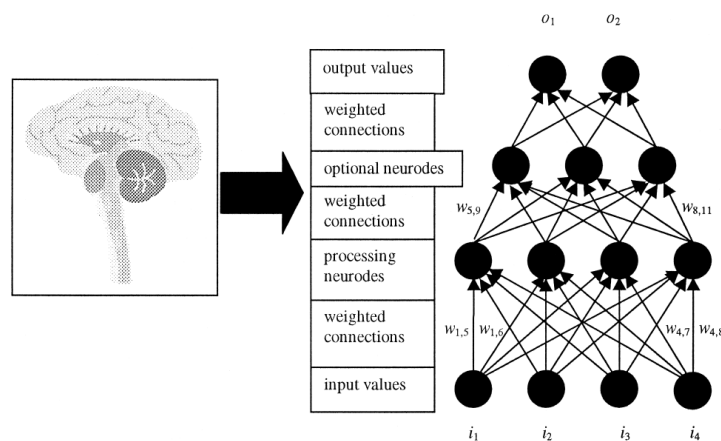


Figure 8, ANN structure (Walczac & Cerpa, 2003)

Artificial Neural Network is a technology resultant from deep studies of the brain and the nervous system and aims to emulate a biological neural network in its electrical activity. Series of processing elements called Perceptrons are connected and arranged in layers or vectors; each layer is then connected from the first input layer until the last output layer in a partial or complete disposition depending on the architecture (Figure 8, ANN structure). Each input values i_n is multiplied by the connection weight $w_{n,m}$ in order to define the apport given to the m following perceptron and through the adjustment of the set of weights, the learning is emulated; all the adjustments are aggregated using functions such as summation, averaging, input maximum or mode value in order to

obtain a single input; finally, a non-linear transfer function, such as sigmoid or hyperbolic-tangent, is applied to obtain the output.

There is a wide range of learning algorithms used by Artificial Neural Networks; nevertheless, they can be classified into two major categories, supervised, which use the difference between predicted and targeted output; unsupervised, that only receive input stimuli and organize itself in order to have hidden processing elements reacting differently to each set of stimuli.

The methodology required for the design of an ANN is:

- 1) Determine data to use.
- 2) Determine input variables.
- 3) Split data into training and test sets.
- 4) Define the architecture of the network.
- 5) Select the appropriate learning algorithm.
- 6) Transform the variables to inputs.
- 7) Train until obtaining acceptable error.
- 8) Test in order to validate the network.

During this process, two critical issues have to be solved, the selection of proper variables that have to adequately describe the input but not giving superfluous information; capturing a sufficient quantity of training examples.

The choice of the dimension of the Artificial Neural Network is fundamental for a proficient result since exceeding in terms of increase could lead to overfitting and, in terms of reduction, could incur underfitting. There are two different approaches for the determination of the input variables; the first is completely relying on the connection weight in order to focus on the critical ones; the second is smoothing the process identifying and removing the "noise" variables that are highly correlated with methods such as PCA.

The learning method is what enables the correct model categorization and time-series problems; examples of unsupervised learning systems are Adaptive Resonance Theory (ART), Self-Organizing Map (SOM), and Hopfield networks that do not require the output values for the training samples and form new categories every new example depending on how closely it matches the previously created; examples of supervised ones are Backpropagation (MLP), Radial Basis Function (RBF) and Fuzzy ARTMAP networks, with a superiority of the first, reflected on commercial availability and

shells support, due to its ability not only to handle noise in the training stage but even to better generalize when noise is not exceeding.

The main architecture of an ANN consists of the number of layers; indeed, the number of hidden layers added to the input and the output layers capable of classifying only data linearly separable; and the number of nodes for each layer, apart from the first hidden layer of an unsupervised learning system that is less flexible and requires an equal quantity of nodes to the input layer. Increasing the number of hidden layers improves the closeness-of-fit of the model but reduces its smoothness or extrapolation capabilities. Increasing the number of nodes per layer improves the accuracy, but exceeding it would result in negligible improvements paid by notable training time augmentation.

The choice of dimensions of the training data is also critical since a big pool results expensive, in terms of acquisition and elaboration, a small one could not be comprehensively representative of the real data.

There are several heuristic rules aiming to characterize the perfect choices of networks, such as percentages of nodes linking consecutive layers or relations between the training set and the number of nodes or time-analyzed; nonetheless, progress experience and iterative method are the only solutions to approach the singular cases that cannot perfectly fit in a general way (Walczac & Cerpa, 2003).

1.6.4. Deep Neural Network

A Deep Neural Network (DNN) is an Artificial Neural Network composed of a stack of multiple processing layers to learn representations of data with multiple levels of abstraction. Previous conventional machine-learning techniques were limited in their ability to process natural data in a raw form since they required a feature extractor able to transform the raw data into a suitable feature vector; instead, Deep-learning methods are representation-learning methods; therefore, they have a set of method that allows a machine to be fed with raw data and automatically discover the representations needed for detection or classification. The importance of this density of layers is not only that, in the resultant representation, they are able to amplify critical aspects and suppress irrelevant variations; but also those layers of features are not designed by human engineering but learned from data using a general-purpose learning procedure, and thus, their success and development is not dependent on expensive engineering by hand but only on computation and data.

In the majority of the cases, DNN is used with a procedure called Stochastic Gradient Descent (SGD); this process is computing the outputs and the errors over few random examples of the pool of data, is computing the average gradient for those and is finally adjusting the weight accordingly; the process is repeated for many small sets of data until the average of the objective function stops decreasing (Yann, Bengio, & Hinton, 2015).

The most popular non-linear function used in Deep Neural Networks is Rectifier Linear Unit (ReLU) which is simply the half-wave rectifier $f(z) = \max(z, 0)$; this function, respect smoother function previously used such as Tanh $f(z) = \tanh(z)$ or Sigmoid $f(z) = \frac{1}{1+e^{-z}}$, solves the vanishing gradient problem avoiding compressing the gradient layer by layer; furthermore, the sparsity model allows a better linearization of the final input and increases the robustness of the model (Glorot, Bordes, & Bengio, Deep Sparse Rectifier Neural Networks, 2011).

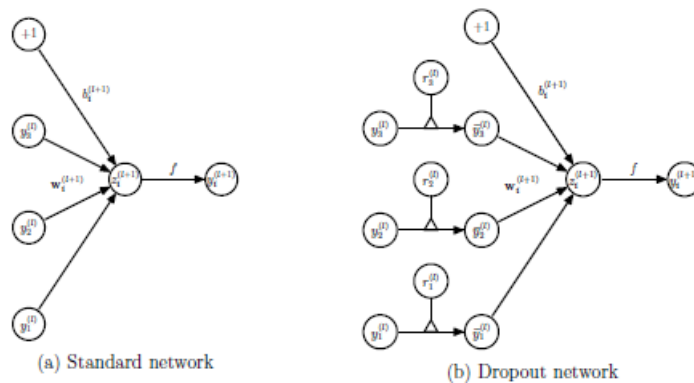


Figure 9, Dropout implementation of the network, from the standard (a) to dropout (b) (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014)

Deep Neural Networks' multiple non-hidden layers allow to learn very complicated relationships; nevertheless, with limited training could build those as a result of sampling noise incurring in overfitting. In the past, many solutions have been implemented, such as weight penalties and soft weight sharing, but now the most effective and used is Dropout. Theoretically, the best way to avoid overfitting is to develop and run all the possible settings of the parameters and average the solutions; this solution has notable results but is paid by considerable downsides; if the network grows, this solution could get prohibitively expensive, computationally demanding and also the amount of data could easily become a boundary. Dropout addresses all those issues, temporarily removing (dropping) units from the network along with all their connections in a random way (Figure

9, Dropout implementation of the network, from the standard (a) to dropout (b)); fixed a probability of drop "p" for each unit a "thinned" version of the network is created and trained every time; in the testing phase, the weights are scaled-down by the probability "p" (Srivastava, Hinton, krizhevsky, Sutskever, & Salakhutdinov, 2014).

1.6.5. Convolutional Neural Network

Convolutional Neural Network (CNN) has deep-forward and a notable ability to generalize, higher compared to network fully connected. It is considered a competitive, if not superior, choice due to the lower number of parameters used that results in a smoother training less affected by overfitting; to the incorporation of the two learning processes of classification and feature extraction; the particular facility in implementing large networks.

A general model of Convolutional Neural Network is composed of four components: convolution layer, pooling layer, activation function, and fully connected layer; the particular differences from a generic network lies in the second layer that is peculiar of this typology of networks.

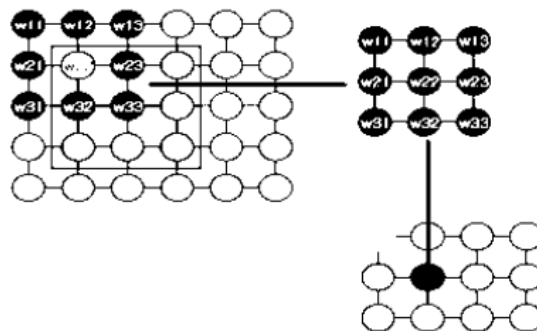


Figure 10, Receptive field between layers (Indolia, Goswami, Mishra, & Asopa, 2018)

In the convolution layer, an individual layer is connected to a group of neurons of the previous layer called receptive field (Figure 10, Receptive field between layers), a kernel of weight is sliding

horizontally and vertically (convolution operation) in order to create a feature map, of reduced dimension, following the formula

$$a_{i,j} = \sigma((W * X)_{i,j} + b)$$

Where X is the input, W is the weight of the kernel, b is the bias, $*$ represents the convolution operation, and σ is non-linearity introduced in the network.

There are several CNN architectures such as LeNet, AlexNet, and Google Net (Indolia, Goswami, Mishra, & Asopa, 2018).

1.6.6. U-Net

U-Net is a particular neural network that is built in order to overcome the boundaries of CNN with a particular solution; the main limitations of CNN is the strong dependency on the size of the training set; therefore, a solution was found feeding it with local regions of the images (patches) that tragically increase the training number; this approach has two main drawbacks represented by a deceleration of the process and possible redundancy due to overlapping, and a trade-off between localization accuracy and the use of context.

The basic concept is to supplement the contracting network by successive layers where the pooling operator is substituted by upsampling; thereby, the second layers increase the resolution of the output, and a successive convolution layer works at a more precise reassembling. In the upsampling stage, there are a large number of features channel which produce higher-resolution layers and, thus, the symmetry between the two parts producing the typical "U" shape.

The overlap-tile strategy also has another upside reducing the dependency of GPU memory of a network in order to increase the resolution.

Another critical solution is the ability to discern between background pixels and touching cell pixels by labels that allow the use of a differentiated weight loss.

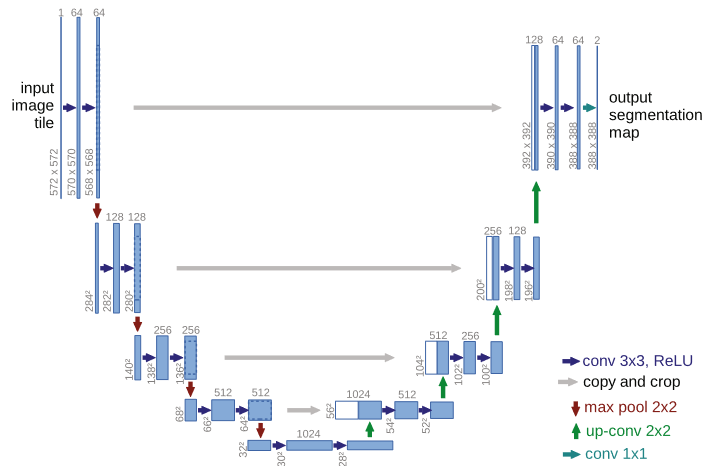


Figure 11, U-Net structure (Ronneberger, Fischer, & Brox, 2015)

The architecture (Figure 11, U-Net structure) is split into contracting path and expansive path; the contracting one consists of the repeated application of 3x3 unpadded convolution, ReLU, and 2x2 max pooling with stride 2 for downsampling; the expansive one consists of upsampling, 2x2 up-convolution concatenation with the corresponding cropped feature map and two 3x3 convolutions, each followed by a ReLU; finally, the structure is presenting a final layer 1x1 convolutions to map each feature vector to the desired number of classes.

This architecture is used for image segmentation after a training performed by images provided with a manual segmentation of them performed by an expert (Ronneberger, Fischer, & Brox, 2015).

1.6.7. Feature Extraction

Feature extraction is a process applied to raw data in order to obtain more manageable data characterized by a lower number of variables (dimensionality reduction). In wide datasets, the number of variables taken into account is an important aspect of computational time and power required. Feature extraction is a method that selects or combines variables into features; thereby, the amount of data is notably reduced, maintaining an accurate representation of the original set.

1.6.8. Decision Tree

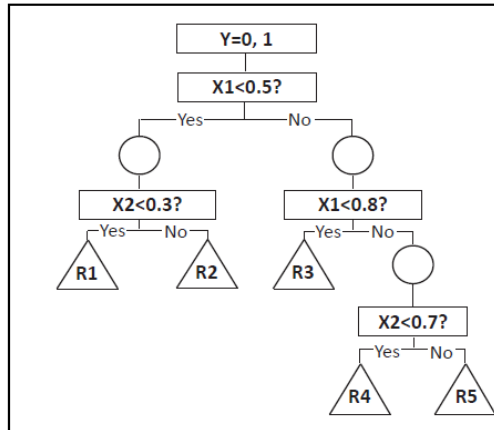


Figure 12, Example of Decision Tree structure (Song & Lu, 2015)

Decision Tree is a supervised classifier commonly used in data mining and based on multiple covariates; it classifies a population into branch-like segments starting from a root node, passing through internal nodes, and finishes with leaf nodes; the name of the method is reminded by the inverted tree shape that the structure assumes linking all the nodes (Figure 12, Example of Decision Tree structure). The non-parametric algorithm of the method allows to efficiently deal with large and complicated datasets without over-complicating the structure.

This method is commonly used for variable selection, importance variable assessment, missing values handling, prediction, and data manipulation.

The complexity of the structure has to be limited during its construction by stopping rules in order to avoid possible overfitting; nevertheless, there are different pruning methods to optimize the algorithm and solve the problem (Song & Lu, 2015).

Chapter 2: State of Art

Retinal fundus images have been widely used and analyzed by the medical community for the diagnosis of diseases, and thus, the approaches have changed over the years, fitting the best accessible resources in relation to the available solution.

Indeed, the problem studied has been analyzed and addressed in a wide range of ways, either supervised or unsupervised; in order to give a brief overview of the different approaches, six different methods related to segmentation of the retina, followed by subsequent analysis, that show notable results have been analyzed, comparing either chosen methodology and obtained results; moreover, the three cases are also proposed for the discrimination Plus and no Plus.

2.1. Segmentation of Retina Images

2.1.1. 2-D Gabor Wavelet and Supervised Classification

The first method (Soares, Leandro, Cesar Jr., Herbert, & Cree, 2006) analyzed is using a wavelet transform feature followed by a supervised classification; among the several available, a 2-D Gabor wavelet has been chosen; due to its directional selectiveness capability of detecting oriented features and fine-tuning to specific frequencies.

The method is preceded by a pre-processing that first is selecting the green channel since it is supposed to be the one with the best vessel/background contrast, then, in order to reduce the false detection of the border of the camera, is applying an iterative algorithm that removes the strong contrast between the retinal fundus and the region outside the aperture, thereafter, the exterior boarder is expanded by inclusion increasing the region of interest (ROI) determined by the camera aperture; finally, the green channel is inverted in order to enlighten the vessel.

Furthermore, given the dimensional nature of the features that might give rise to errors in the classification process, normalization is applied.

The final step consists of the application of a supervised classification in order to obtain the segmentation; the model is trained on a subset in order to reduce the computational cost of the manually segmented training set. Particularly, the classifier used is a Bayesian in which each class-

conditional probability density function is described as a linear combination of Gaussian functions, Gaussian Mixture Model (GMM). GMM represents a halfway between purely non-parametric and parametric models, providing a fast classification phase at the cost of more expensive training.

2.1.2. Trainable COSFIRE Filter

The second method (Azzopardi, Nicola, Mario, & Nicolai, 2015) is based on the model of a simple cell in visual cortex CORF (Combination of Receptive Field) and its implementation called Combination of shifted Filter Responses (COSFIRE); particularly is using a non-linear bar-selective COSFIRE that can efficiently detect bar-shaped structure and thus perfectly fit the problem studied. Furthermore, COSFIRE is a trainable filter whose selectivity is not predefined in the implementation but could be specified from the user in an automatic configuration process.

Also, in this case, there is a first selection of the channel, focusing on the green one; then the following pre-processing is applied: the first consists in the enhancement of the contrast of the vessels and the smooth of the border of the field-of-view (FOV) in order to reduce the possibility of false detection of vessels; second, the border is dilated replacing each pixel of the further layer with the mean value calculated on the inside neighbor; finally, a CLAHE (contrast-limited adaptive histogram equalization) algorithm is applied in order to improve the local contrast by avoiding over-amplification of noise in relatively homogeneous regions.

The filter is configured to be selective for a vertical bar that inputs the response of a center-on DoG (Difference of Gaussian) filter. Thereby, it denotes an excitatory central region and inhibitory surround; particularly, properly tuning the parameters is able to define strong intensity variation between areas described by the distance from the center. In order to focus the filter on a different orientation, the bar structure, used as an input, is modified in terms of orientation.

Finally, in addition to the symmetric B-COSFIRE (with the line point in the center), another one is added lying on end and thus described as asymmetric; this second allows to gather more information about the response between centers and endpoints that should be well defined but due to the noisy background are misrepresented.

2.1.3. Major Vessel Extraction and Sub-Image Classification

Methods based on an algorithm are divided into three stages (Roychowdhury, Koozekanani, & Parhi, 2015): first, two thresholded images are obtained by high-pass filter and by top-hat reconstruction of the red regions of the green plan; thereby, the common regions are defined as major vessels and the other as sub-image; second, the pixels of the sub-image are subjected to a 2-class classification; third, after a further enhancement, a segmented vasculature is created combining the major vessels and the vessels classified in the second stage.

Three more stages are required as pre-processing; the first is required in order to focus only in the retinal region and is achieved by superimposing a fundus mask to the green plan of the scaled (between 0 and 1) fundus image; the second is a contrast adjustment on the image obtained; finally, the third is a vessel enhancement that involves squaring each pixel intensity and re-normalizing the image.

For each pixel are studied, a set of neighborhood-based and gradient-based features and through a ranking of minimal-redundancy-maximal-relevance criterion (mRMR) and a second cross-validation criterion, they are ordered and selected in a final top 8 features.

2.1.4. Ensemble Classification-Based

This method (Fraz M. M., et al., 2012) is based on a 9-D features vector that contains information for each pixel and is chosen in order to eradicate dark and bright lesions; this vector is composed by the orientation analysis of the gradient vector that, through the normalized first-order approximation, is getting the distortion and thereby, the GOA map (I^o), morphological top-hat transformation opening using a linear structuring element-oriented (II^o), line strength measures computed for each pixel by calculating the difference in the average values of a square sub-window centered at the target pixel with the average value of the marked line for the highest value calculated (III^o, IV^o), a linear filter used for multiscale and multidirectional edge detection that acts as a low-level features extractor and background noise suppressor called Gabor Filter (V^o, VI^o, VII^o, VIII^o), intensity for the inverted green plan obtained without any pre-processing (IX^o).

An ensemble classification is then used by strategically generating and combining multiple classifiers in order to improve the performances and reduce the likelihood of a poor or unfortunate selection;

thereby, the method is combining the results of many weak learners into one high-quality ensemble predictor. The strategies chosen for the combination are two: Bagging, in which the decision trees are grown on the bootstrap replicas of the training dataset that are generated by selecting M observation out of N (total training dimension) with replacement, and all the predicted responses are then combined by taking a majority vote on their decisions; Boosting, that is also classifying by resampling the data and combining the voting by the majority, but that is building the strong learner focusing each repetition on the misclassification in order to reduce them (there are many version of this strategy and particularly this study worked with AdaboostM1 and LogiBoost).

