Hybridization CNN and Fuzzy Logic approch for the Identification of Intraocular Eye Disease

Sahraoui Mustapha[1][0000-0002-9049-3229] and Mahmoudi Laouni[2] [0000-0001-6517-515X]

[1 2] University of Mascara, Algeria

sahraoui.musta@univ-mascara.dz
laouni.mahmoudi@univ-mascara.dz

Abstract.

Ocular hypertonia, a major risk factor for glaucoma, leads to gradual retinal degradation and is often asymptomatic in its early stages, making early detection challenging. In recent years, the use of Artificial Intelligence (AI), particularly deep learning, has significantly enhanced the analysis and interpretation of medical images for diagnostic purposes.

This project aims to detect and classify ocular hypertension through the analysis of fundus images using deep learning. A hybrid approach combining a Convolutional Neural Network (U-Net) with fuzzy logic known as a Neuro-Fuzzy system was employed. The U-Net model automatically extracts features from preprocessed fundus images and classifies them into two categories: diseased or normal. Fuzzy logic is then applied to refine this classification into three stages, improving diagnostic precision.

The integration of U-Net with fuzzy logic has demonstrated strong potential in retinal pathology assessment, offering improved handling of uncertainty and variability inherent in medical data

Keywords: Artificial Intelligence, fuzzy Logic, hypertensive ocular, convolutional neural networks (CNN), U-Net, optic nerve head (ONH)

1. Introduction

Intraocular pressure (IOP) is the most important modifiable risk factor for glaucoma and fluctuates considerably within patients over short and long time periods Our field's understanding of IOP has evolved considerably in recent years, driven by tonometric technologies with increasing accuracy, reproducibility, and temporal resolution that have refined our knowledge regarding the relationship between IOP and glaucoma risk and pathogenesis

Glaucoma is defined as progressive optic neuropathy that damages the structural appearance of the optic nerve head and is characterized by permanent blindness.

In ophthalmology, intraocular pressure refers to a group of eye diseases that affect the optic nerve. This damage is primarily caused by elevated intraocular pressure (IOP), particularly in adults and individuals over the age of 55 [1], [3]. Various approaches have been developed to detect retinal diseases such as syndrome of Ocular hypertonia .For instance, successfully trained a machine learning model using the publicly available Retinal Fundus Images for Glaucoma Analysis (RIGA) [8] dataset, achieving notable accuracy. Similarly, in [9] were able to automatically measure the cup-to-disc ratio (CDR) by conducting online contour analysis of retinal images.

In other hand Ophthalmology is the medical specialty that makes greater use of artificial intelligence. While the use of classification and segmentation algorithms based on machine learning and computer vision techniques demonstrated effective and correct diagnosis of this eye disease [15].

Convolutional neural networks (CNNs) with more hidden layers have a more complicated network structure and stronger feature learning and feature expression capabilities than conventional machine learning techniques as a result of the advent of the massive data era.

The remainder of this paper is organized into seven sections. Section 2 provides a concise overview of the relationship between intraocular pressure syndrome and glaucoma. Section 3 introduces fundamental principles of fuzzy logic. Section 4 presents the main concepts underlying Convolutional Neural Networks (CNNs). In Section 5, we describe the proposed CNN and fuzzy logic-based approach for glaucoma detection. Sections 6 and 7 detail the experimentation and implementation of our methodology, which was validated through Python simulations. Finally, the concluding section offers a comparative analysis of the proposed method and the VGGNet-16 architecture, demonstrating reliable and accurate results across multiple test cases

2. The relationship between intraocular pressure and glaucoma

2.1 Glaucoma

In the realm of ophthalmology, the glaucoma family of eye disorders affects the optic nerve. Glaucoma is associated with intraocular problems (eye pressure, IOP), although other causes include excessive blood pressure, migraines, ethnicity, and family history. Elevated IOP in adults and people over 60 years of age causes damage to the optic nerve [1] [3]. The area of the optic disc (OD) known as the optic nerve head (ONH) is where optic nerve degeneration takes place. Every type of glaucoma is incurable, and the majority of its damage cannot be repaired.

Glaucoma is a progressive, neurodegenerative optic neuropathy of multi factorial etiology which results in the death of retinal ganglion cells (RGCs). Intraocular pressure (IOP) is the most important identifiable and, presently, the only modifiable risk factor for glaucoma. IOP stress initiates an RGC axonal injury at the level of the optic nerve head. [4].

Reducing the disease's rate of progression is one of the options available to sufferers. Therefore, early disease resolution is essential to the survival of all medications. Early conclusions are rare, however, due to the absence of self-evident indications. We were able to examine glaucoma by measuring the diameters of the OC and OD (a discouragement within the OD) by the examination of fundus photographs. The CDR for the normal eye is 0.65, according to [5] (see Fig. 1).

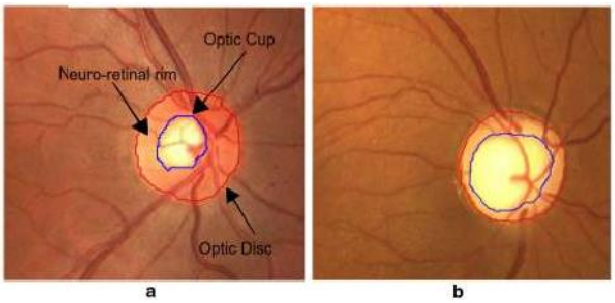


Fig.1. Cropped digital fundus photos surrounding the optic disc. **a** healthy optic disc's main structures, **b** a glaucoma optic disc's.

In the context of image analysis, datasets are collections of relevant images with well defined regions of interest. The number of photos can range from a few hundred to thousands. It has been shown that analysis quality significantly improves with dataset size. [2] [17] [16].

2.2 Intraocular pressures

Recorded observations of blind eyes with nonreactive pupils that felt hard to palpation date back to the ancient Greeks, though it wasn’t until the early 19th century that Gurthrie coined the term “glaucoma” to describe a blinding eye disease associated with increased ocular tension [19] .As technology to accurately measure IOP advanced over the course of the following century and population-based IOP distributions in healthy individuals were measured, the association between glaucoma And IOP was considered an absolute –patients with an IOP greater than 2 standard deviations above the mean of the population average (frequently cited as being 21 mmHg) were considered to have a disease warranting treatment. Mansour Armaly is credited as being among the first to challenge this notion by reporting a cohort of individuals with ocular hypertension who did not develop visual field defects over a 7-year follow-up period, [20] a finding later substantiated by the Ocular Hypertension Treatment Study (OHTS) [18] [3].

Thus, our understanding of the relationship between IOP, glaucoma, and optic nerve pathology, has evolved considerably. It is now recognized that susceptibility to IOP depends on individual characteristics related to the eye’s response to varying IOP levels, since patients who develop glaucoma at IOP within the population normative range still benefit from IOP reduction [25] and many people tolerate an IOP well above the normal range for decades without developing optic neuropathy [18].

