

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE
Université SAAD DAHLAB Blida



Faculté des sciences

Département d'Informatique

MEMOIRE DE MASTER

THÈME

DeepLearning pour le diagnostic de DMLA, glaucome

Et rétinopathie diabétique

Présenté par :

Encadré par :

Hamidat Fella

Mme Zahra Fatma Zohra

IrkiHassiba

Promotion: **2019/2020**

Dédicaces

Je dédie ce mémoire

À mes chers parents ma mère et mon père

À toute ma petite famille

Mes frères et Ma seour

À mes amis

A tous ceux qui, par un mot, m'ont donné la force de continuer...

Remerciements

Je souhaite manifester mes sincères remerciements à Dieu, le tout puissant, pour ses faveurs et ses grâces de m'avoir donné le courage et la patience pour achever ce modeste travail.

*Je tiens à présenter de tout mon cœur mes remerciements à mon binôme **Hamidat Fella** pour son aide et son soutien*

*Aussi mon encadreur Melle «**phd Fatma Zohra Zahra** » pour son aide, ses conseils précieux, sa gentillesse, son encouragement, sa disponibilité et sa confiance qui m'a permis de ne jamais faiblir et de poursuivre toujours plus loin dans mon travail.*

Je remercie naturellement ma famille pour son aide, sa générosité et son soutien moral qui ont été pour moi une source de courage et de confiance.

Je remercie aussi vivement les honorables membres du jury qui ont accepté d'évaluer ce travail.

Remerciements

Au début, je remercie Allah qui nous a aidés à accomplir ce travail,

Je souhaite adresser mes remerciements les plus sincères à ma promotrice Melle Zahra F/Z pour son aide, son suivi, sa disponibilité, ses conseils et sa confiance qui m'a permis de ne jamais faiblir et de poursuivre toujours plus loin dans mon travail.

Je remercie mon binôme IRKI HASSIBA

Nos sincères remerciements à tous les membres du jury qui nous ont fait l'honneur de réviser ce travail.

Je remercie ma famille pour son aide, et son soutien.

Enfin, je remercie mes proches et mes amis, qui m'ont toujours aidé au cours de mon travail.

Résumé

La Dégénérescence Maculaire Liée à l'Age (DMLA), le glaucome et la rétinopathie diabétique sont les principales causes de cécité et de déficience visuelle chez les adultes. La détection précoce de ces maladies par un dépistage régulier est particulièrement importante pour prévenir la perte de vision. Plusieurs erreurs de diagnostic et de suivi peuvent survenir. Par conséquent, un outil d'aide au diagnostic ne sera que bénéfique pour les ophtalmologistes et la santé des yeux des malades.

Les ophtalmologistes se base sur des techniques d'imagerie médicale, l'Angio et l'OCT en grande partie pour le diagnostic et le suivi de l'état des malades pendant la période traitement afin de surveiller l'évolution de la maladie et la réponse des patients aux différents protocoles de traitement..

Le DeepLearning est très performant quand il est appliqué sur des données de type image. Par conséquent, Nous proposons des modèles formés à l'aide d'un réseau de neurones convolutifs afin de bien interpréter les images de l'OCT et l'ANGIO et par conséquent aider l'ophtalmologiste à faire son diagnostic d'une façon plus précise. Les résultats obtenus sont satisfaisants et présentent la robustesse de nos architectures proposées.

Mot clé : DMLA, glaucome, rétinopathie diabétique, Apprentissage supervisé, classification, DeepLearning, Réseaux neuronaux convolutifs.

Abstract

Age-related Macular Degeneration (AMD), glaucoma and diabetic retinopathy are the main causes of blindness and visual impairment in adults. Early detection of these diseases through regular screening is particularly important to prevent vision loss. Many errors in diagnosis and follow-up can occur. Therefore, a diagnostic aid will only benefit ophthalmologists and the health of the eyes of patients.

Ophthalmologists rely on medical imaging techniques, Angio and OCT to a large extent for the diagnosis and follow-up of the patients' condition during the treatment period in order to monitor the evolution of the disease and the patients' response to different treatment protocols.

Deep learning is very effective when applied to image data. Therefore, we propose models trained using a convolutional neural network to properly interpret OCT and ANGIO images and therefore help the ophthalmologist to make a more accurate diagnosis. The results obtained are satisfactory and show the robustness of our proposed architectures.