Finally, binary vessel segmentation is obtained by the probability map, and the best strategy (LogiBoost) is selected by performances and cross-training performances.

2.1.5. Cross-Modality Learning

Unlike the conventional supervised method, this one (Li, et al., 2016) is re-molding the task of segmentation as a cross-modality data transformation problem in which the two modalities are color retinal image and vessel map; a mapping function is used to obtain the corresponding vessel map trying to minimize the loss function and the complexity of weight.

The patches are extracted from the green channel and directly transformed to vessel label patches without pre-processing. The first layer of the model is initialized by pre-training, and the other ones are randomly assigned.

A strong induction capability is required in order to simultaneously extract a complete vessel map, and the model is trying to fit autoencoder and RBM, that are normally used as denoising, are now used in order to transform the input in another space; it is achieved randomly selecting a fixed number of components and forcing their value to 0.

Thereafter, a synthesis strategy is used to construct the probability map; particularly, the pixel estimation is not supported by only the patch containing it but by multiple patches forming the image.

Finally, the binary vessel segmentation is obtained starting from the probability map; the optimal structure and parameters are obtained, evaluating the performances and the cross-training performances.

2.1.6. CNN – Deep Learning

This method (Liskowsky & Krawiec, 2016) is based on a convolutional neural network that is defined as a composite of multiple elementary processing units, each featuring several weighted input and output, performing convolution of input signals with weights and transforming the outcome with some form of non-linearity. Furthermore, it is characterized by particular characteristics in terms of spatial arrangement, units organized in sets of rectangular layers that makes it suitable for processing visual information; local connectivity, that with the application of a receptive field reduces the number of weights in comparison to the fully-connected conventional network; parameter sharing, the weights are shared across units in the same layers forming a feature map and thus reducing the number of parameters and making the extracted features equivariant; pooling of hidden layers, that aggregates outputs of multiple units by other means than convolution reducing the resolution and providing for translational invariance.

Particularly, this method focuses on a "Deep Neural Network" that has a consistent number of hidden layers.

The method uses two steps of pre-processing, Global Contrast Normalization, local brightness, and contrast normalization that abstract the learning process from fluctuations and help to focus on the vessel detections; Zero-phase Component Analysis, removal of universal correlations that strongly correlate pixels of the same structure, by multiplying the data matrix by a whitening matrix.

The images are also augmented in two ways, classic augmentation, randomly applying scaling, rotation, flipping, and gamma correction; extracting patches, thus feeding the algorithm with small and rectangular portions of the image.

The model is tested in many different configurations varying structure of the network as epochs, batch, patch, Dropout; the composition of the training set, balancing or not the origin of the data from multiple databases; the typology of prediction, implementing structured or not a structured prediction; evaluating the performance and differentiating it in terms of reduction of False Positive for more accurate segmentation of veins and increase of True Positive for more effective segmentation of thinner vessels as arteries.

2.1.7. Results obtained by previous methods

In order to give an idea of the different results achieved and thus, being able to properly evaluate the method developed, a common performance parameter has been chosen for the accuracy; the AUC (Area Under the Curve) ROC (Receiver Operating Characteristics).

Method	AUC ROC on STARE Database
2-D Gabor Wavelet and Supervised Classification (Soares, Leandro, Cesar Jr., Herbert, & Cree, 2006)	0.9671
Trainable COSFIRE Filter (Azzopardi, Nicola, Mario, & Nicolai, 2015)	0.9563
Major Vessel Extraction and Sub-Image Classification (Roychowdhury, Koozekanani, & Parhi, 2015)	0.9688
Ensemble Classification-Based (Fraz M. M., et al., 2012)	0.9768
Cross-Modality Learning (Li, et al., 2016)	0.9879
Deep Neural Network (Liskowsky & Krawiec, 2016)	0.9930

Table 1, AUC ROC on DRIVE Database with the different methods proposed in the state of the art

2.2. Plus and no Plus Discrimination

2.2.1. Automated Identification of Retinopathy of Prematurity by Image-Based Deep Learning.

This project is actually split into two different stages (Figure 13, Two stages representation), a first of classification between normal, mild, semi-urgent, and urgent ROP; a second aiming to identify the stage of the illness and evaluate the presence of Plus disease.

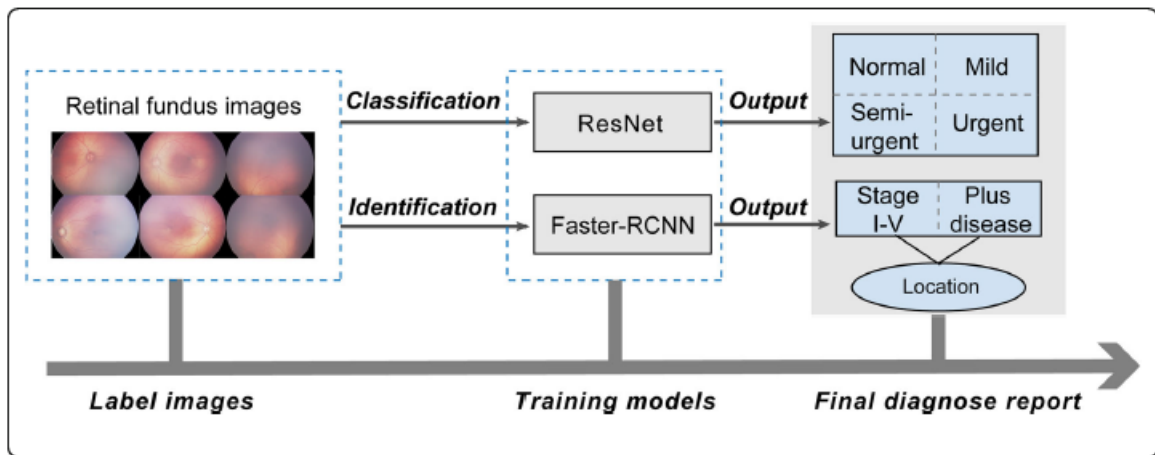


Figure 13, Two stages representation (Tong, Lu, Deng, Chen, & Shen, 2020)

This method is based on the use of CNN with a refined and pre-trained, by large and generic databases, architecture adjusted by the second phase of training with a smaller but specific database. The architecture used for this particular task of identification of the ROP stages and the presence of Plus disease is a Faster R-CNN with a pre-trained Inception-ResNet-v2 model; it was pre-trained on a large image dataset designed for object detection, COCO.

The total data used was a set of almost 39000 retina images labeled by 11 junior and two senior retina experts.

In order to augment the data, random noise and contrast adjustments were applied, and downsizing to the standard resolution was required.

The accuracy of this approach that is actually using a "black box" strategy is 0.896 (Tong, Lu, Deng, Chen, & Shen, 2020).

2.2.2. Plus disease in retinopathy of prematurity: pilot study of computer-based and expert diagnosis.

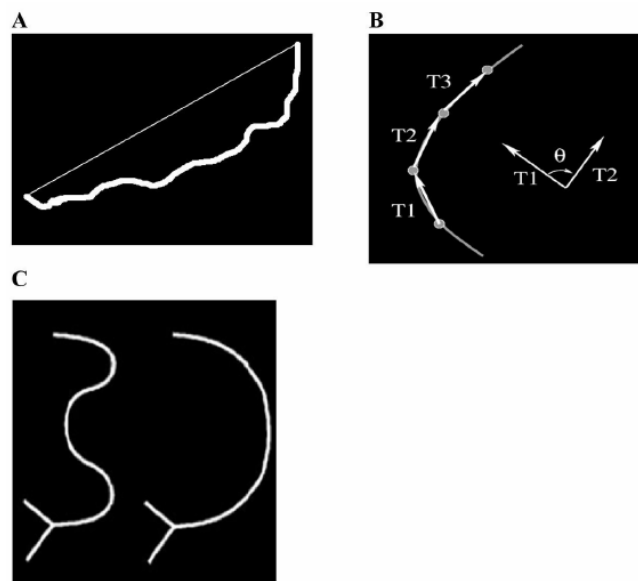


Figure 14, Tortuosity parameters representation, tortuosity index (A), Curvature index (B), and example requiring both for correct classification (C) (Gelman, et al., 532-540)

This paper is testing and comparing the evaluation of Plus and no Plus ROP images through a team of experts composed of pediatric ophthalmologists or retina specialists against a computer-based image analysis system, Retinal Image multiScale Analysis (RISA).

The dataset used is composed of 34 retinal images whose vessels were identified and classified by the four co-authors as arterioles or venules.

Each vessel was analyzed in terms of thickness, tortuosity, and curvature index (Figure 13, Two stages representation); furthermore, each image was associated with the relative mean values of the three parameters.

For each vessel was calculated the accuracy and then the accuracy of the model was obtained; for this set of images, the result was 0.938 (Gelman, et al., 532-540).

2.2.3. Automated diagnosis of plus disease in retinopathy of prematurity using deep convolutional neural networks.

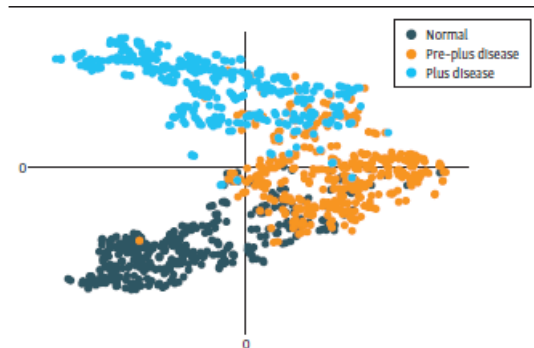


Figure 15, Normal, Plus and Pre-Plus Rop disease clustering (Brown, et al., 2018)

This method uses two different Convolutional Neural Network, a first with U-Net architecture for vessel segmentation and a second with architecture Inception 1 for the diagnosis of Plus, pre-Plus, and no Plus disease (Figure 15, Normal, Plus and Pre-Plus Rop disease clustering).

Indeed, the method starts with a classic segmentation stage but is followed by a black-box approach in the second stage.

The first is trained with 6000 labeled images in order to obtain a 0-1 pixels segmentation of the vessels; the second is pre-trained on a database of 1.2 million images from 1000 classes and then training with the treated images.

The result was cross-validated against eight international ROP experts on an independent test set of 100 images.

The accuracy was averaged between 5 different cross-validations and reached a value of 0.98 (Brown, et al., 2018).

Chapter 3: Methodology

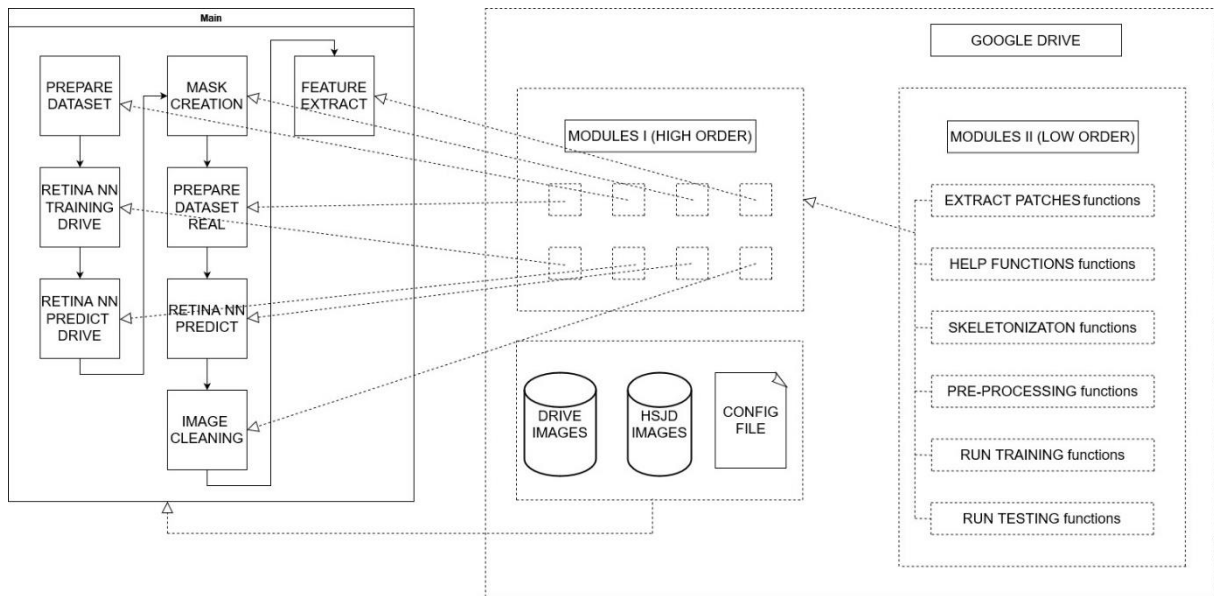


Figure 16, Block Diagram Code

3.1. Main and Modules Structure

The code is conceptually divided into two parts, main and modules; the main is the principal script that is run in order to obtain all the results and data aimed (Asole, 2021); on the other hand, the modules are a wide range of scripts that are used as tools, during the running of the main, in order to compact the code and call functions or lines of codes that are recurrently or iteratively processed. Besides, there are two typologies of modules that is possible to find, modules containing a script that, whenever are called from the main, are throughout processed and that are not requiring to input any parameter as a normal high order function (type I^o); modules containing groups of low order function such as utils that need to be initially imported in order to successively be able to singularly call each of them (type II^o)(Figure 16, Block Diagram Code).

The list of the modules of the type I^o (Asole, 2021) is composed by:

- prepare_datasets_DRIVE.py
- retina_NN_training_DRIVE.py
- retina_NN_predict_DRIVE.py
- mask_creation_REAL.py
- prepare_datasets_REAL.py
- retina_NN_predict_REAL.py

- Images_Cleaning.py
- Masks_creation.py
- Feature_Extract.py

Those modules help to split the solution of a complex problem into different feasible tasks and solving it step by step. Indeed, it is possible to notice as those are not only remarking the stages of processing labeled data (DRIVE Dataset), new not labeled data (REAL dataset), and predicted data; but also, the sub-stages of preparation, training, and prediction.

The list of the modules of the type II^o (Asole, 2021) is composed by:

- extract_patches.py
- help_functions.py
- skeletonization_functions.py***
- pre_processing.py
- run_testing.py
- run_training.py

On the other hand, those modules are providing functions to specific tasks that are complex and require to include a wide range of tuning for the same function, depending on the specific case, such as the extraction of patches that change not only for labeled and not labeled data processing, but also, for example, for the stage of training or testing and for the option of overlap patches or not; furthermore, those modules help to compact the code calling functions or lines of code that are not only iteratively used for each image but also utilized in different stages, such as the pre-processing ones.

Each module has a common structure that is composed by:

- I. Importation of the required libraries.
- II. Definition of the required path in order to interact with files and other modules in the drive.
- III. Definition of the necessary functions.
- IV. Script run in the module (only for modules of the I^o type).

Finally, a further file is created and used by this system that is neither a module of the first nor the second type, the "configuration.txt" file (Asole, 2021); this text file is basically listing by categories all

the paths and the parameters that are required by modules and that, in an automatized logic, are automatically called from this file and easily adjusted whenever the root folder of the whole structure is changed, or any parameter or dataset is modified.

Overall, the main script is performing the following steps:

- 1 Preparing the data for the training and test necessary in order to tune the network.
- 2 Prepare and process, by the network, the “real images”, meant as images not provided in a public database but collected from patients of the hospital involved.
- 3 Clean the segmentation obtained and extract the two required features, tortuosity, and thickness.
- 4 Used the first set of images containing ROP Plus and no Plus cases, through points 2 and 3, in order to train different supervised classifiers.
- 5 Evaluate the classifiers in order to see the quality of the result.

3.2. Connectivity and Modules

In order to be able to use the set of tools described, it is necessary to initially mount, and thus connect to the main script, the Google Drive containing all the modules; indeed, for the specific runtime of the script, is required to generate and verify an authorization code that allows to read, write and move files stored in a particular Google Drive account; furthermore, it is possible to connect the run to the whole drive or specify only the folder targeted specifying the root direction.

Achieved the general connection with the drive; it is necessary to specify all the module file names required during the run of the main script and to link them to their relative directions starting from the root folder specified in the previous stage.

The third stage is diversified between the two categories of module previously defined; with regards to the module of functions (type II^o), it is now necessary to import them in order to make them available at any further call in the script; on the other hand, the other typology (type I^o) has to be imported only when the targeted script is required to be run.

The whole structure described is grouped in the first block of code, "Connectivity and Modules."

Besides this, three other blocks are developed in the main script, "Model Preparation," "Real Image Prediction," and "Parameters Procedure," that are using the provided tools in order to properly develop the process.

3.3. Main Challenges

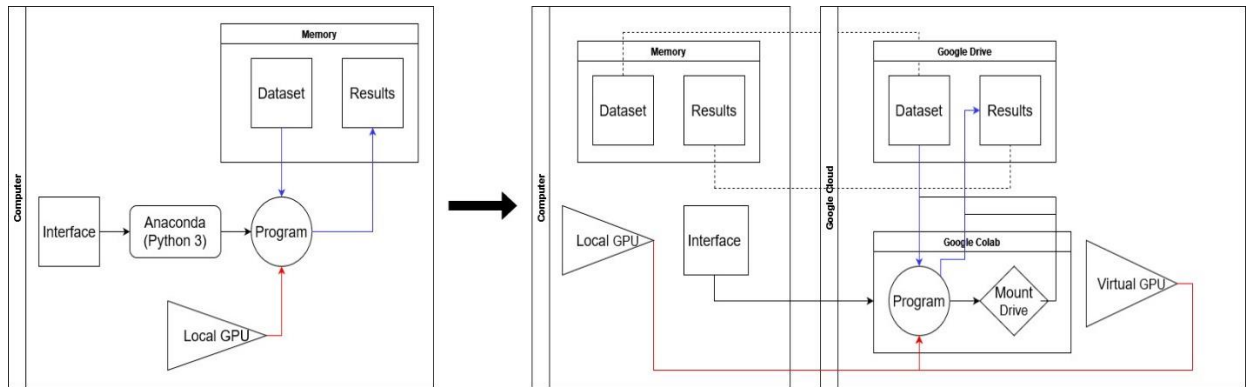


Figure 17, Colaboratory adaptation of standard python structure

The main challenge of this first stage is the definition itself; the GitHub code, used as starter point, is meant to run on "python 3" and not on Google Colaboratory; indeed, the main difference is that all the connections that normally are made by folders in the physical computer, have to be rewritten in order to work on a Google Drive account (Figure 17, Colaboratory adaptation of standard python structure); furthermore, all the function and packages has to be checked and

3.4. Model Preparation

3.4.1. Data Preparation

In this sub-block, the first module of the type I^o, "prepare_datasets_DRIVE.py", is run by importing it; it represents a pre-stage in which the data contained in the driver is prepared and compacted in order to obtain a result that is easily processable by the following training.

After importing the required libraries, the paths for training and test sets are properly defined, and the parameters related to their processing are extracted from the configuration file.

Then, the function "get_datasets" is defined; this function allows to inspect the directories of ground truths, images, and masks, creating NumPy arrays with all the images found inside and, using the function "write_hdf5" (help_functions.py) to create an hdf5 file for each of them; thereby, it creates a file easily processable with the data contained in the drive.

The function "write_hdf5" allows to re-organize heterogeneous, large, and complex data in a format that is self-describing and supports Data slicing; indeed, the function allows to compact the dataset of images, such as a training label or ground truth, to Hierarchical Data Format version 5 files and thereby, makes available, throughout the script, the extraction of only targeted portions of data without having to completely read the dataset with the risk of saturating the memory.