3. Fuzzy logic

A neuro-fuzzy system is a hybrid intelligent system that combines the strengths of neural networks and fuzzy logic. It leverages the learning capabilities of neural networks with the reasoning and interpretability of fuzzy logic, allowing for more effective modeling, reasoning, and decision-making in complex, uncertain, and dynamic environments.

3.1 Neuro-Fuzzy Systems

Neuro-Fuzzy Systems = Neural Networks + Fuzzy Logic .

Neuro-fuzzy systems are artificial intelligence systems that combine fuzzy logic with artificial neural networks. They are capable of processing complex and uncertain data and can solve classification and prediction problems in a way that closely resembles human reasoning (see Fig. 2) .

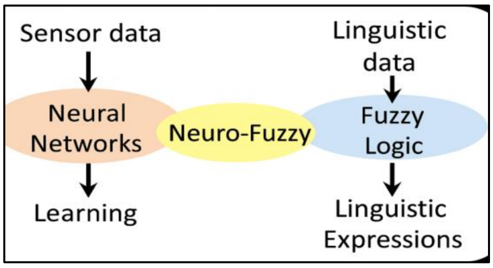


Fig 2: Principle of the Neuro-Fuzzy system .

4. Convolution neural networks

Convolutional Neural Networks (CNNs) are deep learning architectures specialized in processing image and visual data. They consist of hierarchical layers that automatically learn spatial features from input images. The core components include convolutional layers for local feature extraction, pooling layers for dimensionality reduction and overfitting control, and fully connected layers for final classification. Non-linearity is introduced through activation functions like ReLU, while normalization and dropout techniques enhance model generalization. CNN architectures range from simple models such as LeNet and U-Net to more complex ones like VGG, ResNet, and Inception. CNNs have achieved outstanding performance in image classification, object detection, and medical image analysis, particularly in detecting retinal diseases such as glaucoma. (see Fig. 2). [15]

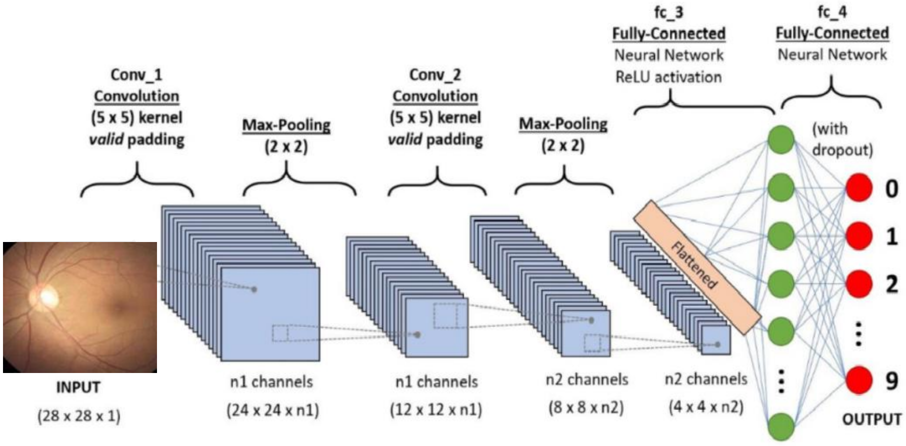


Fig.2. Typical convolutional neural network (CNN) structure

4.1. **Convolutional layers**

Convolutional Neural Networks (CNNs) extract localized features from input data using filters in convolutional layers, which reduces the number of parameters and supports deeper model architectures [6]. By preserving spatial relationships and minimizing issues like vanishing or exploding gradients, CNNs are particularly well-suited for image analysis. [9][10][15].

4.2. **Pooling layers**

In Convolutional Neural Networks (CNNs), pooling layers are used after convolutional layers to perform downsampling by reducing the spatial dimensions of feature maps. This process minimizes computational complexity and the number of parameters, thus helping to mitigate overfitting. Pooling can be applied locally or globally, with the most common techniques being max pooling selecting the maximum value in a region and average pooling calculating the mean. Typically, a 2×2 filter with a stride of 2 is used, halving the spatial resolution while maintaining the depth of the feature maps. [6].

$$f_{X,Y}(S) = max_{a,b=0}^{1} S_{2X+a,2Y+b} \qquad (1)$$

Stride regulates the assignment of depth columns around the width and height, while the depth of the output volume limits the number of neurons in a layer that connect to the same area of the input volume (see Fig. 3).

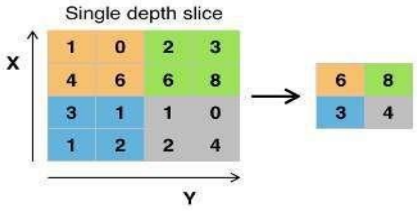


Fig.3 : With a 2x2 filter and a stride of 2, maximum pooling

4.3. **Receptive field**

In neural networks, particularly in CNNs, each neuron receives input from a specific region of the previous layer, known as its receptive field. This region is typically a fixed-size square, such as 5×5 neurons. In fully connected layers, the receptive field encompasses the entire previous layer. As layers are stacked, the receptive field expands in coverage but not in pixel count, enabling neurons to capture broader contextual information while preserving computational efficiency[7].

4.4. Weights

Each neuron in a neural network computes an output by applying a function to inputs from its receptive field, governed by a set of weights and a bias. During training, these parameters are adjusted to improve performance. A key advantage of CNNs is weight sharing: multiple neurons use the same filter across different regions of the input. This significantly reduces the number of learnable parameters and memory requirements, enhancing computational efficiency. [8].

4.5. ReLU layer

The non-saturating activation function [13] is applied by the name "rectified linear unit" (abbreviated "ReLU"), as (2) illustrates:

$$f(x) = \max(0, x) \qquad (2)$$

Negative values are successfully removed from an activation map by setting them to zero [13].

4.6. Loss layer

The loss layer, positioned at the bottom of a neural network, quantifies the error between the model's prediction and the actual target value. In classification tasks involving discrete variables (e.g., 0 or 1 for class membership), the cross-entropy loss function is widely adopted. Rooted in information theory, cross-entropy measures the divergence between predicted and true probability distributions, making it particularly effective for evaluating classification performance [4].

$$loss(x, class) = -\sum_{class=1}^{c} \mathrm{Y}_{\mathrm{x,class}} \log(P_{x,class}) \qquad (3)$$

Given that there are C classes, Y is the estimated chance that x belongs to class I, and P is the actual probability.[13]

5. CNN based glaucoma detection approach

As part of this work, we are tasked with designing and developing a system that classifies the different stages of intraocular hypertension based on fundus (retinal) images. The classification is carried out in three stages according to the severity of the disease.

The retinal image data is divided into two sets: training data and testing data. To ensure the accuracy and validity of our model, the data undergoes a preprocessing phase. We then use a deep learning model called U-Net (based CNN), specifically designed for medical image classification. The model takes the preprocessed training images as input and is trained on a portion of this data to learn the key features of each image, in order to produce an accurate estimation of the probability of the presence or absence of the disease. (seeFig4)

The final system is capable of classifying a new retinal image (the patient's fundus image) using the pre-trained model. If the model predicts that the image contains signs of the disease with a probability higher than 30%, we apply fuzzy logic to determine the disease stage using specific fuzzy inference rules. Otherwise, the patient is considered normal.