Key word: AMD, glaucoma, diabetic retinopathy, Supervised learning, classification, deep learning, convolutional neural networks.

Liste des Tableaux

Tableau 1: Algorithmes d'apprentissage automatique et leurs catégories.[40]	65
Tableau 2:Travaux basés sur les techniques de machine Learning classique pour la détection des maladies oculaires (Rétinopathie Diabétique, Glaucome, DMLA)	69
Tableau 3: Tableau présente quelque systèmes utilisant Deeplearning pour détecter les maladies ophtalmiques (Rétinopathie Diabétique RD, Glaucome, Dégénérescence maculaire lié à l'âge DMLA).....	73
Tableau 4 : Tableau présente deux systèmes utilisant Deep Learning combiner avec machine Learning pour prédire les maladies ophtalmiques(Rétinopathie Diabétique, Glaucome).....	77
Tableau 5 :Tableau récapitulatif des différents défis potentiels d'un système de l'IA dans la détection des maladies ophtalmique.[59].....	79
Tableau 6: configuration du colab	102
Tableau 7: Comparaison des performances des modèles	122
Tableau 8: Fonctions d'activations.[31]	4

Liste des figures

Figure 1: Anatomie de l'œil.[8]	31
Figure 2: Fond d'œil.[5].....	32
Figure 3: Schéma de la rétine. [6].....	32
Figure 4: Rétinopathie diabétique minime.[11].....	35
Figure 5 : Œdème maculaire cystoïde sévère.[11].....	35
Figure 6 :Forme précoce -drusens maculaires-. [12]	36
Figure 7: Forme exsudative de DMLA.[12]	36
Figure 8 : glaucome[11].....	37
Figure9:Techniques d'acquisition des images du fond d'œil.[6]	38
Figure 10: L'appareil d'OCT et résultat de l'examen.[6]	39
Figure 11 :OCT normal.[11].....	39
Figure 12 : Les différents plans d'analyse OCT. [15]	40
Figure 13: Schéma d'un modèle supervisé.[20]	44

Figure 14 : Représentation des classifications.[21]	45
Figure 15 : Principe du Séparateur à Vaste Marge (SVM).[22]	47
Figure 16: Arbre de classification déterminant la classe de migration des observations.[21]	49
Figure 17 : Schéma d'un modèle non supervisé. [20]	50
Figure 18 : Boucle (perception+récompense)/action en apprentissage par renforcement[22]	51
Figure 19: Différentes entre apprentissages mal estimés et bon apprentissage. [21]	52
Figure 20 : La relation entre l'intelligence artificielle, le ML et le Deep Learning	53
Figure 21 : Un neurone réel[23].	54
Figure 22 : Un neurone artificiel. [25]	54
Figure 23 : : schéma du perceptron simple.[26]	55
Figure 24 : Perceptron multicouche. [26]	56
Figure 25 :Une illustration du processus de recherche de l'optimum.[27].....	56
Figure 26 : Fonction d'activation linéaire.[31]	2
Figure 27 : Fonction d'activation non linéaire.[31].....	2

Figure 28 : Représentation graphique de la fonction Sigmoidé.[32]	3
Figure 29: Représentation graphique de la fonction Relu.[32].....	4
Figure 30: Comparaison entre la machine Learning et le DeepLearning.[22]	58
Figure 31: : Un réseau entièrement connecté.[22]	59
Figure 32 : Architecture d'un réseau de neurones convolutifs.[35]	59
Figure 33: Illustration des méthodes de convolution et de Pooling.[35]	60
Figure 34 : Réseau récurrent à portes .[36].....	61
Figure 35 : Une chaîne de cellules LSTM.[36]	61
Figure 36 : Classification des travaux de diagnostic des maladies : RD, DMLA, Glaucome	68
Figure 37 : Schéma global de la solution proposée.	85
Figure 38 : Représentation des processus de formation (phase 1), de validation (phase 2) et de test (phase 3) du modèle de classification [60]	86
Figure 39 : Phase de classification par CNN. [61]	89
Figure 40– L’architecture du modèle CNN 3 couches.	91
Figure 41 : L’architecture du modèle CNN 5 couches.	93