Finally, in the script, the function defined is iteratively run in order to build the three files, image, mask, and ground truth, for both test and training sets.

3.4.2. Training

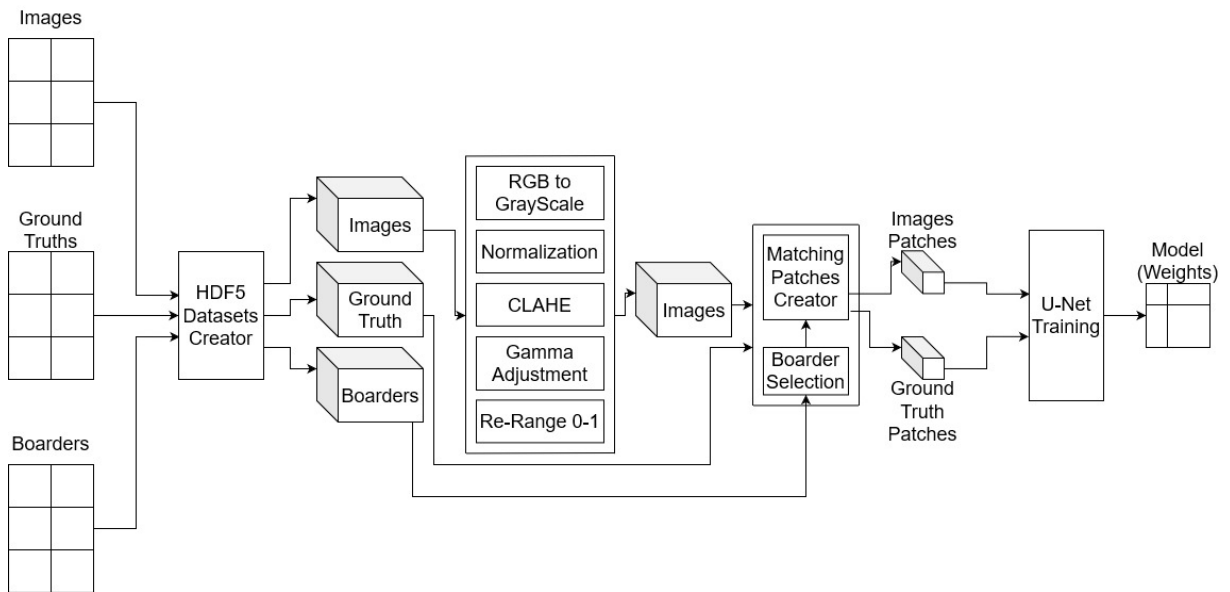


Figure 18, Training workflow

The second module of the first type described, "retina_NN_training_DRIVE.py," is imported and thus runs in this stage in order to create the network and feed it with labeled data in order to train it (Figure 18, Training workflow).

The first step is importing all the required libraries and, in addition importing, from the connected modules, the necessary parameters and functions.

Hereafter, the model is defined in the function required for the training, "get_unet"; the architecture is of a deep convolutional layer U-Net (Table 2, U-Net Structure), indeed, with convolution and expansion stages. The function requires the number of channels and the dimension of the patches that are going to feed the training of the network; inside, all the different layers are defined and connected from input to output. The model is compiled assigning as an optimizer, in order to decide how to deal with the gradient between result and ground truth, the "stochastic gradient descent" (SGD), without any tuning; as loss function, in order to evaluate the mistake of the model, is used "categorical cross-entropy" and as a metric, "accuracy."

Stage	Layer	Dimension	Characteristics
Convolutional 1	Convolutional 2-D	$32 \times (3 \times 3)$	Act = "ReLU" pad = "same"
	Dropout 20%		
	Convolutional 2-D	$32 \times (3 \times 3)$	Act = "ReLU" pad = "same"
	Max pooling 2-D	(2×2)	
Convolutional 2	Convolutional 2-D	$64 \times (3 \times 3)$	Act = "ReLU" pad = "same"
	Dropout 20%		
	Convolutional 2-D	$64 \times (3 \times 3)$	Act = "ReLU" pad = "same"
	Max pooling 2-D	(2×2)	
Convolutional 3	Convolutional 2-D	$128 \times (3 \times 3)$	Act = "ReLU" pad = "same"
	Dropout 20%		
	Convolutional 2-D	$128 \times (3 \times 3)$	Act = "ReLU" pad = "same"
Expansion 1	Up sampling 2-D	(2×2)	
	Concatenate		with "convolutional 2"
	Convolutional 2-D	$64 \times (3 \times 3)$	Act = "ReLU" pad = "same"
	Dropout 20%		
	Convolutional 2-D	$64 \times (3 \times 3)$	Act = "ReLU" pad = "same"
Expansion 2	Up Sampling 2-D	(2×2)	
	Concatenate		with "convolutional 1"
	Convolutional 2-D	$32 \times (3 \times 3)$	Act = "ReLU" pad = "same"
	Dropout 20%		
	Convolutional 2-D	$32 \times (3 \times 3)$	Act = "ReLU" pad = "same"
Expansion 3	Convolutional 2-D	$2 \times (1 \times 1)$	Act = "ReLU" pad = "same"
	Core Reshape	$(2 \times H_{patch} L_{patch})$	
	Core Permute	(2×1)	
Final	Core Activation		Fun = "softmax"

Table 2, U-Net Structure

Then, by the function "get_data_training" (extract_patches.py), the model starts loading the training images and ground truth, obtaining from them the patches required in order to feed the network; besides, a sample of them is saved in order to visualize them with the results.

"get_data_training" is a function that, given the information relative to the path of training datasets and the parameters of the patches targeted, starts loading the images by the function "load_hdf5" (help_functions.py), pre-treats them by "my_PreProc" (pre_processing.py) and then reduces the

range of values of the pixel from between 0 and 255 to between 0 and 1. Next, the images are cut evenly on the bottom and top in order to obtain squared and centered resultant sets of images; Finally, the patches are randomly extracted by the function "extract_random" (extract_patches.py) and the consistency between the ones relative to images and masks is checked by "data_consistency_check" (extract_patches.py).

"load_hdf5" is a function that by h5py allows loading the hdf5 files previously created and stored by the function "write_hdf5".

"my_PreProc" is a function that first checks the correctness of the data passed, in terms of dimensions and shape; next, converts them in greyscale and by the functions "dataset_normalized" (pre_processinging.py), "clahe_equalized" (pre_processinging.py) and "adjust_gamma" (pre_processinging.py) treats them in order to enhance, as much as possible, the differences in contrast and shades that are then analyzed by the network in order to segment the images (Figure 19, images and relative histograms through the pre-processing, gray scale conversion, normalization, CLAHE filter, gamma adjustment filter). This set of functions, after checking the shape and length of the data that has to process, for each image of the array, reduces the range of values of the pixels between 0 and 1 and then translates the minimum to 0; then, splits the images into small blocks and amplify the noise found between defined limits; finally, it enhances the contrast between relevant details expanding the difference of shade of the ranges desired.

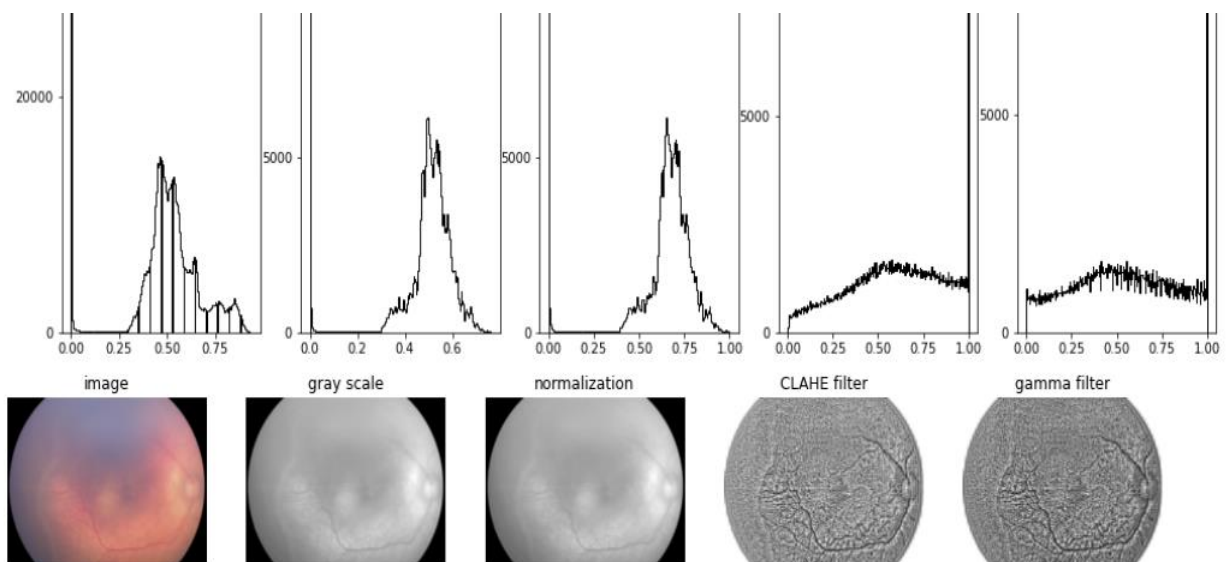


Figure 19, images and relative histograms through the pre-processing, gray scale conversion, normalization, CLAHE filter, gamma adjustment filter

"extract_random" is a function that, after checking if the operation is feasible due to the number of channels and dimensions of patches and images, randomly extracts an even number of patches for each couple image-ground truth and checks that each of them is contained in the Field of View (FOV) by the function "is_patch_inside_FOV" (extract_patches.py) that simply analyses the dimensions of the patches required and the standard dimension of the FOV (that represents the area of the image targeted by the network and that could mask out by a "BOARDER" mask image).

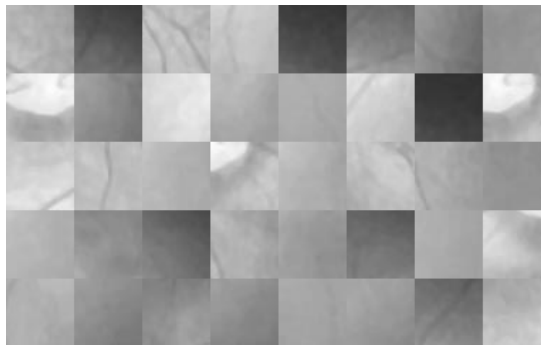


Figure 20, Patches example

Obtained the patches (Figure 20, Patches example), the model is built running the model function previously defined; furthermore, the structure is saved, and a checkpoint is defined in order to save and store the best weights run by the network during the training.

Finally, the ground truth patches are reshaped in order to reduce memory consumption by the function "masks_Unet" (help_functions.py), which reshapes the array of images in a matrix by stretching each image in a vector, then a double and opposite layer of 0 and 1 is created representing on the first "channel," the pixels of the ground truth and on the second, everything out of it; finally, the model is fit, saving also the last weights obtained.

3.5. Prediction and Performance Evaluation

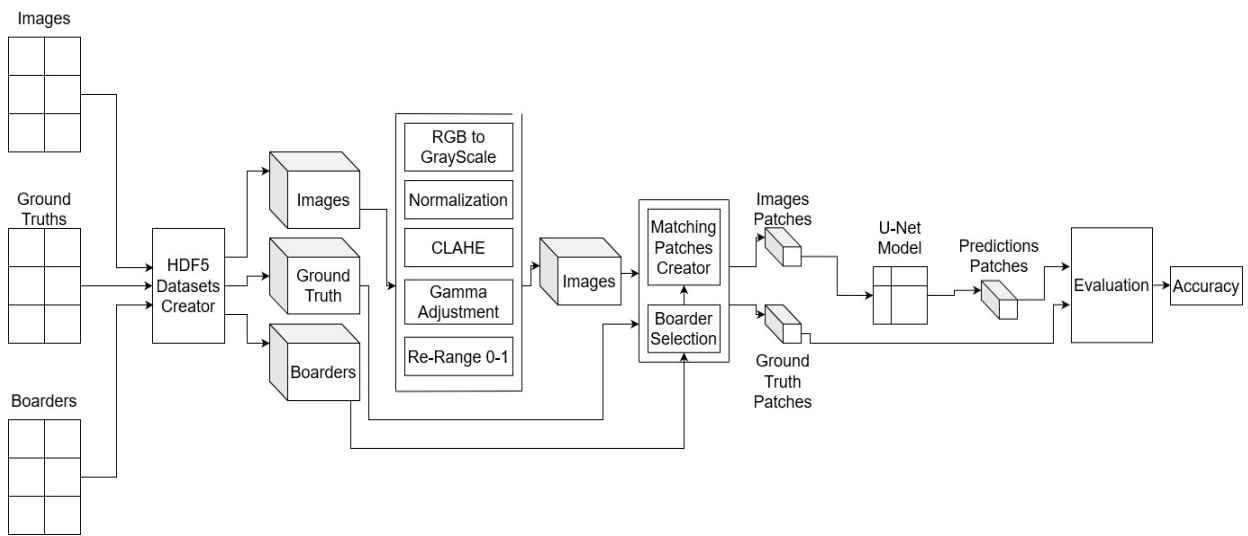


Figure 21, Test workflow

In this sub-block, "retina_NN_predict_DRIVE.py," another module of the first type described, is called, and it uses the network previously created and trained in order to make a prediction and test the quality of it (Figure 21, Test workflow).

The first step is importing all the required libraries and extracting parameters, images, and functions from the other linked files; particularly important, in this case, are all the parameters and weights obtained and stored from the previous training stage.

Next, the original data for the testing is load and processed in order to obtain the patches required for the test; specifically, the average model is imposed to True, and thus, the images can overlap by creating a new patch for each stride movement and not at the end of the previous one; indeed, the set of functions used are the ones customized for the "overlap case," and the function used for the patches' creation is "get_data_testing_overlap" (extract_patches.py). This function is following a chain of calls of function very similar to the one explained for the training; though, there is a substantial difference for the final extractor of patches that are not randomly working but dividing the images systematically, taking into account the dimensions and the striding parameters imposed in order to overlap; the new function' name is "extract_ordered_overlap" (extract_patches.py). Furthermore, this function is adding a check function that in case the numbers of pixels for image and patch are not compatible with the striding imposed, it is reframing the image with extra lines of

black pixels, the function version used for this overlapping case is called "paint_border_overlap" (extract_patches.py).

Thereafter, the model structure is loaded and tuned by the best weight obtained in the previous step of training; thereby, the prediction is obtained and properly reshaped by the function "pred_to_imgs" (help_functions.py) that reshapes each image from a vector a matrix adding the proper position for the number of channels.

Next, the prediction patches are used to recompose the images by the function "recomponer_overlap" (extract_patches.py) in the same logic that was used to obtain them; the original images are pre-processed by the function "My_PreProc" and the ground truth loaded.

Afterward, the masks are applied to the images in order to focus on the FOV required by the function "kill_border" (extract_patches.py); this function is inspecting all the pixels of each image and, by a supportive function "inside_FOV_DRIVE" (extract_patches.py), is checking if the relative pixel is out of the FOV defined by the relative mask; in this case, it imposes the value 0 to the pixel of the image.

Successively, image, prediction, and ground truth are reduced to the original dimension, getting rid of possible extra frames obtained during the previous processes.

Thereafter, the evaluation of the results is made; using a mask and ground truth, every pixel of an image is evaluated, and the overall statistic is studied with the use of different algorithms and methods, Area under the ROC; precision-recall curve, confusion matrix, Jaccard similarity index, F1 score. Particularly the function "pred_only_FOV" (extract_patches.py) is creating two new arrays containing only the pixels of the prediction and the ground truth that are contained in the relative mask and, thereby, allows a possible comparison not affected by anything out of the FOV desired.

Finally, all the different parameters calculated are grouped and saved.

Moreover, throughout the code are inserted a series of checkpoints in which dimensions and types of variables are inspected in order to be able to detect and also spot possible mistakes.

3.6. Prediction for Real Images

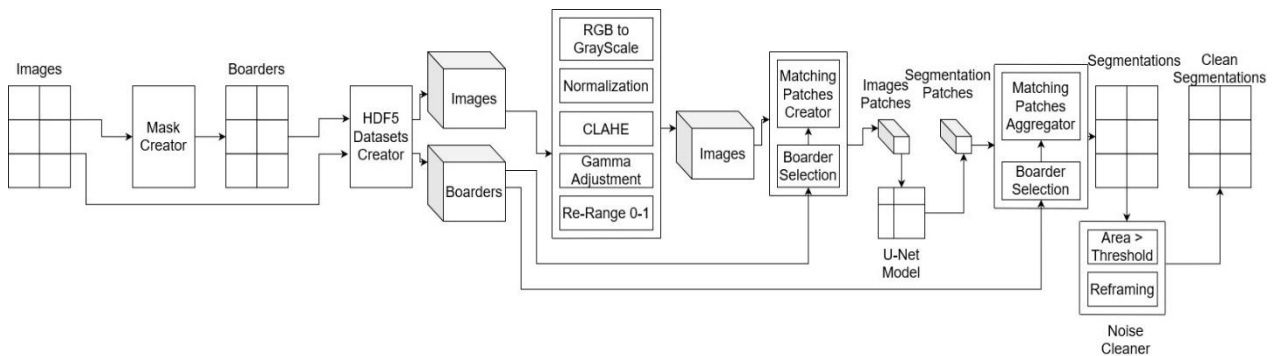


Figure 22, Predicted segmentation workflow

The process of prediction is very similar to the previous explained; nevertheless, it is passing from images provided of label and mask to images lacking any additional information; indeed, further pre-processing is required (Figure 22, Predicted segmentation workflow).

3.6.1. Mask Creation

At this stage, the module "Mask_Creation.py" is imported and used in order to create a set of masks for the new images that are provided without mask (BOARDER) or ground truth.

After the required imports of the libraries, the patterns are also imported and defined from the configuration file; in this case, they are targeting not the previous folder with the DRIVE images but another one filled with images belonging to the database shared between the Spanish Hospitals.

First, each image in the folder is imported, converted to greyscale, and changed of the format in order to be able to use particular commands and functions; second, a new destination direction path is created in order to be able to store it; third, the mask is created separating all the black pixels from the other shades without discriminating between the retina and the rest of the image in the Field of View. Two approximations are applied (Figure 23, Mask creation by first and second approximation); first, the black pixels, theoretically represented by the intensity 0, are considered between 0 and 1 in

the pixel light scale that goes between 0 and 255; second, an erosion filter is applied in order to slightly reduce the dimension of the field of view. Both the approximations are made in order to try to define a border that actually is not clean and stark but though avoids its classification that is very complex and focuses on the more central pixels that are the ones in which the structures studied are contained.

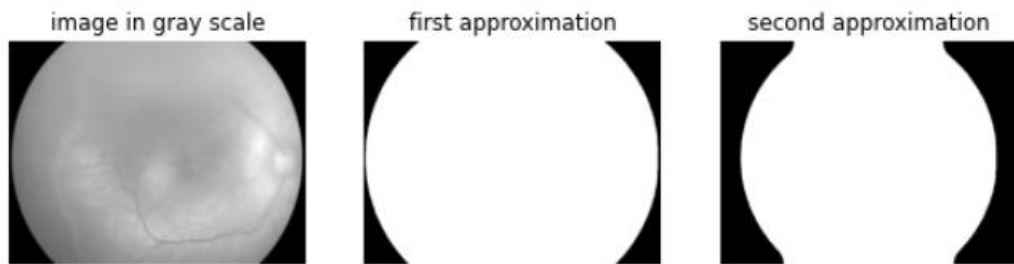


Figure 23, Mask creation by first and second approximation

3.6.2. Data Preparation Real

In this stage, the module "prepare_datasets_REAL.py" is imported in order to prepare the new data to be processed in a similar way to the previously explained process used for the labeled data.