This method can assist healthcare professionals in diagnosing potential eye diseases quickly and accurately, which can improve treatment outcomes and help preserve patients' vision in the long term. (seeFig.4)

5.1. **Learning phase**

The CNN glaucoma detection approach's initial step is givin in the flowchart below (see Fig 4). The first step in this learning phase is to import the dataset, which is then trained using several pre-processing steps. Lunch is the next step in the CNN training process, which yields a trained model that can be saved.

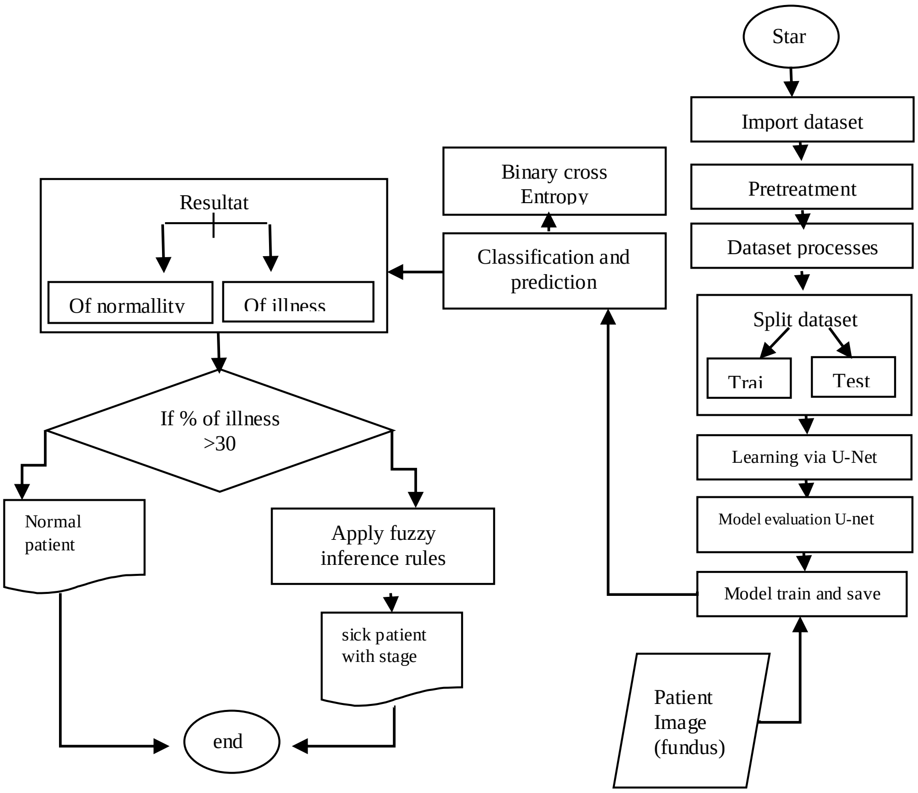


Fig.4 : Flowchart represents the algorithmic steps for classification face

5.2. Classification phase

The patient's fundus image serves as the basis for the second component of the CNN-based glaucoma diagnostic method. Following CNN training, the model is produced (see Fig. 4). The binary function's output represents the decision. According to Fig. 4, the latter is understand in two situations: either the patient has glaucoma disease or not.

6. Experimentation

6.1. Datasets settings

The set of data used in this study is publicly available and was downloaded from the KAGGLE website, which is intended for the diagnosis of intraocular hypertension. The two main classes that make up this ensemble are called " sick " for photographs of sick patients and "Normal " for images of healthy patients.

The data base includes 4865 retinal pictures in total, varying in shape and length, created as follows:

- 1556 photos for the "Normal " class; 3309 photos for the "glaucoma " class. (see Fig.5)

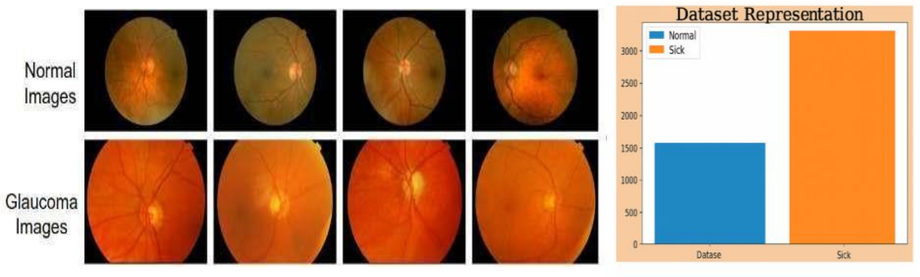


Fig.5. Examples of new publicly available dataset (KAGGLE-Data). Normal and glaucoma fundus images

6.2. Dataset Preparation

Before feeding data into the CNN (U-Net), image processing techniques must be applied to ensure the data is suitable for building an effective deep learning model. In our work, data preparation involves three main steps:

Preprocessing and spliting dataset

Images are processed to enhance clarity and compatibility with the model. This step includes resizing, grayscale conversion, and thresholding (see Fig.6 a).

The dataset was divided into two sets: a training set (80%) containing 3891 images, and a test set (20%) containing 974 images (see Fig.6 b).

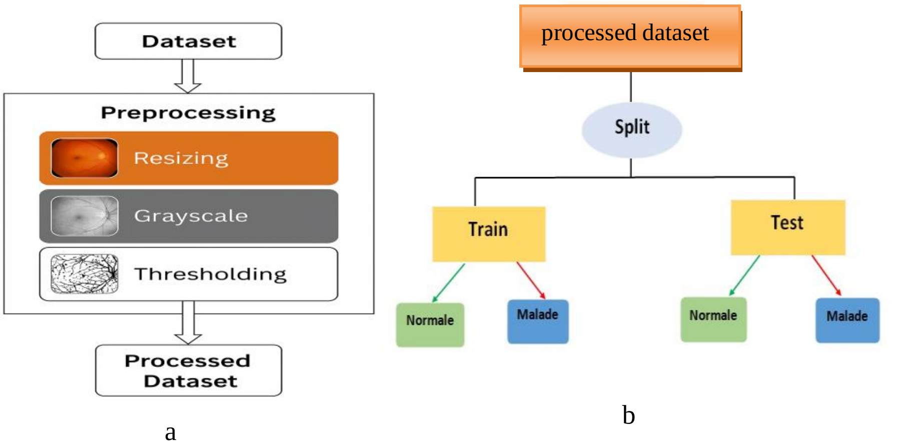


Fig.6 : a) step of processing includes resizing, grayscale conversion, and thresholding b) Splitting of dataset

- **Dataset-Augmentation**

  Data augmentation addresses overfitting issues by increasing the dataset size. The dataset provided by Kaggle is imbalanced. To balance it, data augmentation is necessary. The following techniques were applied to both the training and test sets (see Fig 7.a, 7.b):
- Image normalization
- Image rotation
- Image flipping (horizontal and vertical)
- Image zooming

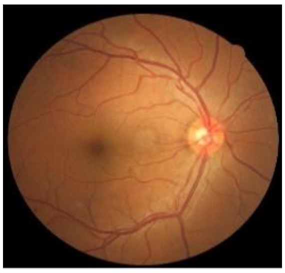


Fig 7.a: Original image.