With "Real data" are called all the images referring to patients from the hospitals and not to public databases used in phase of training

Basically, in this sub/block, the process, previously explained, is repeated targeting the new images analyzed; nevertheless, in this case, the files produced are only two because the previous step was managing test and training images, and they all have ground-truth, but now there is one set of images provided only with the just created masks.

3.6.3. Prediction Real

Finally, the module "retina_NN_predict_REAL.py" is run in order to get now the prediction of not labeled data; indeed, the purpose of the prediction is not anymore essaying the quality of the

network built and trained but exploiting the tool created in order to obtain new reliable data used for following stages.

Also in this case, the process is similar to the previous one with small differences due to a change of data and parameter targeted and the absence of any ground truth or label.

Indeed, the first difference lies in the paths that address the REAL dataset with its parameters but still targeting the weight obtained with the training of the previous data; following the chain of calls of function; it is possible to see that also the function used in order to create the patches is customized for this particular stage, "get_data_testing_overlap_real" (extract_patches.py) is basically the same function of the previous block but is focusing only on the images patches and not on the ground truth ones. Finally, with respect to the previous module, after the actual prediction, there is no evaluation through any algorithm because of the lack of any ground truth to use as a benchmark.

3.7. Main Challenges

The main challenge of this stage is to develop the GitHub code in order to integrate it into the project in development.

- 1) The code, as in the previous stage, has to be adapted for Google Colaboratory
- 2) The code, in terms of the input stage and the function, has to be modified in order to work not only with the image dimensions and format of the particular drive used during the training (as in the GitHub version) but with any image passing them through all the stages and functions without losing any sensible data.
- 3) Lack of mask for normal retina images.

3.8. Feature Extraction (Classifiers Training)

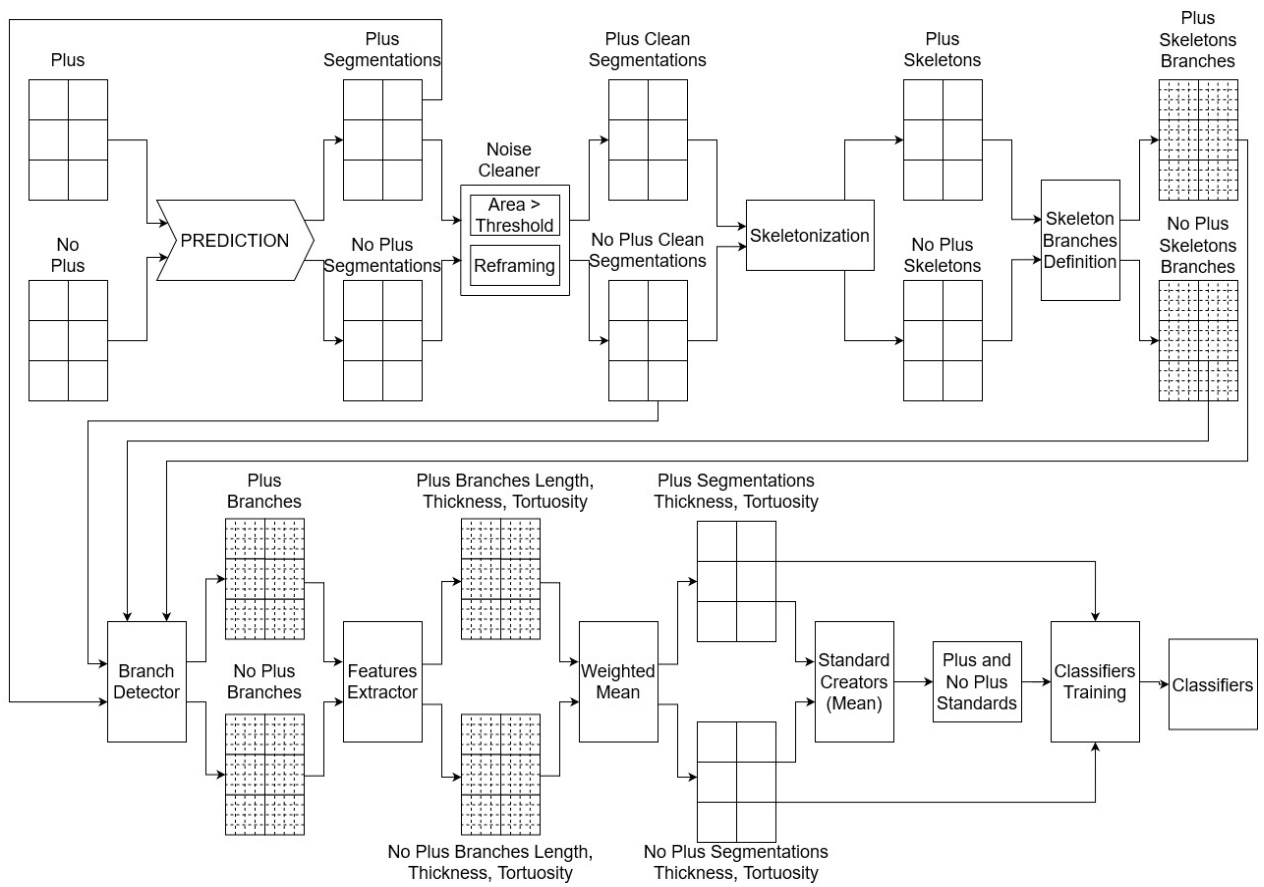


Figure 24, Standard creation workflow

In this stage, the segmented images are pre-processed and used in order to extract the required parameters and thereby create a standard for each category that is going to be applied as a benchmark in order to successively classify new images (Figure 24, Standard creation workflow).

3.8.1. Rumour Clean

The process of cleaning of the images from any noise (Figure 25, Cleaning Process) that could negatively affect any further study is applied by importing and, thus, running another module of the first type, "Images_Cleaning.py."

This sub-block, for each image found in the specified folder, extracts the image, creates another destination direction, applies a customized function with the properly tuned filters, and saving it. Thereby, another result images folder is created to store the images previously segmented but properly cleaned from rumours.

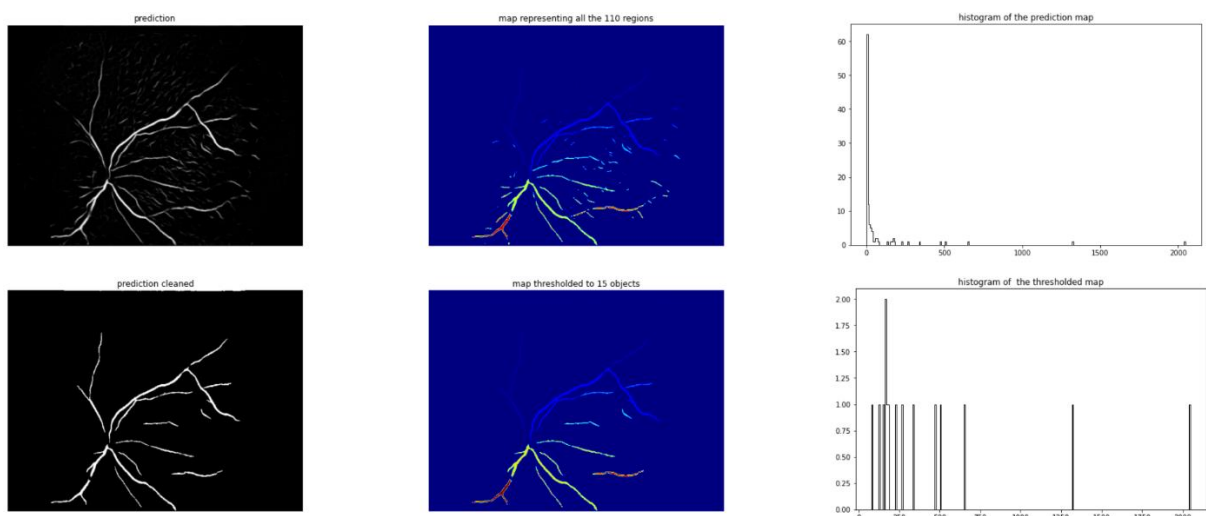


Figure 25, Cleaning Process

The function used in order to clean the image is "img_clean" (skeletonization_functions.py); it starts converting the image in a proper greyscale format and applying an Otsu threshold in order to detect as many regions as possible; next, it differentiates them by a labeled map and, by a threshold, erase from the image all the regions with a low area that are the results of a small portion of pixels produced by noise and thus, negligible. Finally, it substitutes the frame of the image with a black border of pixels in order to avoid possible misdetection mainly due to the application of the mask, in order to focus in the correct area of the image, and also negligible due to the location.

3.8.2. Skeletonization

Finally, the last module of the type I^o "Skeletonization.py" is run in order to get all the information required from the segmented images obtained.

The first step is importing libraries and functions required; particularly in this block, the functions used are not defined with the purpose to help the training and the segmentation but instead to help the skeletonization and the extraction of the features; indeed, the functions targeted are the ones stored in the "skeletonization_function.py" module.

Second, the images, previously cleaned from rumors, are acquired and stored in two arrays discerning between images of retinas affected by the disease in a "plus" stage (ROP) and not (NO-ROP) through the function "img_acquisition" (skeletonization_function.py).

"img_acquisition" is a function that, given a path, inspects the targeted folder storing all the images found in an array that is finally returned after the proper pre-processing achieved by the function "prep" (skeletonization_function.py). "prep" is a compact function that normalizes the images between 0 and 1 and then, with a threshold, assigns all the lower values to 0 and all the high one to 1.

Next, two basic datasets are created, one for ROP and one for NO-ROP; both are created, passing through every image contained in the relative array and storing the targeted information in the dataset.

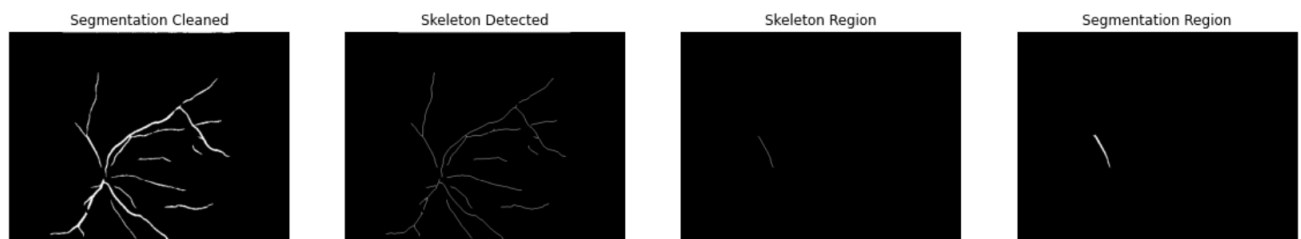


Figure 26, Skeletonization process through the stages of segmentation, skeleton detected, skeleton region, segmentation region

By the function "skeleton_description" (skeletonization_function.py), first, from each image is extracted the recognized structure, that means a branched structure without thickness, representing the skeleton of the previous segmentation obtained; then, the structure is split into single segments by the intersection points that are used to trace the shortest path along with the skeleton; next, with

the function "SelectionArea" (skeletonization_function.py), from the segment of the skeleton, the relative area of the segmentation is obtained (Figure 26, Skeletonization process through the stages of segmentation, skeleton detected, skeleton region, segmentation region). This data is then used in order to create an intermediate array with the morphological information of each area of the segmentation. Finally, this array is used to fill another one with only a subset of the obtained parameters, identification number of the image, identification number of the area, tortuosity, thickness, centroid, ROP/NO-ROP class; ruling also out the areas with a low extension and hence, low impact.

"SelectionArea" is a function that fed with a segmented image, a skeleton segment, and the starter point of it, is getting a first check selecting by a logical "and-function" only the pixels part of the segmentation and dilation of the skeleton segment and then, a second check by flood filling the targeted region with a pointer, created by the function "Start_flood" (skeletonization_function.py), in order to avoid any possible overlapping of the dilated area with unconnected and close areas. "Start_flood" is a function that, given a skeleton segment and a start point of it, select the closest point to the start one, along the segment in order to avoid that a center spotted between two pixels could lead to an approximated center out of the segment and thereby a consistent mistake in the flood filling stage (Figure 27, Two stages segmentation region definition: Segmentation overlapping, Flood filling).

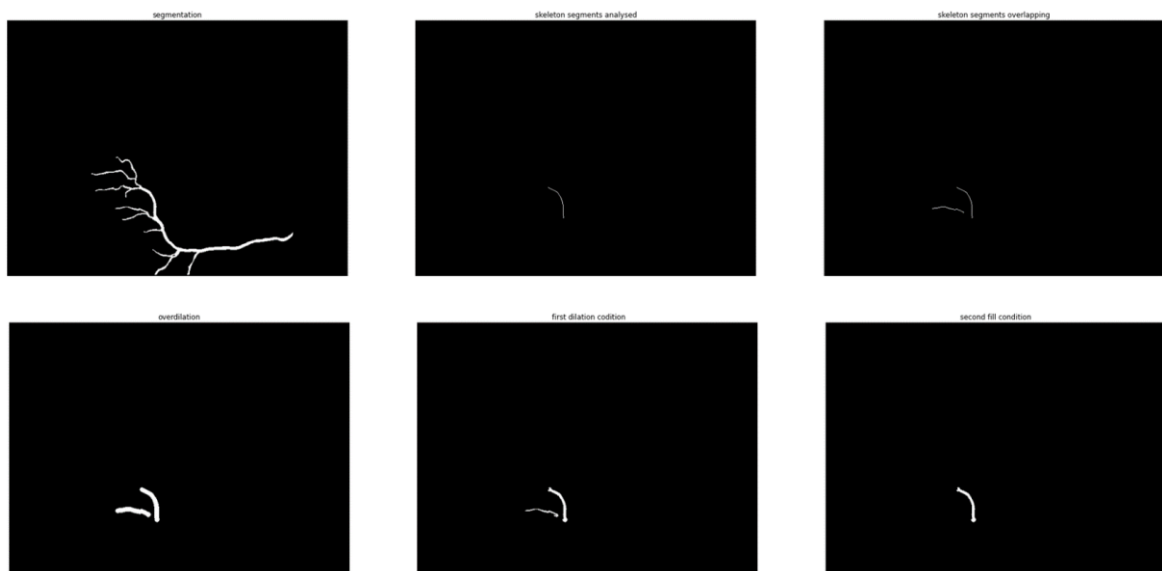


Figure 27, Two stages segmentation region definition: Segmentation overlapping, Flood filling

All the new arrays are next compacted in the two datasets; indeed, the arrays obtained are two lists of all the segments information obtained from the images, for both categories, organized by identifier numbers of the image and segment of structure.

Another version of the datasets is then created, expanding the dimension in order to allow a visual representation of the studied characteristics in terms of quantity; each segment is replicate as many times as its length.

Thereafter, a complete dataset is created merging the two previously obtained, and by those three datasets, some standard ranges are defined between the two categories in order to be able to discern the two classes without any ground truth provided.

Those standards are first analyzed in order to see any possible overlapping and, thus, mistakes, and after testing in order to be able to assign an accuracy to the classification tool provided (Figure 28, Standard classifier evaluation workflow).

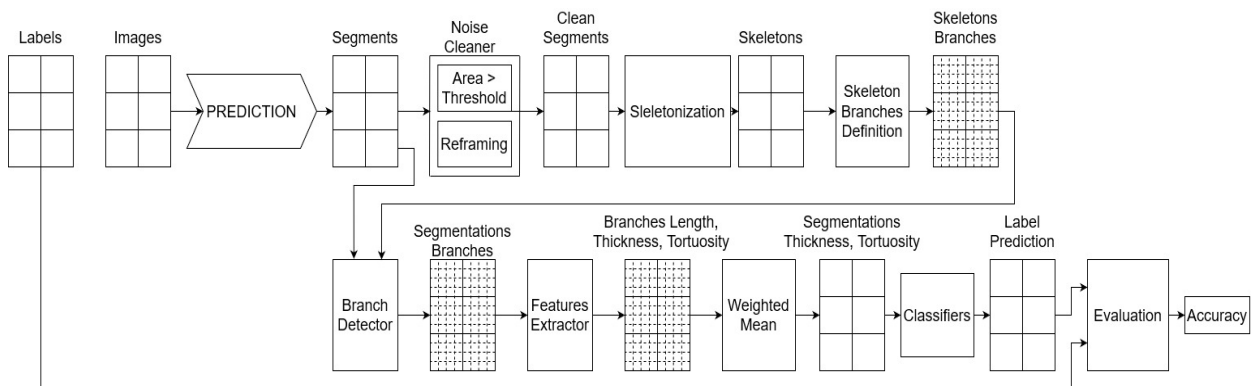


Figure 28, Standard classifier evaluation workflow

3.9. Main Challenges

- 1) Selection of the main vein and arteries, getting rid of the noise or incomplete segmented branches.
- 2) Repartition of the structure in sub-branches
- 3) Definition of the area of a heterogeneous range of sub-branches with a sufficient level of accuracy but without incurring in overlapping between different ones.
- 4) Definition of tortuosity and thickness.
- 5) Integration of different data representations such as datasets, masks, images
- 6) Data interpretation and classifier selection.

3.10. Classifiers Training

In order to define the train the classifiers, and thus define a standard that is going to be used to compare new images, four different analyses are developed from a pool of 20 ROP plus and 20 ROP no plus cases; the different images are analyzed by tortuosity and thickness of the branches that are weighted in order to obtain mean values. The four analyses are:

- Double Threshold: the most restrictive analyses that classifies as ROP plus only the values with both the parameters greater than the greatest values found in the standard pool (Figure 29, Double Threshold).

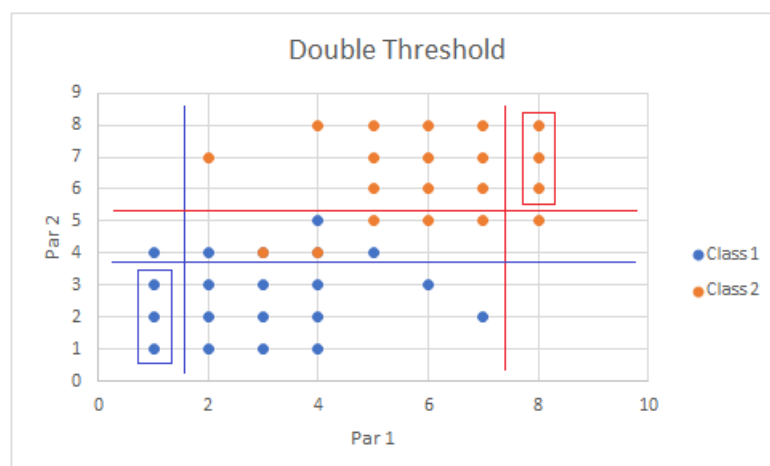


Figure 29, Double Threshold

- Mean Distance Threshold: this method classifies ROP plus and not using a threshold that is selected as the mean point of the average values of the two different classes in terms of both the parameters used (Figure 30, Mean Distance Threshold (Own Production)).

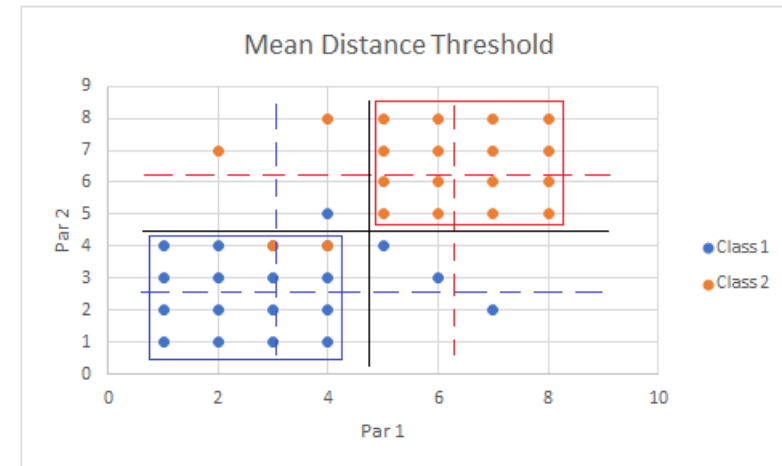


Figure 30, Mean Distance Threshold

- Smart Threshold: this classifier is similar to the previous one but, instead of using the mean point of the averages, is calculating the two thresholds that are the best classifying the pool used for the standard creation (Figure 31, Smart Threshold (Own Production)).

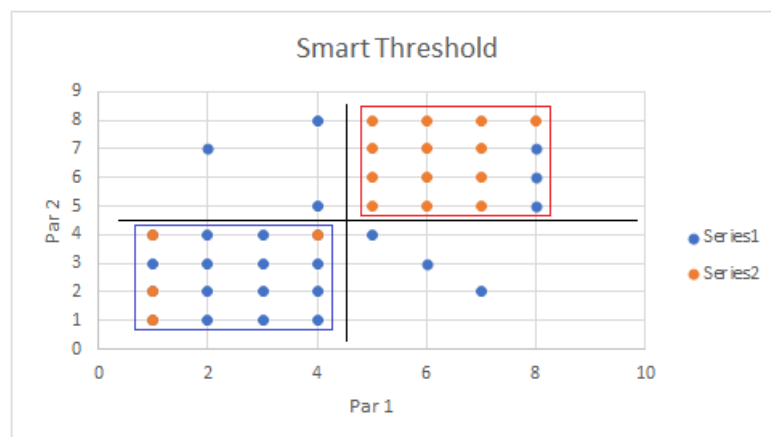


Figure 31, Smart Threshold

- % Threshold: this classifier is a refinement of the Mean Distance Threshold that aims to classify all the images not detected because with opposite results in terms of tortuosity and thickness; therefore, it calculates the percentage distance for both the parameters in order to classify by higher proximity (Figure 32, % Threshold).

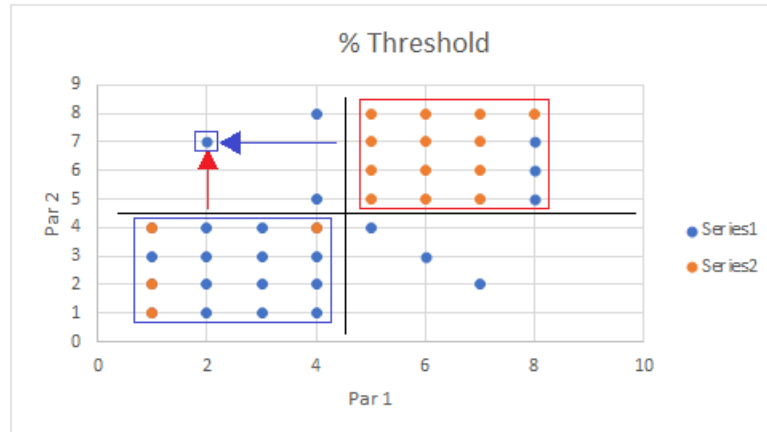


Figure 32, % Threshold

- Decision Tree: the last method consists of the application of a decision tree in order to have a more complex and refined comparison (Figure 33, Decision Tree(Own Production)).

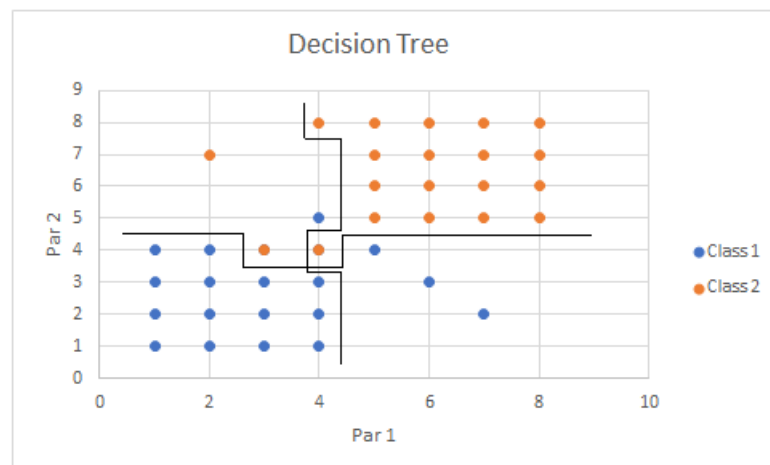


Figure 33, Decision Tree

3.11. Performance Evaluation

In order to evaluate the classifiers, six pools of balanced data, each composed of 20 ROP plus images and 20 NO ROP images, are selected and tested. The first three image pools are selected in order to test if the quality of the image, in terms of noise contrast and overall clarity (Figure 34, Image examples of the Low quality (A), Medium quality (B), and High quality (C) pools(Own Production)), are strongly affecting the results; the second three are randomly selected by the pool of data in order to test the robustness of the classification.

The Pool of data used to train the classifiers is quite limited; nevertheless, it has been used only as a preliminary study in order to assess the feasibility of the classification; also due to the computational time necessary for the feature extraction. Hence, a wider pool of data will be processed in order to evaluate the result in future stages of the project.

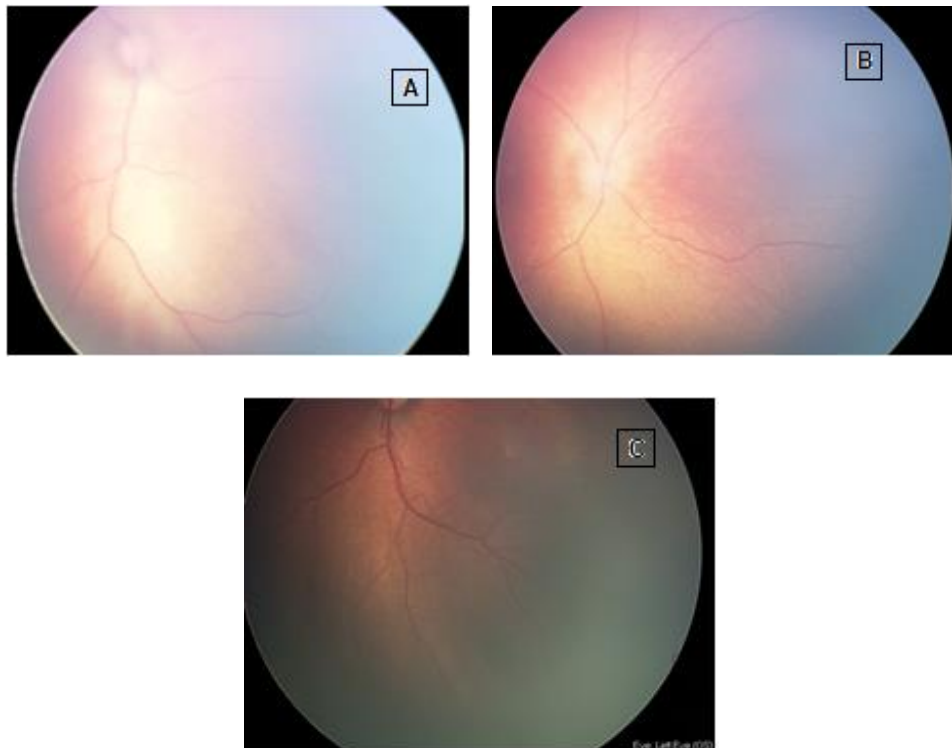


Figure 34, Image examples of the Low quality (A), Medium quality (B), and High quality (C) pools

Chapter 4: Results

4.1. Data Augmentation by Patches

The result of this process (Figure 35, Image and Ground Truth Patches Databases Creatio) is two databases of patches representing the retina by image and segmentation, following the same order, in order to properly feed a training, and cleaned by all the pixels representing the background that are not providing any useful information to the process.

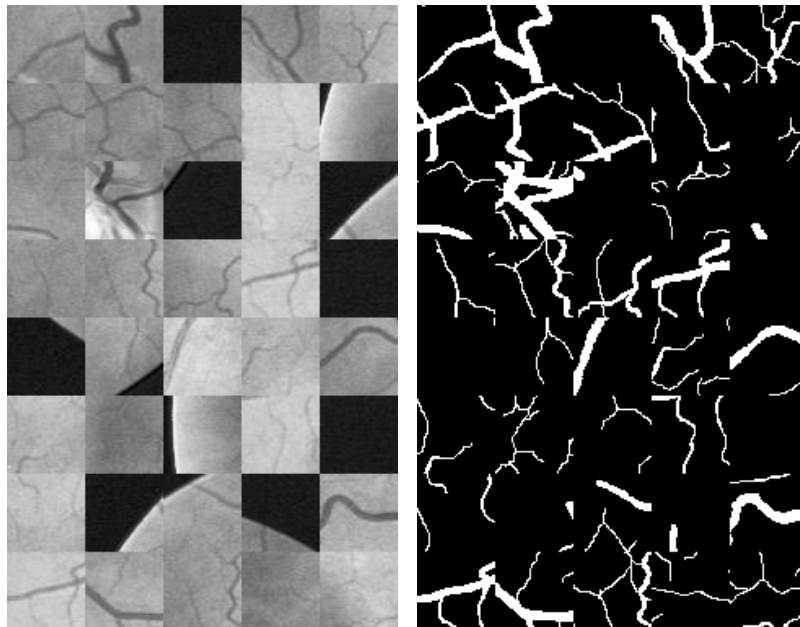


Figure 35, Image and Ground Truth Patches Databases Creation

4.2. Pre-processing

The result of this stage is a new database of images in grayscale with enhanced colour and contrast that simplifies the following step of segmentation.

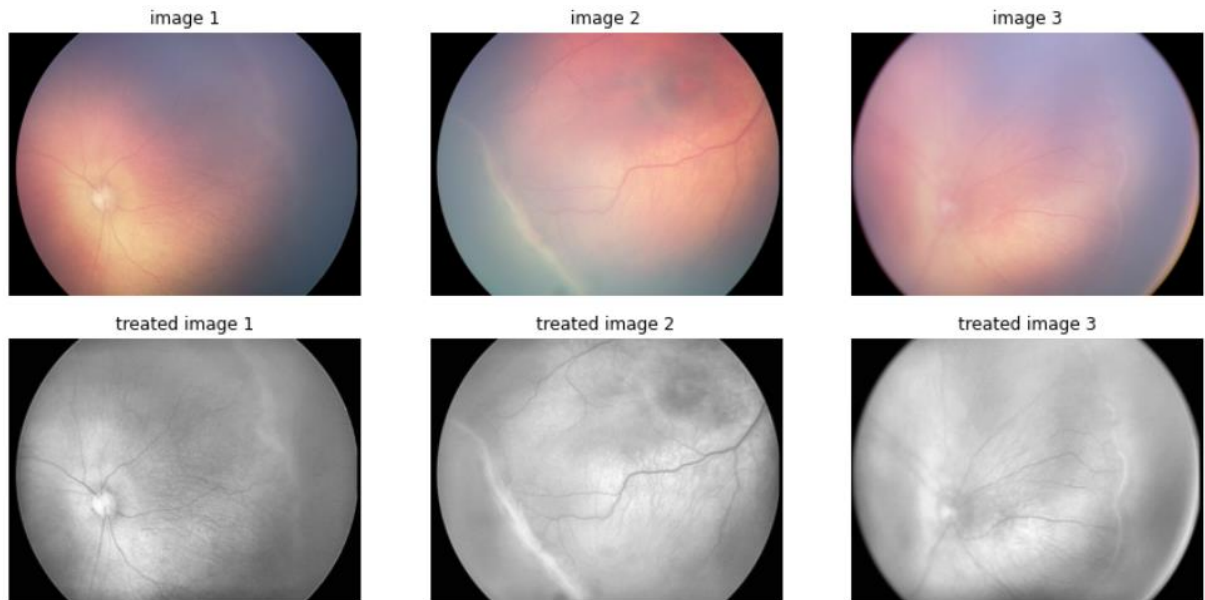


Figure 36, Raw and Pre-processed Images

4.3. U-NET (Segmentation)

The result obtained by U-Net are the segmentations of the retina images; indeed, a black and white image discerning only between vein and arteries of the retina and the left pixels of the image. During the training and test phase made on the Drive set (**Error! Reference source not found.**), the quality of the segmentation is underlined by the cleaning of the picture in which background and feature are not related to the vein and arteries are not affecting the result. Indeed, changes in color and shades, and borders are not represented in the segmentation of the image.

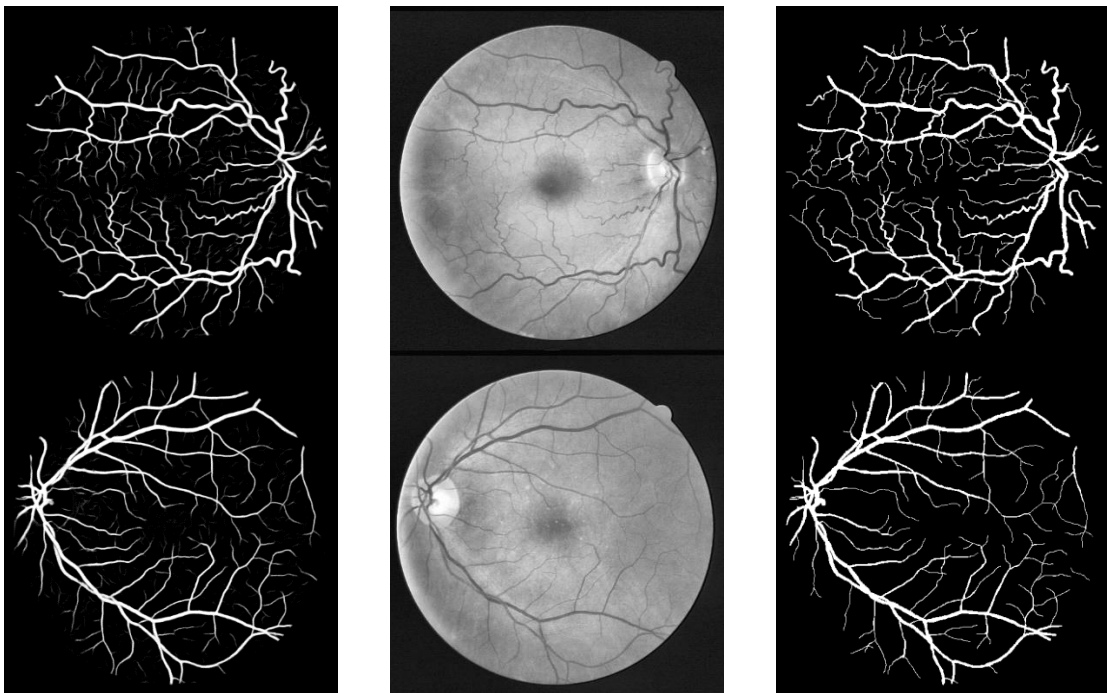


Figure 37, U-Net Segmentation

In Table 3, U-Net evaluation, the result has been reported with the main parameters underlining the quality of not only the correct classification of the segment's pixels (True Positive), essential for detection of the arteries; but also the low level of misclassification of the pixels not belonging to the structure (False Positive) is achieving adequate quality.

Evaluator	Score
Area under ROC curve	0.9837
Jaccard similarity score	0.9615
F-measure score (F1)	0.8315
Accuracy	0.9615
Sensitivity	0.7902
Specificity	0.9849
Precision	0.8772

Table 3, U-Net evaluation

The actual result of new images is more affected by quality and noises in the random images captured by the red cam analysis; indeed, the level of False Positive is increased as shown in

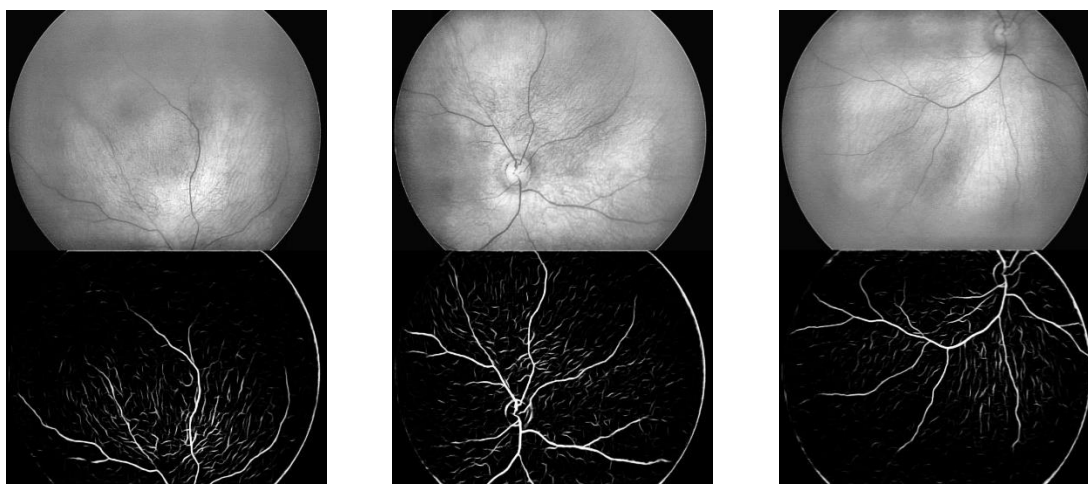


Figure 38, segmentation of retina image provided by the SJD hospital

4.4. Cleaning

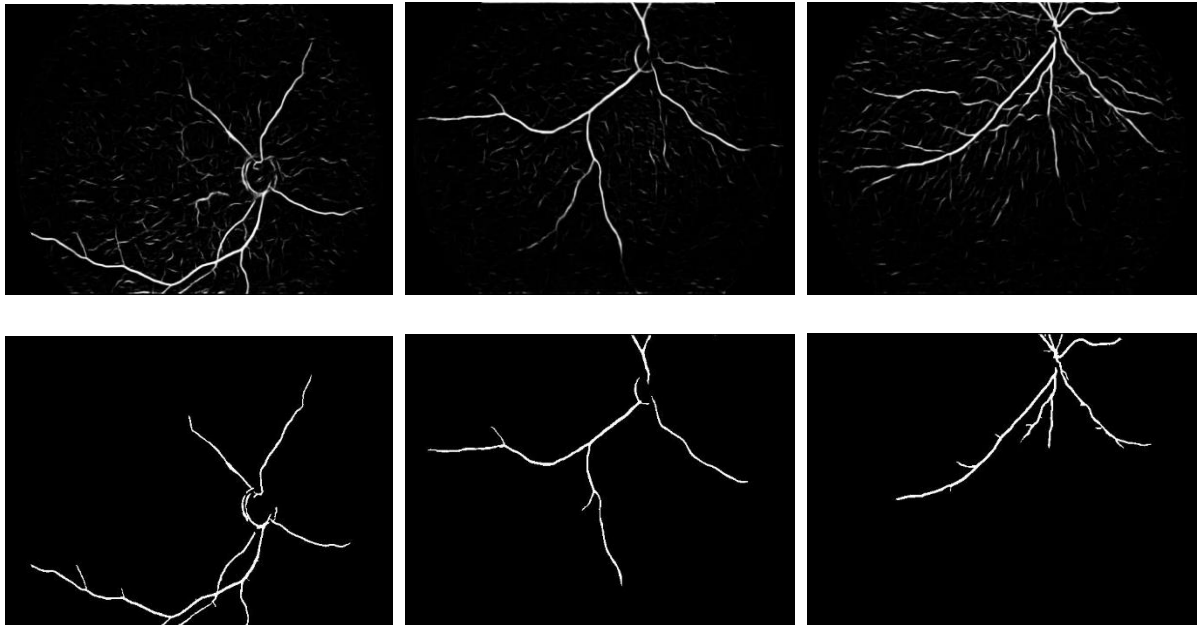


Figure 39, Segmentation Cleaning

The cleaning of the image obtains a clear focus on the pixels targeted, getting rid of the false positive due to noise, incomplete or incorrect segmentation, and incorrect reframing (Figure 39, Segmentation Cleaning); this last correction is properly solving problems related to both image frame and

- Image frame

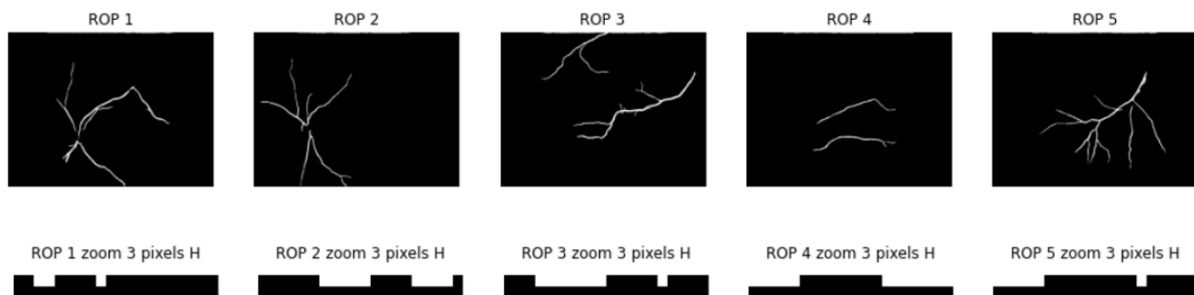


Figure 40, Zoom of the top border of segmentation frames

- the FOV border frame

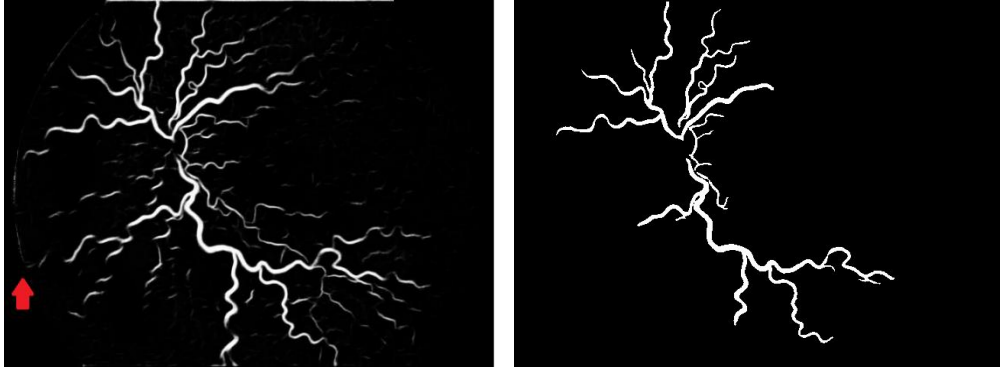


Figure 41, Field of View Border correction

4.5. Skeletonization

The result of this process (Figure 42, Process of skeletonization) is a pixel-wise structure describing branch by branch how the retina is developing from the macula till the end of the ramification. The structure is able to define the start and endpoint of each branch (Table 4, Skeleton Description with number of branches, type (single or multi-connected), start point, endpoint) in order to localize and properly analyze them.