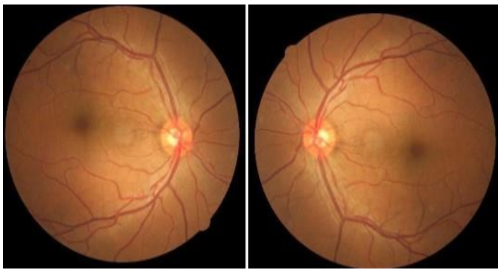
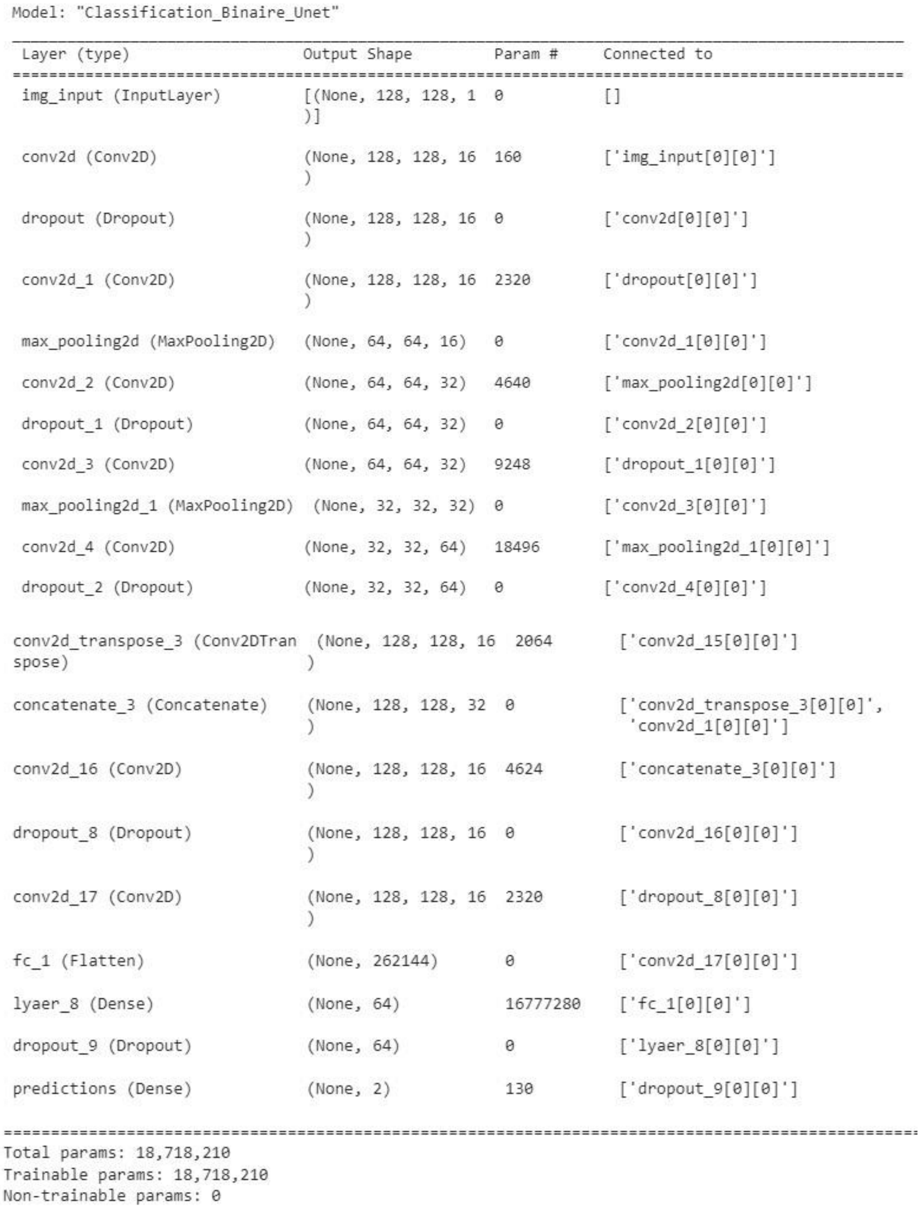


Fig 7.b: Images after horizontal flip (left) and vertical flip (right).

6.3. U-Net Model Creation

The architecture of our model is quite large, so we have illustrated it in the figures below:



```
Model: "Classification_Binaire_Unet"
_____________________________________________________________________________________________
 Layer (type)                   Output Shape         Param #     Connected to
=============================================================================================
 img_input (InputLayer)         [(None, 128, 128, 1  0           []
                                )]

 conv2d (Conv2D)                (None, 128, 128, 16  160         ['img_input[0][0]']
                                )

 dropout (Dropout)              (None, 128, 128, 16  0           ['conv2d[0][0]']
                                )

 conv2d_1 (Conv2D)              (None, 128, 128, 16  2320        ['dropout[0][0]']
                                )

 max_pooling2d (MaxPooling2D)   (None, 64, 64, 16)   0           ['conv2d_1[0][0]']

 conv2d_2 (Conv2D)              (None, 64, 64, 32)   4640        ['max_pooling2d[0][0]']

 dropout_1 (Dropout)            (None, 64, 64, 32)   0           ['conv2d_2[0][0]']

 conv2d_3 (Conv2D)              (None, 64, 64, 32)   9248        ['dropout_1[0][0]']

 max_pooling2d_1 (MaxPooling2D)  (None, 32, 32, 32)  0           ['conv2d_3[0][0]']

 conv2d_4 (Conv2D)              (None, 32, 32, 64)   18496       ['max_pooling2d_1[0][0]']

 dropout_2 (Dropout)            (None, 32, 32, 64)   0           ['conv2d_4[0][0]']

conv2d_transpose_3 (Conv2DTran  (None, 128, 128, 16  2064        ['conv2d_15[0][0]']
spose)                          )

concatenate_3 (Concatenate)     (None, 128, 128, 32  0           ['conv2d_transpose_3[0][0]',
                                )                                 'conv2d_1[0][0]']

conv2d_16 (Conv2D)              (None, 128, 128, 16  4624        ['concatenate_3[0][0]']
                                )

dropout_8 (Dropout)             (None, 128, 128, 16  0           ['conv2d_16[0][0]']
                                )

conv2d_17 (Conv2D)              (None, 128, 128, 16  2320        ['dropout_8[0][0]']
                                )

fc_1 (Flatten)                  (None, 262144)       0           ['conv2d_17[0][0]']

lyaer_8 (Dense)                 (None, 64)           16777280    ['fc_1[0][0]']

dropout_9 (Dropout)             (None, 64)           0           ['lyaer_8[0][0]']

predictions (Dense)             (None, 2)            130         ['dropout_9[0][0]']

=============================================================================================
Total params: 18,718,210
Trainable params: 18,718,210
Non-trainable params: 0
```

6.4. Performance Metrics

Performance metrics are used to evaluate learning algorithms and are a crucial aspect of machine learning. Evaluating the performance of a classification system is a very important phase, as it reflects the reliability of the proposed system. In medical applications, overall accuracy and error rate alone are not sufficient to assess performance.

In this work, we analyzed the behavior of our U-Net classifiers by applying several performance metrics to better evaluate the system and understand its behavior. The adopted metrics are defined as follows [23]:

Confusion matrix

The confusion matrix allows the performance of each class to be evaluated individually. It is a matrix with dimensions equal to the number of classes

Accuracy (Correct Classification Rate)

Accuracy is the most straightforward and natural indicator for evaluating the performance of a classification system. It represents the percentage of correctly identified instances by the system and is easy to calculate [23].

$$\text{Accuracy (ccr)} = \frac{\text{number of correctly identified elements}}{\text{total number of elements}} = \frac{100*(\text{TP}+\text{TN})}{\text{TP}+\text{FP}+\text{TN}+\text{FN}} \quad (4)$$

Precession (Prec)

The percentage of instances classified as positive that are actually positive [71].

$$\text{Prec} = \frac{100*\text{TP}}{\text{TP}+\text{FP}} \quad (5)$$

Recall

The percentage of positive instances that are correct8ly classified by the model [71]

$$\text{Recall} = \frac{100*\text{TP}}{\text{TP}+\text{FN}} \quad (6)$$

F-Measure (F1 Score)

This metric is used to calculate the balance between precision and recall. It is given by the equation below [71]:

$$\mathrm{F1} = \frac{((1+\beta^2)*\mathrm{prec}*\mathrm{recall})}{((\beta^2*\mathrm{prec})+\mathrm{recall})} \qquad (7)$$

The parameter β allows weighting either precision or recall; it is generally set to 1.

7. Simulation Results and Discussion

Before experimenting with the final model designed for intraocular hypertension classifiers, we conducted several parametric tests on two models: U-Net and VGG16, then compared their results. Table.1 presents the results obtained by our U-Net model on three different datasets, each with varying image sizes. In each case, accuracy and loss values were calculated, along with the execution time. These measurements help evaluate both the quality and speed of the results achieved for the different classification tasks performed.

Table .1: Representation of the three different datasets.

	Number of Images	**Diseased Class**	**Normal Class**	**Number of Iterations**	**Batch Size**	**Accuracy**	**Loss**	**Time**
Dataset 1	7492	1098	6394	100	80	99%	0.024	1h20 min
Dataset 2	10865	6309	4556	100	80	81%	0.40	2h
Dataset 3	4865	3309	1556	100	80	92%	0.19	25 Min

8.1. **Presentation of obtained Performance**

To visualize the performance of our deep learning U-Net model over time during training, we created graphs for Accuracy, Loss, Recall, Precision, and AUC.(see Fig 8)

- Accuracy (score) and Loss

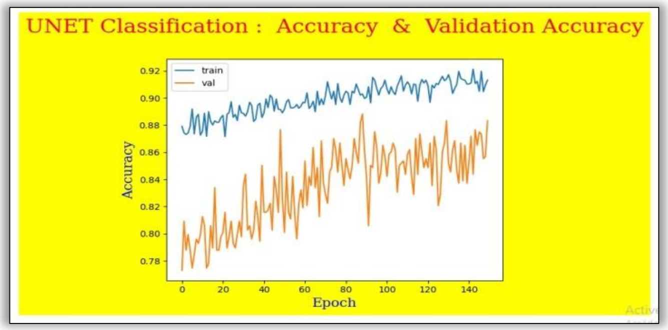


Fig 8 : Accuracy (Score) curve of the U-Net model over 150 epochs.

After examining the results shown in Fig.8, we observe that the accuracy curves for both the training and test sets gradually increase as the number of epochs grows. This indicates that the model is learning new information with each epoch.

Although the training set accuracy reaches a high maximum of 92%, the test set shows a slightly lower accuracy, peaking at 86%. This may suggest that the model is overfitting to the training data and does not generalize well to new data. It is important to note that this overfitting could be due to the relatively small size of the dataset used for this project (about 4,865 images), rather than an inherent weakness of the model.

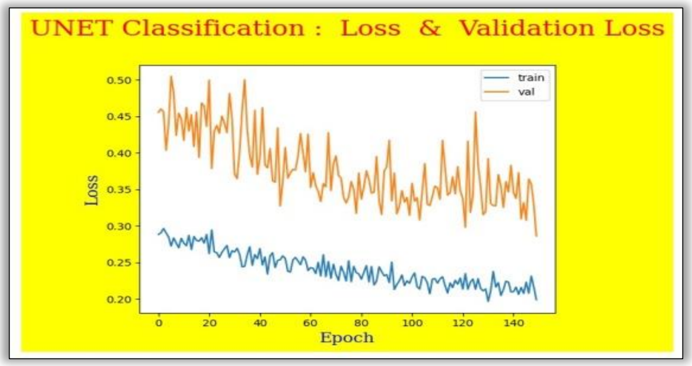


Fig 9 : Loss curve of the U-Net model over 150 epochs.

After analyzing Fig.9, we can observe that the loss curves for both the training and test sets decrease significantly as the number of epochs increases. The initial training loss was around 0.3, but it quickly dropped to a final value of 0.19 after 150 epochs. Similarly, the test loss also gradually decreased, reaching a final value of 0.28 at the same epoch.

This suggests that the model was trained effectively and its performance steadily improved over time. The low loss rates for both training and test sets indicate that the model is capable of providing accurate predictions for both datasets, with minimal classification errors.