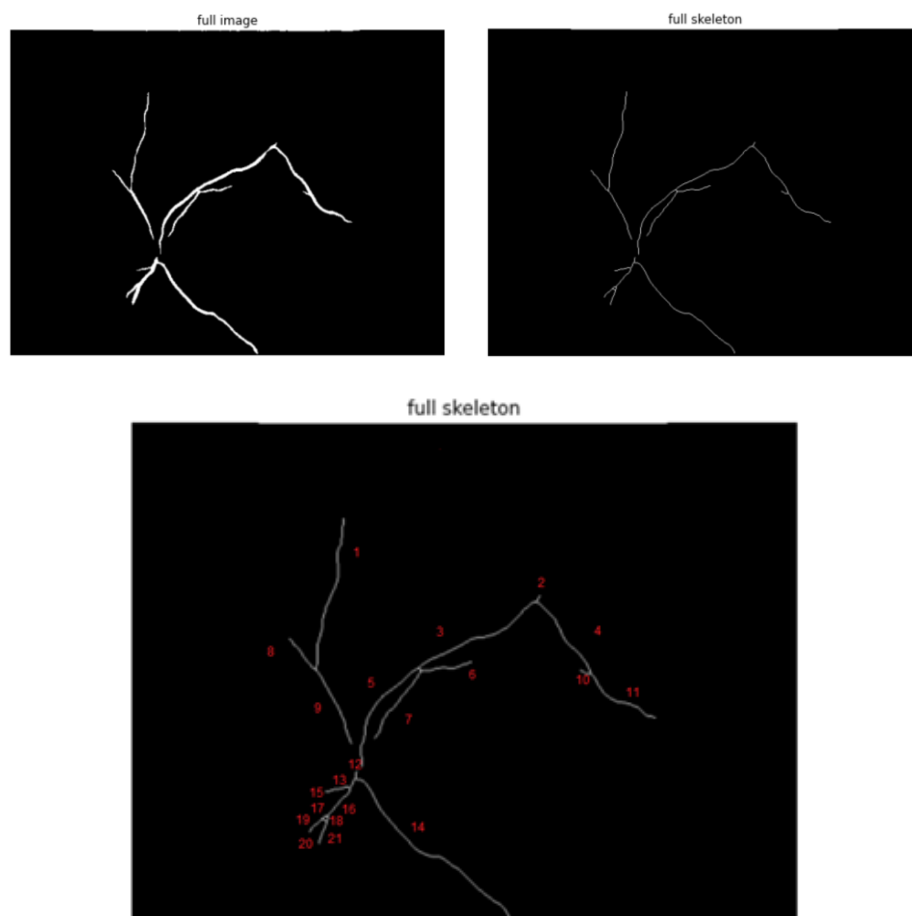


Figure 42, Process of skeletonization

Nº	Type	X1	Y1	X2	Y2
1	1	92	203	237	176.75
2	1	166	392	172	389
3	2	172	389	237	277
4	2	172	389	241	441
5	1	237	277	330	221
6	1	237	277	230	326
7	1	237	277	303	233
8	1	237	177	208	151
9	1	237	177	308	211
10	1	241	441	238	431
11	1	241	441	283	503
12	1	342	215	335	216
13	2	342	215	351	209
14	1	342	215	475	362
15	1	351	209	354	186
16	2	351	209	378	188
17	2	378	188	381	183
18	2	378	188	382	187
19	1	381	183	393	171
20	2	381	183	382	187
21	1	382	187	404	180

Table 4, Skeleton Description with number of branches, type (single or multi-connected), start point, endpoint

4.6. Standard Pool Distribution

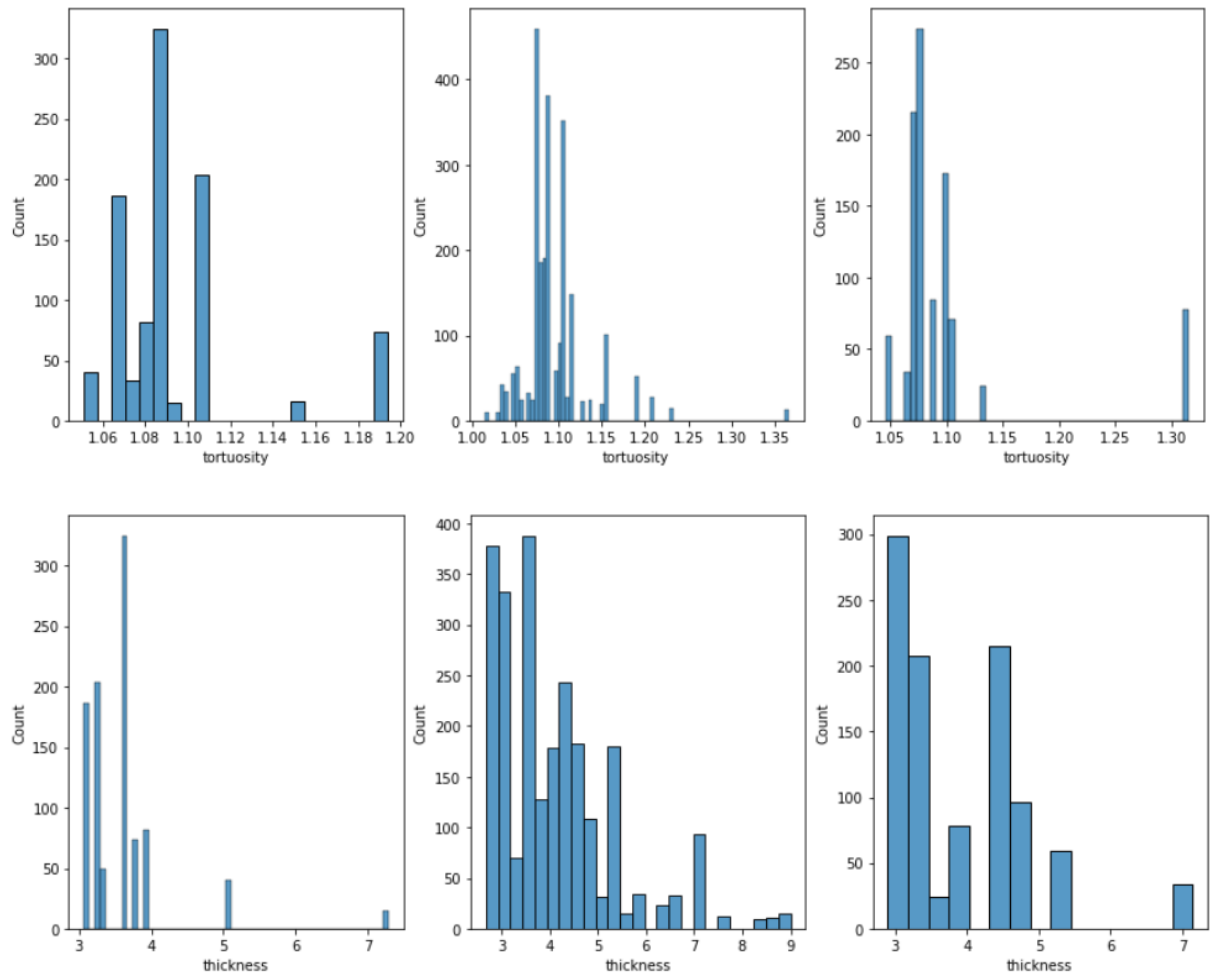


Figure 43, Examples of Thickness and Tortuosity branches distribution for three retina images

For each retina image of the pool, the tortuosity and thickness are calculated (Figure 43, Examples of Thickness and Tortuosity branches distribution for three retina images) and averaged (Table 5, Standard Pool Mean Tortuosities and Thicknesses) in order to analyze only their mean values compared between ROP plus and no plus (Figure 44, Standard Pool distribution of ROP and NO ROP Mean Values).

NO PLUS		PLUS	
Mean Tortuosity	Mean Thickness	Mean Tortuosity	Mean Thickness
1.088807	3.796529	1.143823	4.581064
1.092248	4.168963	1.148229	4.908451
1.180694	4.943092	1.120048	5.365388
1.1132	4.763832	1.111923	5.105639
1.117021	4.530974	1.102446	4.878457
1.102112	4.200837	1.130479	5.553317
1.105766	5.023197	1.126934	4.468652
1.107312	4.536827	1.152833	4.445546
1.108953	4.331468	1.113259	4.598814
1.088948	4.544616	1.11106	4.751981
1.145389	4.488362	1.112427	5.241904
1.089245	4.211988	1.113666	5.019616
1.085585	3.941188	1.107489	4.427118
1.096915	4.0685	1.252589	5.309727
1.091725	4.381738	1.154194	4.638733
1.090453	4.502076	1.135137	4.705603
1.091336	4.483907	1.089068	4.929474
1.084214	3.781686	1.142503	4.368733
1.104265	4.502997	1.256121	5.643639
1.137464	3.936122	1.122382	6.238625

Table 5, Standard Pool Mean Tortuosities and Thicknesses

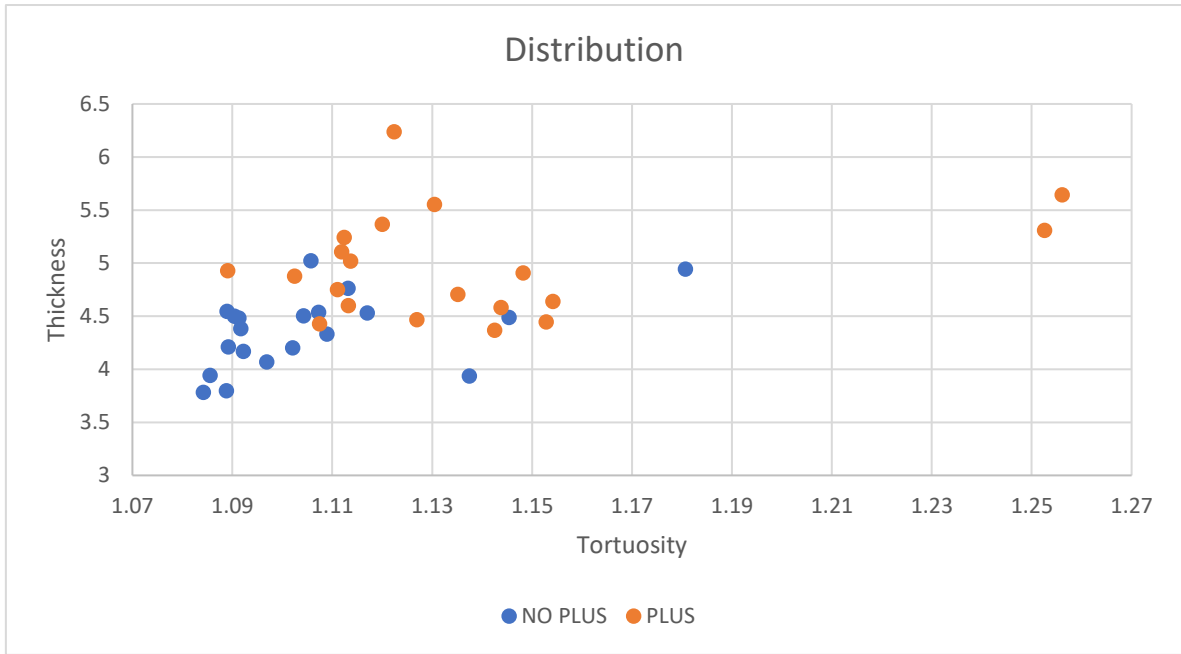


Figure 44, Standard Pool distribution of ROP and NO ROP Mean Values

4.7. Double Threshold

Defined the minimum and maximum of the two classes of the pool in term of thickness and tortuosity:

	NO PLUS		PLUS	
	Tortuosity	Thickness	Tortuosity	Thickness
Min	1.084214	3.781686	1.089068	4.368733
Max	1.180694	5.023197	1.256121	6.238625

Table 6, Min and Max Values of Tortuosity and Thickness for the ROP Plus and No Plus of the Standard Pool

Those are used in order to spot which images are both falling, with regards to thickness and tortuosity, in the same classification by the double threshold classifier are substrated the reference values from the mean values (Table 7, Double Threshold Results); and thus, following the restrictive classification (Table 8, Double Threshold Scor).

NO PLUS				PLUS			
No Plus Threshold		Plus Threshold		Plus Threshold		No Plus Threshold	
Tort	Thick	Tort	Thick	Tort	Thick	Tort	Thick
0.0002605	0.572204	-0.091887	-1.22667	-0.036871	-0.442133	-0.054755	-0.21233
-0.00318	0.19977	-0.088446	-0.85423	-0.032465	-0.114746	-0.059161	-0.53972
-0.091626	-0.57436	0	-0.08011	-0.060646	0.3421909	-0.03098	-0.99665
-0.024132	-0.3951	-0.067494	-0.25936	-0.068771	0.0824417	-0.022855	-0.73691
-0.027954	-0.16224	-0.063673	-0.49222	-0.078248	-0.14474	-0.013378	-0.50972
-0.013044	0.167896	-0.078582	-0.82236	-0.050214	0.5301204	-0.041412	-1.18458
-0.016698	-0.65446	-0.074928	0	-0.05376	-0.554545	-0.037866	-0.09992
-0.018245	-0.16809	-0.073381	-0.48637	-0.027861	-0.577651	-0.063765	-0.07681
-0.019885	0.037266	-0.071741	-0.69173	-0.067435	-0.424383	-0.024191	-0.23008
0.0001197	-0.17588	-0.091746	-0.47858	-0.069633	-0.271216	-0.021993	-0.38325
-0.056321	-0.11963	-0.035305	-0.53484	-0.068267	0.2187066	-0.023359	-0.87317
-0.000177	0.156746	-0.091449	-0.81121	-0.067028	-0.003581	-0.024598	-0.65088
0.0034828	0.427546	-0.095109	-1.08201	-0.073205	-0.596079	-0.018421	-0.05838
-0.007847	0.300233	-0.083779	-0.9547	0.0718949	0.2865299	-0.163521	-0.94099
-0.002657	-0.013	-0.088969	-0.64146	-0.0265	-0.384464	-0.065126	-0.27
-0.001385	-0.13334	-0.090241	-0.52112	-0.045557	-0.317594	-0.046069	-0.33687
-0.002268	-0.11517	-0.089358	-0.53929	-0.091626	-0.093723	0	-0.56074
0.0048542	0.587047	-0.09648	-1.24151	-0.038191	-0.654464	-0.053435	0
-0.015197	-0.13426	-0.076429	-0.5202	0.0754273	0.6204422	-0.167053	-1.27491
-0.048397	0.432612	-0.043229	-1.08708	-0.058312	1.2154285	-0.033314	-1.86989

Table 7, Double Threshold Results

	Number	Percentage [%]
Correct	5	12.5
Not Detected	35	87.5
Error	0	0

Table 8, Double Threshold Score

4.8. Mean Distance Threshold

Defined the mean of the two classes of the pool in terms of thickness and tortuosity:

	Tortuosity		Thickness	
	NO PLUS	PLUS	NO PLUS	PLUS
Average	1.106083	1.13733	4.356945	4.959024
Mean Dist.	1.121707		4.657984	

Table 9, Mean Values of Tortuosity and Thickness for the ROP Plus and No Plus of the Standard Pool

Those are used in order to classify every parameter of every image (Table 10, Mean Distance Threshold Results), producing a score less affected by lack of detection but with a higher risk of misdetection (Table 11, Mean Distance Threshold Score).

NO PLUS		PLUS	
Tortuosity	Thickness	Tortuosity	Thickness
0.032899	0.861455	0.022117	-0.07692
0.029458	0.489021	0.026522	0.250466
-0.05899	-0.28511	-0.00166	0.707403
0.008507	-0.10585	-0.00978	0.447654
0.004685	0.12701	-0.01926	0.220473
0.019595	0.457147	0.008773	0.895333
0.015941	-0.36521	0.005227	-0.18933
0.014394	0.121157	0.031126	-0.21244
0.012753	0.326517	-0.00845	-0.05917
0.032758	0.113369	-0.01065	0.093997
-0.02368	0.169623	-0.00928	0.583919
0.032462	0.445997	-0.00804	0.361631
0.036122	0.716797	-0.01422	-0.23087
0.024791	0.589484	0.130882	0.651742
0.029981	0.276246	0.032487	-0.01925
0.031254	0.155909	0.01343	0.047618
0.030371	0.174078	-0.03264	0.27149
0.037493	0.876298	0.020796	-0.28925
0.017441	0.154987	0.134415	0.985655
-0.01576	0.721863	0.000675	1.580641

Table 10, Mean Distance Threshold Results

	Number	Percentage [%]
Correct	21	52.5
Not Detected	14	35
Error	3	7.5

Table 11, Mean Distance Threshold Score

4.9. Smart Threshold

Defined the two parameters that best classifies the two clusters:

	Tortuosity	Thickness
Value	1.11	4.55

Table 12, Smart Values of Tortuosity and Thickness for the ROP Plus and No Plus of the Standard Pool

In the same way, those are used in order to classify every parameter of every image (Table 13, Smart Threshold Results), producing a higher score (Table 14, Smart Threshold Score Table 11, Mean Distance Threshold Score) but risking higher overfitting around the standard pool data.