- Precision, Recall, and AUC

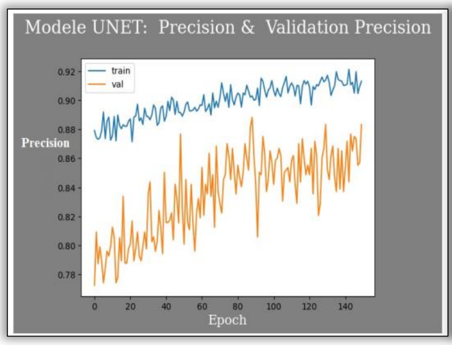


Fig.10 : Precision curve of the U-Net model

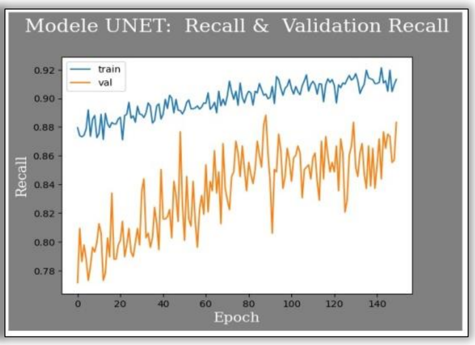


Fig.11: Recall curve of the U-Net model

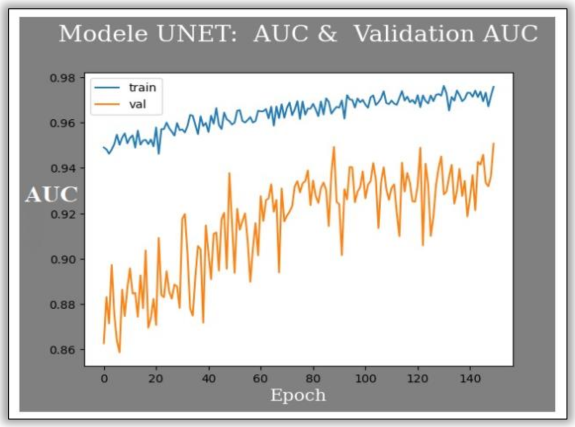
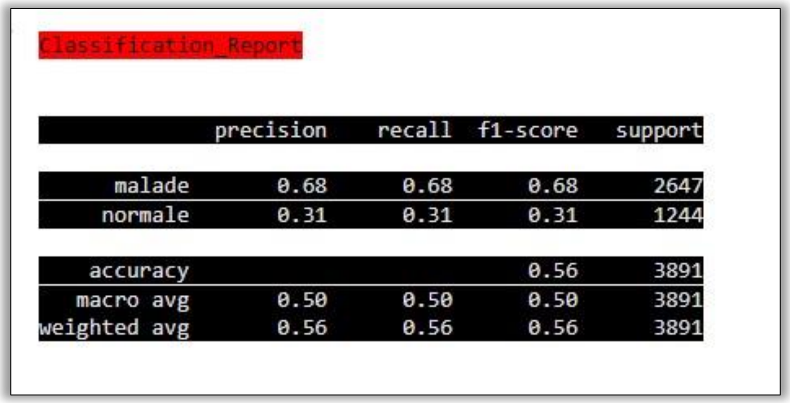


Fig.12: AUC curve of the U-Net model.

After careful evaluation of the results presented in the previous figures, we observe that the trends of the curves are similar and upward. The metrics (Recall, Precision, and AUC) steadily improve over the epochs for both datasets (training and testing). This can be an important indicator of the model’s effectiveness, as these metrics reflect its ability to provide accurate predictions on new data.(see Fig. 10,11,12)

Classification Report

To comprehensively evaluate our model’s performance, we need to analyze precision, recall, and F1 score through a classification report, as shown in Fig.13:



```
Classification_Report

              precision    recall  f1-score   support

      malade       0.68      0.68      0.68      2647
     normale       0.31      0.31      0.31      1244

    accuracy                           0.56      3891
   macro avg       0.50      0.50      0.50      3891
weighted avg       0.56      0.56      0.56      3891
```

Fig.13: Classification report.

When evaluating the classification report results of our U-Net model, we note that the performance varies between the two classes. Indeed, the diseased class has a recall, precision, and F1-score of 0.68, indicating that the model is reasonably able to identify diseased cases with 68% accuracy. However, the normal class has very low recall, precision, and F1-score values of 0.31, suggesting that the model struggles to correctly identify this class.

This may be due to several factors, such as imbalances in the dataset or increased complexity in distinguishing the normal class. It is important to note that the support for the diseased class (2,647) is significantly higher than that of the normal class (1,244), which may also influence classification performance.

Confusion matrix

To understand the types of errors made by our classification model, we will use the confusion matrix, which provides a summary of the prediction results.

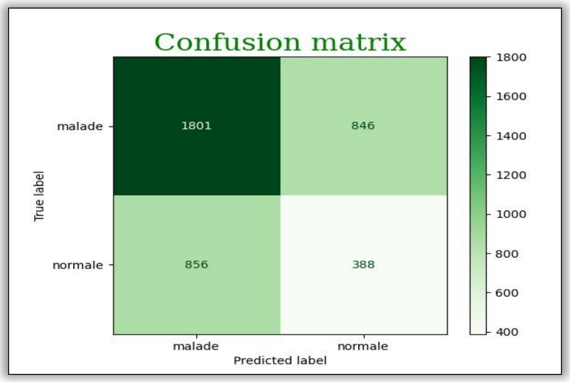


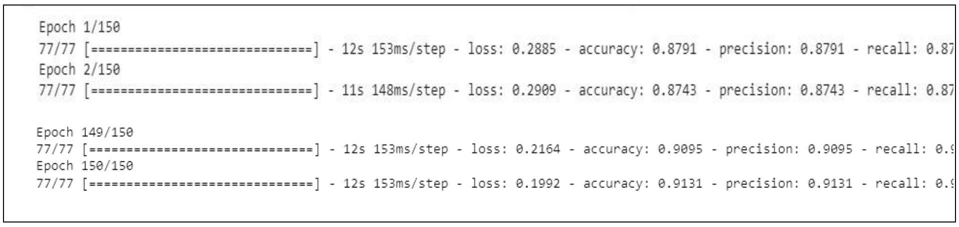
Fig.14: confusion Matrice.

Based on the confusion matrix obtained for our classification model, the performance for the diseased and normal classes can be analyzed as follows: the model was able to correctly identify a large number of diseased cases (TP = 1801). However, the model had a significant number of false positives (FP = 846), meaning it misclassified many normal cases as diseased.

Furthermore, for the normal class, the model misclassified a large number of normal cases as diseased (FN = 856), indicating difficulty in correctly identifying this class. Additionally, the number of correctly identified normal cases (TN = 388) is very low compared to the total number of normal cases.

7.1. Comparison of Results Between U-Net and VGG16 Networks

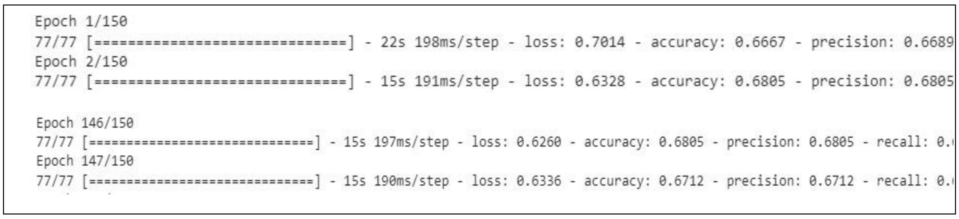
After training our two models, we will discuss the results shown in Fig.15and 16 below:



```
Epoch 1/150
77/77 [==============================] - 12s 153ms/step - loss: 0.2885 - accuracy: 0.8791 - precision: 0.8791 - recall: 0.87
Epoch 2/150
77/77 [==============================] - 11s 148ms/step - loss: 0.2909 - accuracy: 0.8743 - precision: 0.8743 - recall: 0.87

Epoch 149/150
77/77 [==============================] - 12s 153ms/step - loss: 0.2164 - accuracy: 0.9095 - precision: 0.9095 - recall: 0.
Epoch 150/150
77/77 [==============================] - 12s 153ms/step - loss: 0.1992 - accuracy: 0.9131 - precision: 0.9131 - recall: 0.
```

Fig.15: Training results of the U-Net model.



```
Epoch 1/150
77/77 [==============================] - 22s 198ms/step - loss: 0.7014 - accuracy: 0.6667 - precision: 0.6689
Epoch 2/150
77/77 [==============================] - 15s 191ms/step - loss: 0.6328 - accuracy: 0.6805 - precision: 0.6805

Epoch 146/150
77/77 [==============================] - 15s 197ms/step - loss: 0.6260 - accuracy: 0.6805 - precision: 0.6805 - recall: 0.
Epoch 147/150
77/77 [==============================] - 15s 190ms/step - loss: 0.6336 - accuracy: 0.6712 - precision: 0.6712 - recall: 0.
```

Fig.16: Training results of the VGG16 model.

a) The following tables show the training results of our two models for the first and last iterations:

- **Iteration 1:**

Table 2: Training results of the two models for iteration 1**.**

	Accuracy	Loss	Precision	recall	AUC	Val accuracy	Val loss	Val precision	Val recall	Val AUC
U-Net	0.87	0.28	0.87	0.87	0.94	0.78	0.4	0.78	0.78	0.88
VGG16	0.6	0.70	0.66	0.66	0.68	0.69	0.6	0.69	0.68	0.68

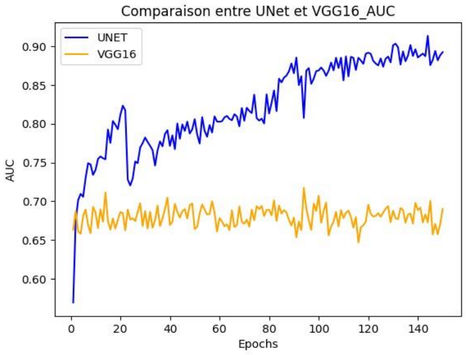
- **Itération 150 :**

Table 3: Training results of the two models for iteration 150.

	Accuracy	Loss	Precision	recall	AUC	Val accuracy	Val loss	Val precision	Val recall	Val AUC
U-Net	0.91	0.1	0.91	0.91	0.97	0.88	0.2	0.88	0.88	0.95
VGG16	0.68	0.6	0.68	0.68	0.68	0.69	0.5	0.69	0.68	0.68

After analyzing the performance of the U-Net and VGG16 models shown in Tables 2 and 3, we conclude that the performance metrics of the U-Net model steadily increase with the number of epochs, indicating that the model continues to learn and improve as training progresses. In contrast, the performance metrics of the VGG16 model show little increase over the epochs and remain almost stable, suggesting that the model quickly reaches a performance plateau.

b) The following figures display the curves of the performance metrics Recall, Precision, and AUC for the U-Net and VGG16 models:



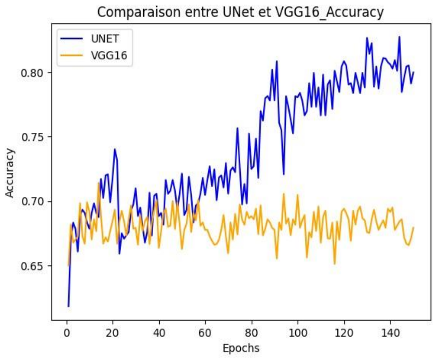
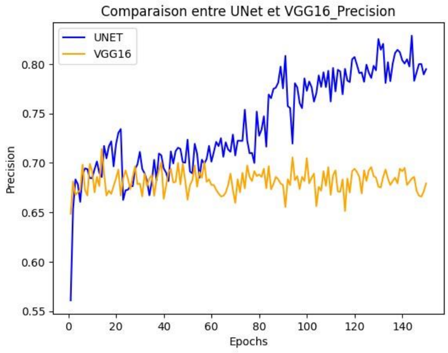


Fig17: AUC of U-Net et VGG16

Fig.18: Accuracy of U-Net et VGG16



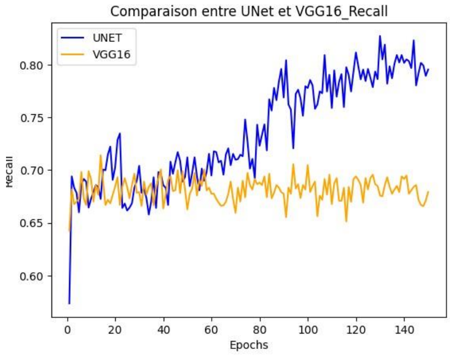


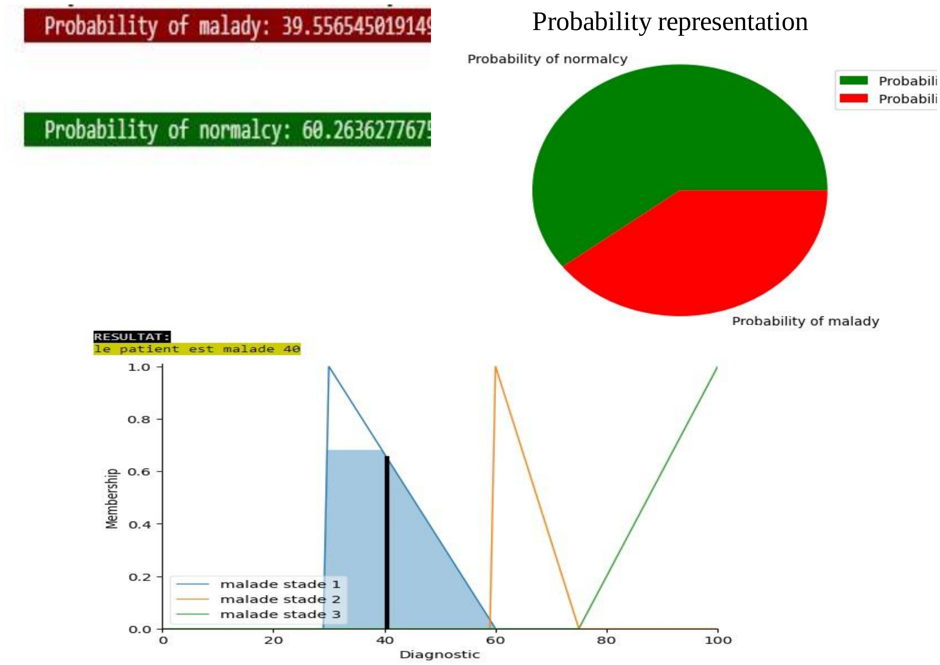
Fig.19: Précision du modèle U-Net et VGG16. **Fig.20**: Recall du modèle U-Net et VGG16.