NO PLUS		PLUS	
Tortuosity	Thickness	Tortuosity	Thickness
0.021193	0.753471	0.033823	0.031064
0.017752	0.381037	0.038229	0.358451
-0.07069	-0.39309	0.010048	0.815388
-0.0032	-0.21383	0.001923	0.555639
-0.00702	0.019026	-0.00755	0.328457
0.007888	0.349163	0.020479	1.003317
0.004234	-0.4732	0.016934	-0.08135
0.002688	0.013173	0.042833	-0.10445
0.001047	0.218532	0.003259	0.048814
0.021052	0.005384	0.00106	0.201981
-0.03539	0.061638	0.002427	0.691904
0.020755	0.338012	0.003666	0.469616
0.024415	0.608812	-0.00251	-0.12288
0.013085	0.4815	0.142589	0.759727
0.018275	0.168262	0.044194	0.088733
0.019547	0.047924	0.025137	0.155603
0.018664	0.066093	-0.02093	0.379474
0.025786	0.768314	0.032503	-0.18127
0.005735	0.047003	0.146121	1.093639
-0.02746	0.613878	0.012382	1.688625

Table 13, Smart Threshold Results

	Number	Percentage [%]
Correct	28	70
Not Detected	8	20
Error	3	7.5

Table 14, Smart Threshold Score

4.10. % Threshold

Starting from the results obtained on the Mean Distance Threshold, the further classification, with regards to the “not detected” images, is performed in order to obtain a comprehensive classification (Table 15% Threshold Results) yet at the expense of reliability (Table 16, % Threshold score).

NO PLUS			PLUS		
Tortuosity	Thickness	% Tot	Tortuosity	Thickness	% Tot
19.13776	35.06214	54.1999	12.86537	-3.13076	9.734609
17.13611	19.90368	37.03979	15.42819	10.19423	25.62242
-34.3134	-11.6042	-45.9176	-0.9647	28.79206	27.82736
4.948416	-4.30812	0.640299	-5.69144	18.22	12.52856
2.725423	5.169445	7.894868	-11.2038	8.973481	-2.23035
11.39849	18.60638	30.00487	5.103251	36.44099	41.54424
9.272893	-14.8645	-5.59164	3.040847	-7.70604	-4.6652
8.373204	4.931223	13.30443	18.10643	-8.64645	9.459979
7.41878	13.28957	20.70835	-4.91388	-2.4083	-7.32218
19.05583	4.614228	23.67005	-6.19293	3.825757	-2.36718
-13.7764	6.903818	-6.87256	-5.39798	23.76612	18.36814
18.88341	18.15254	37.03595	-4.6775	14.71878	10.04128
21.01217	29.17437	50.18654	-8.27062	-9.39653	-17.6671
14.4213	23.99262	38.41393	76.13524	26.5266	102.6618
17.44033	11.24352	28.68385	18.89791	-0.78357	18.11434
18.18045	6.345649	24.5261	7.81259	1.938109	9.750699
17.6669	7.085139	24.75204	-18.9862	11.04992	-7.93629
21.80991	35.66625	57.47617	12.09733	-11.7728	0.324509
10.14579	6.308137	16.45393	78.19009	40.11718	118.3073
-9.16648	29.38058	20.21409	0.39275	64.33375	64.7265

Table 15% Threshold Results

	Number	Percentage [%]
Correct	31	77.5
Not Detected	0	0
Error	9	22.5

Table 16, % Threshold score

4.11. Decision Tree

This classifier is producing a more complex system of thresholds linked to the whole set of values of the standard pool.

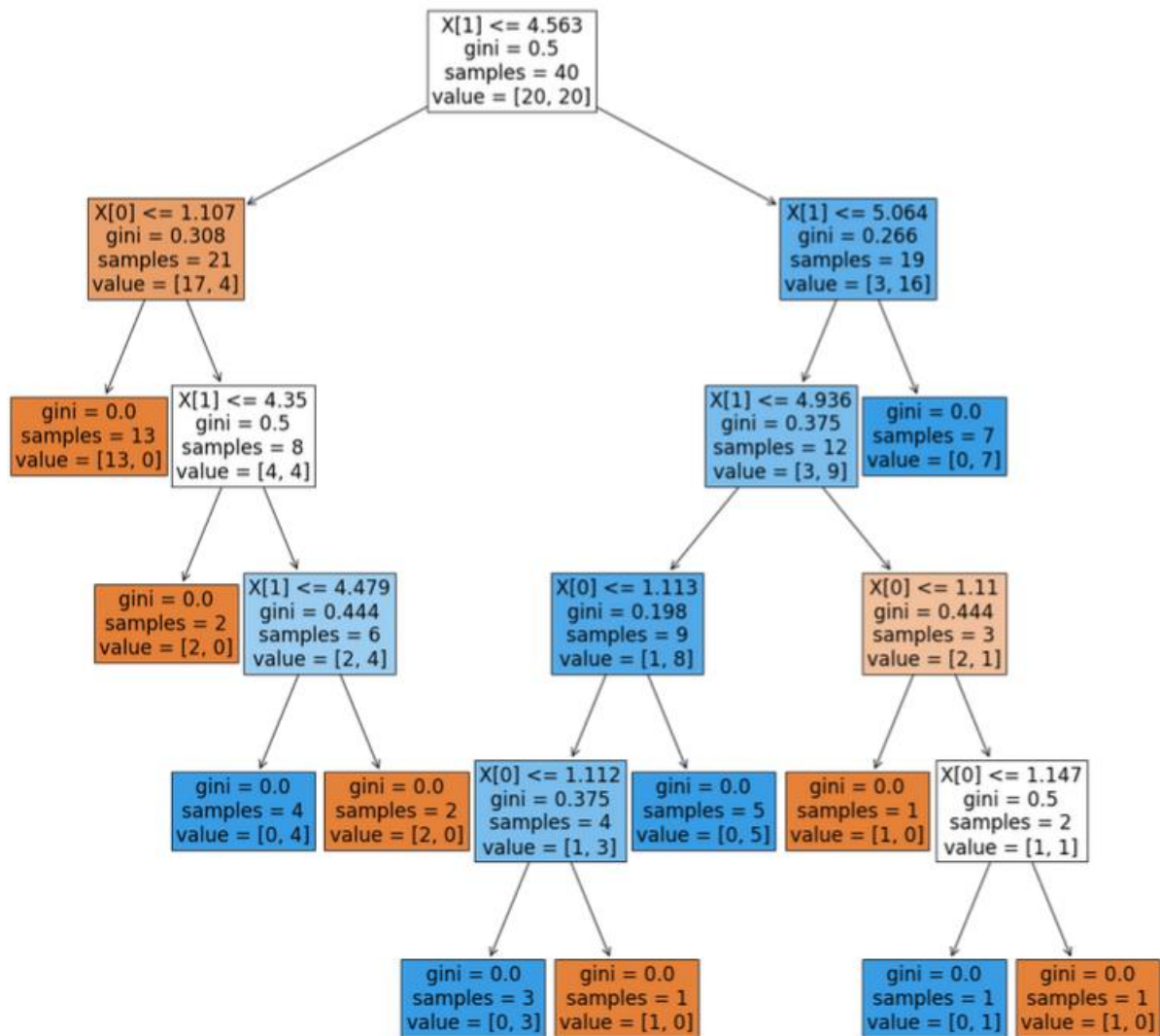


Figure 45, Decision Tree Classifier Flowchart

4.12. Different Pools Evaluation by the Threshold Proposed

The four classifiers using a threshold are tested with the six Pool of data and return the following result in terms of number correctly spotted, mistakes, and not identified.

		Standard Pool		Low Quality		Mean Quality		High Quality		Random Pool 1		Random Pool 2		Random Pool 3	
		Nº	%	Nº	%	Nº	%	Nº	%	Nº	%	Nº	%	Nº	%
Double Threshold	Corr	5	12.5	3	15	2	10	3	15	3	15	2	10	2	10
	ND	35	87.5	17	85	18	90	17	85	17	85	18	90	18	90
	Err	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Mean Threshold	Corr	21	52.5	10	50	10	50	10	50	13	65	11	55	9	45
	ND	14	35	8	40	8	40	7	35	5	25	4	20	7	35
	Err	3	7.5	2	10	2	10	3	15	2	10	5	25	4	20
Smart Threshold	Corr	28	70	8	40	6	30	9	45	15	75	10	50	8	40
	ND	8	20	11	55	12	60	10	50	3	15	8	40	9	45
	Err	3	7.5	1	5	2	10	1	5	2	10	2	10	3	15
% Threshold	Corr	31	77.5	15	75	11	55	15	75	16	80	12	60	13	65
	ND	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Err	9	22.5	5	25	9	45	5	25	4	20	8	40	7	35

Table 17, Result of classification in terms of the different classifier using simple threshold applied on the six Pool analyzed

	Average Score
Double Threshold	1
Mean Threshold	0.908
Smart Threshold	0.85
% Threshold	0.683

Table 18, Average Score Classifiers

The results can be read in order to see the different ratio Correct Match – Reliability of the classifier, in which the correct match is the correct classification of the Plus or No Plus status and the Reliability considers only the cases of mistake as negative, not assuming as a mistake the impossibility of correct classification. Those are evaluated for each classifier:

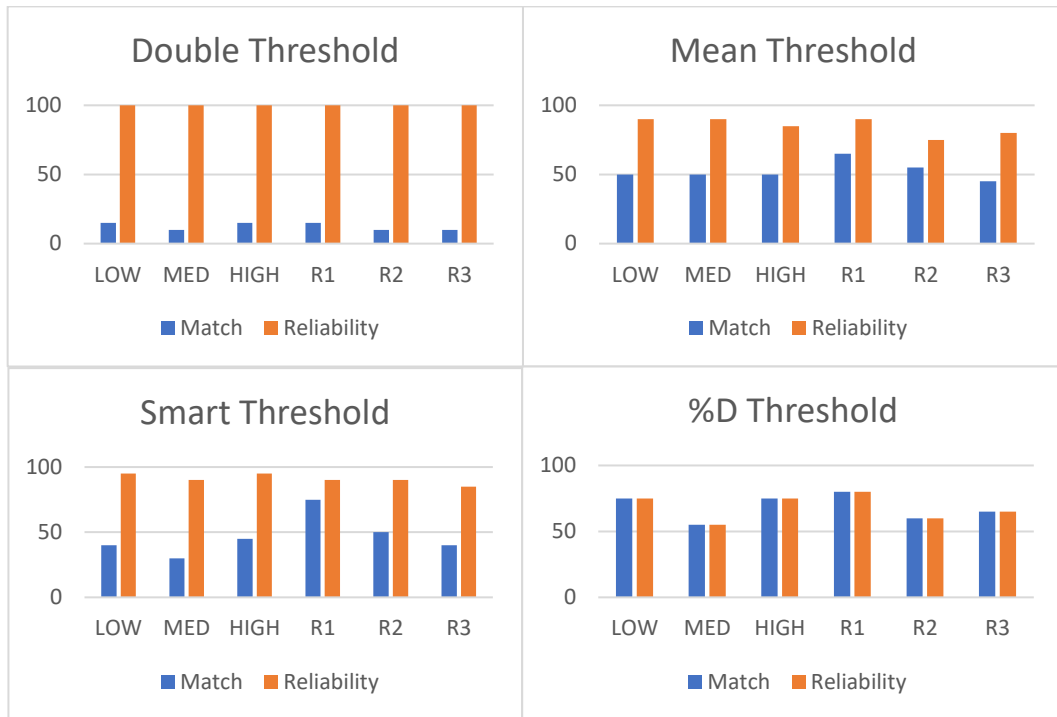


Figure 46, Double Threshold, Mean Threshold, Smart Threshold, and % Threshold evaluation by the Match-Reliability Ratio

Finally, an overall evaluation of the results, taking into account all the data analyzed, is performed:

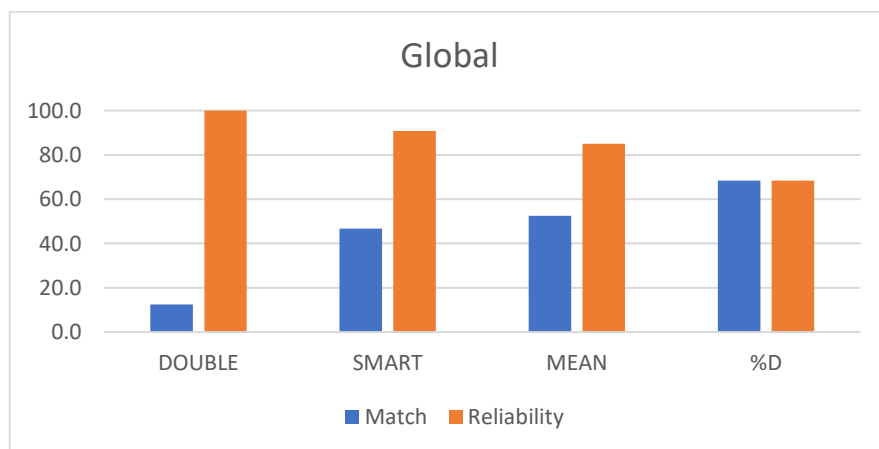


Figure 47, Global Evaluation by the Match - Reliability Ratio

Decision Tree Results

	Accuracy
Low Quality	0.65
Mean Quality	0.6
High Quality	0.7
Random Pool 1	0.9
Random Pool 2	0.6
Random Pool 3	0.8
Average	0.7083

Chapter 5: Discussion of the results

The first result reported (Figure 35, Image and Ground Truth Patches Databases Creatio) are patches obtained; the importance of those is the possibility of passing from a large and complex image to a set of more simple smaller images with a twofold gaining; not only the images interpreted by the network are simpler and thus the computational time and power required are lower, but also the sets of training, validation and test images are augmented enhancing the chances of a correct interpretation without underfitting; furthermore, this result allows to target only specific regions of the image, getting rid of a group of pixels that are lacking sensible information, such as the background.

The pre-processing (Figure 36, Raw and Pre-processed Images) allows to obtain a new set of images with a more focused range of information that helps the network to concentrate only on the relevant information and also to enhance them to simplify its learning process; therefore, the resultant images are in a grey-scale with a higher contrast emphasizing the vessels in their borders and details.

The U-Net results (Figure 37, U-Net Segmentation) shows that the particular structure of this architecture with the convolutional and decompression stages is able to properly reproduce the structure with a notable accuracy (Table 3, U-Net evaluation), compared to the previously reported studies (Table 1, AUC ROC on DRIVE Database with the different methods proposed in the state of the art). The stark difference between the segmentation of a clean and refined image from a public dataset (Figure 37, U-Net Segmentation) and a segmentation of a new image obtained from the hospital (Figure 38, segmentation of retina image provided by the SJD hospital), underlines the heterogeneity of the possible results strongly affected by several factors such as noise, quality of the image, RedCam machine used.

The Cleaning stage (Figure 39, Segmentation Cleaning) obtains a new set of images that are more likely to be properly described by the feature extractors due to the absence of noise due to the original image recording quality, but also possible noise produced in the process of segmentation such as the border of the FOV (Figure 41, Field of View Border correction) or the border of the image (Figure 40, Zoom of the top border of segmentation frames). Overall, the stage simplifies the process of feature extraction, getting rid of the noise at the expense of losing smaller branches.

The skeletonization process (Figure 42, Process of skeletonization) organize the image by a pixel-wise structure splitting each branch and categorizing them (Table 4, Skeleton Description with number of

branches, type (single or multi-connected), start point, endpoint) by the intersection point that is linking them; this structure allows not only to work singularly on the particular branch but also to follow the structure passing through different intersection point and thus obtaining the characteristics of agglomerate of different arteries and veins; furthermore, the reduction of the segmentation thickness to zero, allows to easily calculate directional parameters and to compare the structure with the original obtaining information about the thickness either.

The standard pool of images is passed in the process of segmentation and features extraction obtaining a set of the different distribution of thickness and tortuosity for each image (Figure 43, Examples of Thickness and Tortuosity branches distribution for three retina images); those are averaged defining the mean characteristic of the images (Table 5, Standard Pool Mean Tortuosities and Thicknesses) in order to be easily evaluated looking for a pattern between Plus and No Plus (Figure 44, Standard Pool distribution of ROP and NO ROP Mean Values).

The standard creation allows to give, by different classifiers, a more complex and comprehensive classification of the illness in terms of reliability; indeed, the method is able to classify with different levels of confidence an increasingly higher percentage of images. The first and more restrictive method, Double Threshold, allows the classification of a narrow percentage of images with almost perfect accuracy (Table 8, Double Threshold Scor); on the other hand, the % Threshold Classifies the whole range of images with a lower level of accuracy (Table 16, % Threshold score); finally, the two other classifiers allows to have an intermediate answer with discrete accuracy in spotting midway percentage of images (Table 11, Mean Distance Threshold Score; Table 14, Smart Threshold Score). The standard creation is strongly influenced by the particular pool of data used for its definition; therefore, different pools could result in different thresholds and match-reliability ratios; indeed, the farther definition of classifier relying on a greater and, thus, safer, are necessary to avoid any possible overfitting or underfitting and enhance the robustness of the model.

Moreover, a more complex classifier, Decision Tree, is used in order to fit in a better and more specific way around the selected standard pool; this classification allows to correctly classify the whole pool of images splitting the two range of values of the parameters in smaller threshold following a decisions line properly customized.

The first notable characteristics underlined by the table of results (Table 17, Result of classification in terms of the different classifier using simple threshold applied on the six Pool analyzed) are:

- The high level of reliability of the Double Threshold, shown by the error field equal to zero for all the pools of data; and the comprehensive classification of the % Threshold, underlined, also in this case, by the line of zeros referring to the non identified.
- The intermediate situation of the two other simple threshold classifiers providing useful additional information without costing much in terms of decrease of reliability.
- A comparable behavior of the classifiers, when tested with the different pool of data, was proposed, with higher stability for the Double Threshold and more fluctuation for the % Threshold and the Random 1 pool for the Smart Threshold; probably more affected since they are customized to fit better the training pool and thus, could have lower results for particular different pool.

It is possible to see how there is no best classifier but just a trade-off between classifying with a high level of accuracy many images and classify many images with lower accuracy (Figure 47, Global Evaluation by the Match - Reliability Ratio)

Finally, the decision tree results show how this more complex classifier can fit overall better than the other analyzed; furthermore, apart from the % Threshold with comparable results, it shows better results in each pool classification proposed.

Chapter 6: Conclusions and Future Works

The aim of this project was to create a first tool able to discern between ROP and ROP Plus cases in order to evaluate the possible room for improvement for a further and more comprehensive project developed in an enlarged frame of time and resources. In this particular perspective, the results

obtained underline a positive conclusion of this first stage, the classification of the retina images pass through two steps, a first segmentation of the image in order to obtain a data format easily analyzable and clean, and a second able to extrapolate the features necessary for the illness assessment; the first result was achieved with an accuracy of 0.9615 and the second one with a greater score of 0.7083.

Nevertheless, higher results are feasible for both the simple threshold classifiers, which are closer to the definition of a possible threshold between the different categories as theorized by doctors in order to spot the illness and complex classifiers; indeed, increasing the training pool of data and the complexity of the classifiers is possible to enhance also the quality of the result.

It is important to remark the main accomplishment achieved during the project:

- Segmentation process through a U-Net architecture obtaining adequate result in order to perform proficient analysis in the following stages.
- Data augmented application, by patches, aimed to overcome the high complexity in the structure segmented and the low dimension of the available datasets.
- Pre-treatment of the images by a range of “cv2” filters in order to level the quality of heterogeneous sources and to facilitate the network focus in the targeted structures.
- Generalize code and function in order to be able to process any image in terms of format, quality, and dimension, adding a border mask stage and automatizing the process.
- Select and standardize the features necessary in order to evaluate the Retinopathy of Prematurity accordingly to the guidelines followed by doctors and obtaining an algorithm able to mimic the human process.
- The realization of a process of classification between ROP Plus and no Plus.