Based on the obtained results, it is clear that the performance differs between the two networks, with U-Net performing better. The curves show that the VGG16 model is more stable, with performance stabilizing more quickly, although at a lower level.

This can be explained by several possible reasons, notably the specific design of the U-Net model for medical image segmentation. The skip connections used by U-Net to merge features at different resolution levels may enhance its ability to capture finer structures in images, which partly explains why the model's performance continues to improve with more training epochs.

In contrast, the VGG16 model uses a deeper architecture with smaller convolutions but fewer complex feature combinations. It is therefore possible that the problem is more challenging for VGG16, which may explain why its performance plateaus earlier.

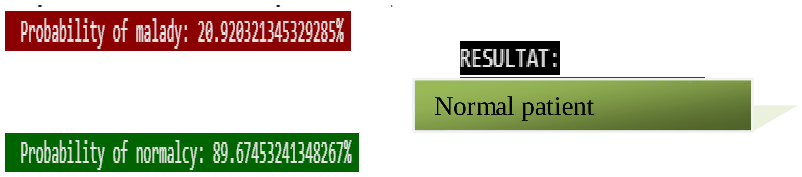
7.2. **Test Results**



```
Probability of malady: 39.556545019149
Probability of normalcy: 60.2636277675
```

Probability representation

Fig.21: Example of a result for a diseased patient**.**



```
Probability of malady: 20.920321345329285%
Probability of normalcy: 89.67453241348267%
RESULTAT:
```

Normal patient

Fig 22: Example of a result for a normal patient.

In this paper, we presented our proposed methodology to design a classification model for the stages of intraocular hypertension, using a convolutional neural network (CNN) specifically U-Net combined with the concept of fuzzy logic. We also reviewed the state of the art regarding CNN applications in medical imaging.

To implement our methodology, we relied on a variety of hardware and software tools. Then, the results obtained by our U-Net model were presented and compared with those from the VGG16 network. This comparison showed that the U-Net model enables precise and reliable classification of the different stages of intraocular hypertension, representing a significant improvement over traditional methods.

Finally,this work creates numerous research opportunities. More research is required to refine some of the concepts that we have introduced.

References

1. F. C. Hollows and P. A. Graham: Intra-ocular pressure, glaucoma,and glaucoma suspects in a denned population. Brit. J. Ophthalmol, 50 (5954089), 570-586, (1966).

2. R. E. W. Rafæl C. Gonzalez: Digital Image Processing, 2008.

H. M. Leibowitz, L. R. Krueger : The Framingham eye study monograph: An ophthalmological and epidemiological study of cataract, glaucoma, diabetic retinopathy, macular degeneration, and visual acuity.J.Surv.Ophthalmol (24),335-610,(1980).

3. D. J. Spalton, R. A. Hitchings, and P. Hunter: Atlas of Clinical Ophthalmology ,Elsevier, Amsterdam (2013).

4. Z. Zhang, F. S. Yin, J. Liu, W. K. Wong, N. M. Tan, B. H. Lee, J. Cheng,and T. Y. Wong : An online retinal fundus image database for glaucoma analysis and research," in Proc. Int. Conf. IEEE Eng. Med. Biol, pp. 3065-3068,(2010).

6. Ciresan, Dan; Ueli Meier; Jonathan Masci; Luca M. Gambardella; Jurgen Schmidhu- ber : Flexible, High Performance Convolutional Neural Networks for Image Classifi- cation. Proceedings of the Twenty-Second International Joint Conference on Artifi- cial Intelligence,vol 2 ,pp 1237–1242. Retrieved (2013).

7. Habibi, Aghdam, Hamed : Guide to convolutional neural networks : a practical appli- cation to traffic-sign detection and classification. Switzer- land. ISBN 9783319575490. OCLC 987790957 (2017).

8. Balas, Valentina E.; Kumar, Raghvendra; Srivastava, Rajshree : Recent Trends and Advances in Artificial Intelligence and Internet of Things. Springer Nature. (2019).

9. Fuente-Arriaga, E. M. Felipe-Riverón, and E. Garduño-Calderón : Application of vascular bundle displacement in the optic disc for glaucoma detection using fundus images, J. A. D. L. Comput. Biol. Med., (47), 222-235, (2014).

10. Ciresan, Dan, Ueli Meier, Jonathan Masci :Flexible, High Performance Convolutional Neural Networks for Image Classification. Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011.

11. Azulay, Aharon; Weiss, Yair: Why do deep convolutional networks generalize so poorly to small image transformations?. Journal of Machine Learning Research. 20 (184), (2019).

12. LeCun, Yann.: convolutional neural networks, Retrieved (2013).

13. Krizhevsky, Alex; Sutskever, Ilya; Hinton, Geoffrey E. :ImageNet classification with deep convolutional neural networks,Communications of the ACM. 60 (6): 84–90. (2017).

14. Stamper, R.L., 2011. A history of intraocular pressure and its measurement. Optom. Vis.Sci. 88 (1), E16–E28. https://doi.org/10.1097/OPX.0b013e318205a4e

15. Chen X, Xu Y, Wong DWK, Wong TY, Liu J: Glaucoma detection based on deep convolutional neural network. Inannual international conference of the IEEE engi- neering in medicine and biology society (EMBC),p.715–8. (2015).

16. World Health Organization. Bulletin of the World Health Organization, http://www. who.int/bulle tin/.lastaccessed 5 May 2016.

17. 3. Bock R, Meier J, Nyúl LG, Hornegger J, Michelson G. Glaucoma risk index: au- tomated glaucoma detection from color fundus images, J.Surv. Ophthalmol(14),471–81. 2015

18. Shubham Joshi, B. Partibane,al :Glaucoma Detection Using Image Processing and Supervised Learning for Classification, Journal of Healthcare Engineering, vol7, p 1-12March (2022).

19. Stamper, R.L., 2011. A history of intraocular pressure and its measurement. Optom. VisSci. 88 (1), E16–E28. https://doi.org/10.1097/OPX.0b013e318205a4e7.

20 Armaly, M.F., 1966. On the distribution of applanation pressure and arcute scotoma. In: Patterson, G., Miller, S.J., Patterson, G.D. (Eds.), Drug Mechanisms in Glaucoma. Little, Brown.

21 Kass, M.A., Heuer, D.K., Higginbotham, E.J., et al., 2021. Assessment of Cumulative incidence and severity of primary open-angle glaucoma among participants in the ocular hypertension treatment study after 20 Years of follow-up. Jama Ophthalmol. 139 (5), 558–566. https://doi.org/10.1001/jamaophthalmol.2021.0341.

22 Anderson, D.R., Hendrickson, A., 1974. Effect of intraocular pressure on rapid axoplasmic transport in monkey optic nerve. Invest. Ophthalmol. 13 (10), 771–783.

23 Anyanwu MN, Shiva SG. Comparative analysis of serial decision tree classification algorithms. International Journal of Computer Science and Security. 2009 Jun; 3 (3): 230-40.