This process is just the core used in order to obtain a starter point for a project aiming to achieve further results; indeed, not only it is meant to be refined and implemented in order to increase the quality, in terms of accuracy overall, of the two stages explained, but also to be implemented in a wider pipeline of analysis of the images.

Some of the implementations expected are:

- Feature Extractions refinement in terms of Feature and quality of extraction.
- Pre-process of ROP versus No ROP discrimination in order to evaluate the presence of the illness before its gravity level.

- The highlight of the areas of the images with the most sensitive information for the classification.
- Development of a Neural Network able to discern between the different classes without passing through the feature but with a black box logic in order to compare the results obtained.
- Introduction of the pre Plus stage.
- Creation of a dash application able to overcome the gap of knowledge between doctors and engineers, merging all the results described in an easily usable tool.

Chapter 7: References

Alajbegovic-Holimic, J., Zvizdic, D., Alimanovic-Halilovic, E., Dodik, I., & Duvnjak, S. (2015). Risk Factors for Retinopathy of Prematurity in Premature Born Children. *Medical Archives*, vol. 69, nº 6, 409-413.

- Asole, V. (2021, 09 29). *GitHub Repository for "Automated Classification of Retinopathy of Prematurity in Newborns"*. Retrieved from GitHub: https://github.com/asole91/TFM_RetinopathyPlusDetection
- Azzopardi, G., Nicola, S., Mario, V., & Nicolai, P. (2015). Trainable COSFIRE filters for vessel delineation with application to retinal images. *Medical Image Analysis*, vol. 19, 46-57.
- Ben-Sira, I., Deutman, A., Fledelius, H., Flynn, J., Garner, A., Gole, G., . . . Tanaka, Y. (1984). An International Classification of Retinopathy of Prematurity. *Archives of Ophthalmology*, vol. 102, 1130-1134.
- Bisong, E. (2019). *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Berkeley, CA: Apress.
- Blondel, M., Brucher, M., Buitinck, L., Cornapeau, D., Dawe, N., Du, S., . . . Weiss, R. (2011). Scikit-Learn: Machine Learning in Python. *Journal of Machine Learning Research*, vol. 12, 2825-2830.
- Bradski, G. (2000). The OpenCV Library. *Dr. Dobbs's Journal of Software Tools*, vol. 120, 122-125.
- Brown, J. M., Campbell, P. J., Beers, A., Chang, K., Ostmo, S., Chan, P. R., . . . Chiang, M. F. (2018). Automated Diagnosis of Plus Disease in Retinopathy of Prematurity Using Deep Convolutional Networks. *JAMA Ophthalmology*, vol. 136, nº 7, 803-810.
- Chandrasekaran, B. (1983). Evaluating Artificial Intelligence System for Medical Diagnosis. *AI Magazine*, 34-37.
- Chen, L., & Wong, G. (2019). Transcriptome Informatics. *Encyclopedia of Bioinformatics and Computational Biology*, 324-340.
- Chiang, M. F., Quinn, G. E., Fielder, A. R., Ostmo, S. R., Chan, P. R., Berrocal, A., . . . Zin, A. (2021 (article in press)). International Classification of Retinopathy of Prematurity, Third Edition. *Arch Ophthalmol*.
- Chollet, F., & others. (2015). *Keras*. Retrieved from GitHub: <https://github.com/fchollet/keras>
- Colette, A. (2013). *Python and HDF5*. O'Reilly.
- Da Silva, E. A., & Mendonca, G. V. (2005). Digital Image Processing. In E. A. Da Silva, & G. V. Mendonca, *The Electrical Engineering Handbook* (pp. 891-910). Burlington: Academic Press.

- Data Protection Act 2018. (2018). Retrieved from GOV.UK: <<https://www.gov.uk/government/collections/data-protection-act-2018>>
- Fraz, M. M., Remagnino, P., Hoppe, A., Uyyanonvara, B., Rudnicka, A. R., Owen, C. G., & Barman, S. A. (2012, September 9). An Ensemble Classification-Based Approach Applied to Retinal Vessel Segmentation. *IEEE Transactions on Biomedical Engineering*, vol. 59, 2538-2548.
- Fraz, M. M., Remagnino, P., Hoppe, A., Uyyanonvara, B., Rudnicka, A. R., Owen, C. G., & Berman, S. A. (2012). Blood Vessel Segmentation Methodologies in Retinal Images. *Computer Methods and Programs in Biomedicine*, vol. 108, 407-433.
- Gelman, R., Jiang, L., Du, Y. E., Martinez-Perez, E. M., Flynn, J. T., & Chiang, M. F. (532-540). Plus Disease in Retinopathy of Prematurity: Pilot Study of Computer-Based and Expert Diagnosis. *Journal of AAPOS : the official publication of the American Association for Pediatric Ophthalmology and Strabismus*, vol. 11, nº 6, 2007.
- Gelman, R., Martinez-Perez, E. M., Vanderveen, D. K., Moskowitz, A., & Fulton, A. B. (2005). Diagnosis of Plus Disease in Retinopathy of Prematurity Using Retinal Image multiScale Analysis. *Investigative Ophthalmology & Visual Science*, vol. 46, nº 12, 4734-4738.
- Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep Sparse Rectifier Networks. *International Conference on Artificial Intelligence and Statistics*, 315-323.
- Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep Sparse Rectifier Neural Networks. *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS 2011)* (pp. 315-323). PMLR.
- Gole, G. A., Ells, A. L., Katz, X., Holmstrom, G., Fielder, A. R., Capone, A., . . . Wallace, D. K. (2005). The International Committee for the Classification of Retinopathy of Prematurity revisited. *Arch Ophthalmol.*, vol. 123, 991-997.
- Harris, C. R., Millman, J. K., Van der Walt, S., Gommers, R., Virtanen, P., Cornapeau, D., . . . Oliphant, T. E. (2020). Array Programming with NumPy. *Nature*, vol. 585, 357-362.
- Hunter, J. D. (2007). Matplotlib: A 2D Graphic Environment. *Computing in Science & Engineering*, vol. 9, nº 3, 90-95.
- Indolia, S., Goswami, A. K., Mishra, S. P., & Asopa, P. (2018). Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach. *Procedia Computer Science*, vol. 132, 679-688.

- Kareen, S., Gelman, R., Martinez-Perez, E. M., Jiang, L., Berracal, A. M., Hess, D. J., . . . Chiang, M. F. (2007). Evaluation of Computer-Based System for Plus Disease Diagnosis in Retinopathy of Prematurity. *Ophthalmology*, 59-67.
- Kim, T.-Y., Son, J., & Kim, K. (2011). The Recent Progress in Qualitative Medical Image Analysis for Computer Aided Diagnosis Systems. *Healthcare Informatics Research*, 143-149.
- Komal, A., & Subhadra, J. (2018). Classification of Retinopathy of Prematurity: From Then till Now. *Community Eye Health Journal*, vol. 31, nº 101, S4-S7.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet Classification with Deep Convolutional Neural Networks. *Communication of the ACM*, vol. 60, nº 6, 84-90.
- Li, Q., Feng, B., Xie, L., Liang, P., Zhang, H., & Wang, T. (2016, January). A Cross-modality Learning Approach for Vessel Segmentation in Retinal Images. *IEEE Transactions on Medical Imaging*, vol. 35, 109-118.
- Liskowsky, P., & Krawiec, k. (2016, November 11). Segmenting Retinal Blood Vessels with Deep Neural Networks. *IEEE Transactions on Medical Imaging*, vol. 35, 2369-2380.
- Lundth, F. (2001). *Python Standard Library*. Sebastopol, CA: O'Reilly & Associates, Inc.
- McKinney, W. (2010). Data Structures for Statistical Computing in Python. In *Proceedings of the 9th Python in Science Conference* (pp. 56-61). Stefan van der Walt and Jarrod Willman.
- Mutlu, F. M., & Sarici, S. U. (2013). Treatment of Retinopathy of Prematurity: a review of conventional and promising new therapeutic options. *Int. J. Ophthalmology*, vol.6, nº 2, 228-236.
- Nunez-Iglesias, J., Blanch, A. J., Looker, O., Dixon, M. W., & Tilley, L. (2018). A New Python Library to Analyse Skeleton Images >Confirms Malaria Parasite Remodelling of the Red Blood Cell Membrane. *PeerJ*, vol. 6, 4312.
- RetCam Envision™ ophthalmic imaging system*. (2021). Retrieved from Natus: <https://newborncare.natus.com/products-services/eye-imaging/retcam-envision>
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Network for Biomedical Image Segmentation. In N. Navab, J. Hornegger, W. Wells, & A. Frangi, *Medical Image Computing and Computer-Assisted Intervention - MICCAI* (pp. 234-241). Munich: Springer, Cham.

- Roychowdhury, S., Koozekanani, D. D., & Parhi, K. K. (2015). Blood Vessel Segmentation of Fundus Images by Major Vessel Extraction and Sub-Image Classification. *IEEE Journal of Biomedical and Health Informatics*, 1118-1128.
- Saugstad, O. D. (2006). Oxygen and Retinopathy of Prematurity. *Journal of Perinatology*, vol. 26, S45-S50.
- Shah, P. K., Prabhu, V., Karandikar, S. S., Ranjan, R., Narendran, V., & Kalpana, N. (2016). Retinopathy of Prematurity: Past, Present and Future. *World Journal of Clinical Pediatrics*, 35-46.
- Soares, J. V., Leandro, J. J., Cesar Jr., R. M., Herbert, J. F., & Cree, M. J. (2006, SEPTEMBER 9). Retinal Vessel Segmentation Using 2-D Gabor Wavelet and Supervised Classification. *IEEE Transactions on Medical Imaging*, vol. 25, 1214-1222.
- Song, Y.-Y., & Lu, Y. (2015). Decision Tree Methods: Application for Classification and Prediction. *Shanghai Archives of Psychiatry*, vol. 27, nº2, 130-136.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *The Journal of Machine Learning Research*, vol. 15, nº1, 1929-1958.
- Srivastava, N., Hinton, G., krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Network from Overfitting. *Journal of Machine Learning Research*, vol. 15, 1929-1958.
- Staal, J., Abramoff, M. D., Niemeijer, M., Viergever, M. A., & van Ginneken, B. (2004). Ridge-Based Vessel Segmentation in Color Images of the Retina. *IEEE Transaction on Medical Imaging*, vol. 23, 501-509.
- T., T. L. (1942). Fibroblastic Overgrowth of persistent Tunica Vasculosa Lentis in Infants Born Prematurely, II. Reports of Cases-Clinical Aspects. *Trans Am OphthalmolSoc*, vol. 40, 262-268.
- Tadesse, M., Dhanireddy, R., Mittal, M., & Higgins, R. D. (2002). Race, Candida Sepsis, and Retinopathy of Prematurity. *Biol Neonate*, vol. 81, 86-90.
- Tong, Y., Lu, W., Deng, Q.-q., Chen, C., & Shen, Y. (2020). Automated Identification of Retinopathy of Prematurity by Image-Based Deep Learning. *Eye and Vision*, vol. 7, nº 1.
- Umesh, P. (2012). Image Processing in Python. *CSI Communications*, 23-24.

- Van der Walt, S., Schonberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., . . . Yu, T. (2014). Scikit-Image:Image Processing in Python. *PeerJ*, 453.
- Van Rossum, G. (2020). *The Python Library Reference*. Python Software Foundation.
- Van Rossum, G., & Drake, F. L. (2009). *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.
- Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., . . . Van Mulbregt, P. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, vol. 17, 261-272.
- Walczac, S., & Cerpa, N. (2003). Artificial Neural Networks. In S. Walczac, & N. Cerpa, *Encyclopedia of Physical Science and Technology* (pp. 631-645). Tarzana, CA: Academic Press inc.
- Waskom, M. L. (2021). Seaborn: Statistical Data Visualization. *Journal of Open Source Software*, vol. 6, n° 60, 3021.
- Yann, L., Bengio, Y., & Hinton, G. (2015). Deep Learning. *Nature*, vol. 521, n°7553, 436-444.

Annex A1 Specification RetCam

RetCam Envision™

Technical Specifications

General Device Information

Applied Part Classification	Type BF (body floating)
Applied Parts	Handpiece and Lens piece
Ingress Protection Rating (IP code)	ME Equipment: IPX0 Monitor: IPX1 Footswitch: IPX8

Device Characteristics

Device Dimensions (WxDxH-minimum)	660 x 610 x 1372 mm (26 x 24 x 54 in)
Device Weight	With FA (61-000300): Approximately 85 kg (188 lb) fully loaded Without FA (61000400): Approximately 82.6 kg (182 lb) fully loaded
Display Properties	21.5" IPS touchscreen HD 1920 x 1080 monitor
CPU and Storage	Intel® Core i7 16 GB Random Access Memory 1 TB Solid State Storage Drive

Operating System

Microsoft Windows® 10 Operating System (OS)

Software	v7.0
Instrument Control Firmware	v1.0

Electrical/Power Supply

Ratings	100-240 V~, 50/60 Hz, 400 VA
Fuses	Fuses: 3AG 6.3A 250V slo-blo 5 x 20 mm
Power Consumption	400 W maximum with all options
Power Cord	Meeting the following specifications: Detachable hospital-grade 305 cm (10 ft) maximum length Rated at 10A (minimum) UL/CSA listed REACH, RoHS compliant
Battery LI-ION (2)	14.4V
Battery Charging Module	240W
Full Battery Charge Operation (battery run time when AC supply is interrupted)	10 mins
Battery Recharge Time	If completely discharged, charge a minimum of 5 hours

Environmental Conditions for Use

Operating Environmental Limits

Temperature	10° – 35° C (50° – 95° F)
Relative Humidity	30% – 90% noncondensing
Atmospheric Pressure	70 – 106 kPa (20.7 – 31.3" Hg)
Altitude	-382 – 3012 m (-1255 – 9822 ft)

Storage (unboxed)

Temperature	-10° to 55° C (14° to 131° F)
Relative Humidity	30% – 90% noncondensing
Atmospheric Pressure	70 – 106 kPa (20.7 – 31.3" Hg)
Altitude	-382 – 3012 m (-1255 – 9822 ft)

Transport (in original shipping material)

Temperature	-29° – 50° C (-20° – 122° F)
Relative Humidity	10% – 85% noncondensing
Atmospheric Pressure	50 – 106 kPa (14.7 – 31.3" Hg)
Altitude	-382 – 5574 m (-1255 – 18,288 ft)



*NOTE: Specifications are subject to change without notice.

natus.

RetCam Envision™

Applicable Standards

Australia
EN ISO 15223-1:2016
Medical devices — Symbols to be used with medical device labels, labelling and information to be supplied — Part 1: General requirements
EN 60601-1:2006
Medical electrical equipment - Part 1: General requirements for basic safety and essential performance
EN 60601-1-6:2010
Medical electrical equipment - Part 1-6: General requirements for basic safety and essential performance - Collateral standard: Usability (IEC 60601-1-6:2010)
IEC 60601-1-2:2014 – EMC Fourth Edition
Medical electrical equipment - Part 1-2: General requirements for basic safety and essential performance - Collateral Standard: Electromagnetic disturbances - Requirements and tests
IEC 62304:2006+AMD1:2015
Medical device software - Software life cycle processes
ISO 10940:2009
Ophthalmic Instruments - Fundus Cameras

Canada
CAN/CSA-C22.2 NO. 60601-1:14 (R2018)
Medical Electrical Equipment - Part 1: General Requirements for Basic Safety and Essential Performance (Adopted IEC 60601-1:2005, third edition, 2005-12, including amendment 1:2012, with Canadian deviations)
CAN/CSA-C22.2 NO. 60601-1-6:11 + A1 (R2016)
Medical electrical equipment - Part 1-6: General requirements for basic safety and essential performance - Collateral standard: Usability (Adopted IEC 60601-1-6:2010, third edition, 2010-01)
IEC 60529
Degrees of protection provided by enclosures (IP Code)
IEC 62304:2006+AMD1:2015
Medical device software - Software life cycle processes
IEC 62366:2014 - Ed. 1.1
Medical devices – Application of usability engineering to medical devices

EU
EN ISO 15223-1:2016
Medical devices — Symbols to be used with medical device labels, labelling and information to be supplied — Part 1: General requirements
EN 60601-1:2006
Medical electrical equipment - Part 1: General requirements for basic safety and essential performance
EN 60601-1-6:2010
Medical electrical equipment - Part 1-6: General requirements for basic safety and essential performance - Collateral standard: Usability (IEC 60601-1-6:2010)
EN 60601-1-2:2015
Medical electrical equipment - Part 1-2: General requirements for basic safety and essential performance - Collateral Standard: Electromagnetic disturbances - Requirements and tests
IEC 60529
Degrees of protection provided by enclosures (IP Code)
IEC 62133:2012
Secondary cells and batteries containing alkaline or other non-acid electrolytes - Safety requirements for portable sealed secondary cells, and for batteries made from them, for use in portable applications

EN 62304:2006
Medical device software - Software life-cycle processes

Japan
IEC 60601-1: 2005+A1:2012+Cor1:2014
Medical electrical equipment - Part 1: General requirements for basic safety and essential performance


USA
ANSI AAMI ES60601-1:2005(R)2012 and A1:2012
C1:2009(R)2012 and A2:2010(R)2012 (Consolidated Text) Medical electrical equipment - Part 1: General requirements for basic safety and essential performance (IEC 60601-1:2005 MOD)
EN ISO 15223-1:2016
Medical devices — Symbols to be used with medical device labels, labelling and information to be supplied — Part 1: General requirements
IEC 60601-1-6:2010+A1:2013
Medical electrical equipment - Part 1-6: General requirements for basic safety and essential performance - Collateral standard: Usability
IEC 60601-1-2:2014 – EMC Fourth Edition
Medical electrical equipment - Part 1-2: General requirements for basic safety and essential performance - Collateral Standard: Electromagnetic disturbances - Requirements and tests
IEC 62133:2012
Secondary cells and batteries containing alkaline or other non-acid electrolytes - Safety requirements for portable sealed secondary cells, and for batteries made from them, for use in portable applications
IEC 62304:2006+AMD1:2015
Medical device software - Software life cycle processes
ISO 10940:2009
Ophthalmic Instruments - Fundus Cameras
UL2054:2011
Standard for Household and Commercial Batteries
UL1642:2009
UL Standard for Safety for Lithium Batteries

Worldwide
ANSI Z80.36:2016
For Ophthalmics - Light Hazard Protection for Ophthalmic Instruments
EN ISO 13485:2016
Medical devices - Quality management systems - Requirements for regulatory purposes
ISO 15004-2:2007
Ophthalmic instruments — Fundamental requirements and test methods — Part 2: Light hazard protection
ISO 10993-1
Biological evaluation of medical devices - Part 1: Evaluation and testing within a risk management process
ISO 14971:2019
Medical Devices – Application of Risk Management to Medical Devices
ISO 15223-1:2016, Corrected version 2017-03
Medical devices — Symbols to be used with medical device labels, labelling and information to be supplied — Part 1: General requirements
ISTA 2A Shipping Tests
International Safe Transit Association
ISTA 2B Shipping Tests
International Safe Transit Association

*NOTE: Specifications are subject to change without notice.

Healthcare solutions with one thing in mind. You.

©2020 Natus Medical Incorporated. All Rights Reserved. All product names appearing on this document are trademarks or registered trademarks owned, licensed to, promoted or distributed by Natus Medical Incorporated, its subsidiaries or affiliates. 033371 RevC



Natus Medical Incorporated
natus.com

Annex All Images for Classifier Training