Figure 42 – L’architecture du modèle CNN 7 couches.	95
Figure 43: Architecture de VGG [75].....	96
Figure 44:Architecture ResNet-50 préformée	97
Figure 45 :Une architecture du modèle DenseNet121[77]	98
Figure 46. Architecture MobileNet.....	99
Figure 47 : Le logo de python.....	103
Figure 48: (Extrême gauche) Néovascularisation choroïdienne (CNV) .[83].....	105
Figure 49.Distribution de classes de l'ensemble de données	105
Figure 50: Structure de répertoire.	106
Figure 51: Test et accuracy du modèle CNN 3 couches.....	107
Figure 52 : Représentations graphiques de précision et de perte CNN 3 couches	108
Figure 53 : Matrice de confusion CNN 3 Couches.....	108
Figure 54 : Test et Accuracy CNN 5 couches	109
Figure 55 : Représentations graphiques de précision et de perte CNN 5 couches	109
Figure 56 : Matrice de Confusion CNN 5 Couches.....	110

Figure 57 ; Test et Accuracy CNN 7 Couches.....	110
Figure 58 : Représentations graphiques de précision et de perte CNN 7 couches.	110
Figure 59 : Matrice de confusion CNN 7 Couches.....	111
Figure 60 : Test et Accuracy VGG16	111
Figure 61: Représentations graphiques de précision et de perte VGG16.....	111
Figure 62 : Matrice de Confusion VGG16	112
Figure 63: Test et Accuracy ResNet50	112
Figure 64: Représentations graphiques de précision et de perte ResNet50.....	112
Figure 65: Matrice de Confusion ResNet50	113
Figure 66: Test et Accuracy Mobile Net (Module1)	113
Figure 67 : Représentations graphiques de précision et de perte MobileNet(Module1)	113
Figure 68 : Matrice de Confusion MobileNet (Module1).....	114
Figure 69 : Test et Accuracy DenseNet121	114
Figure 70 : Représentations graphiques de précision et de perte DenseNet121	114
Figure 71 : Matrice de Confusion DenseNet121	115

Figure72 : Comparaison des performances des modèles.....	115
Figure 73 : représente La phase de prédiction(Module1).....	116
Figure 74 : Exemple de glaucome.	118
Figure 75: la distribution de jeu de données (train, val, test).....	118
Figure 76: représentation deX_train	119
Figure 77: exemples d'image fond d'œil glaucome	120
Figure 78: Test et AccuracyRetinal _MobileNet.....	120
Figure 79: Représentations graphiques de précision et de perte Retinal_MobileNet.....	121
Figure 80 : Matrice de Confusion Retinal_MobileNet	121
Figure 81: Test et AccuracyMobileNet (Module2)	121
Figure 82: Représentations graphiques de précision et de perte MobileNet(Module2). ...	122
Figure 83: Matrice de Confusion MobileNet(Module2).....	122
Figure 84: représente phase de prédiction (Module2).	123
Figure 85 : Les deux(02) parties de Convolutional Neural Network.[62].....	6
Figure 86 : Les quatre types de couches pour un réseau de neurones Convolutif.[62].....	6

Figure 87 : Schéma du parcours de la fenêtre de filtre sur l'image .[62].....	7
Figure 88 : A Gauche : Un réseau standard sans drop-out. A droite : Un réseau avec un drop-out.[64].....	9
Figure 89: Illustration des différentes valeurs de fonction de coût obtenues par les différents algorithmes d'optimisation[68].....	13
Figure 90 : Diagramme de haut niveau de l'apprentissage par transfert à partir d'un modèle de CNN préformé.[13]	14

Liste des abréviations

RD : Rétinopathie Diabétique
DMLA : Dégénérescence Maculaire liée à L'âge
OCT : Tomographie par Cohérence Optique
FD : Fond d'Oeil
OCTA : OCT Angiographie
OMC : Œdème Maculaire Cystoïde
AMIR : Anomalies Microvasculaires Intrarétiniennes
MLA : Maculopathie Liée à L'âge
AV : Acuité Visuelle
IA : Intelligence Artificielle
ML : Machine Learning
DL : Deep Learning
SVM : Machine à Vecteurs de Support
GLM : Modèle Linéaire Généralisé
CART : Classification and Regression Tree
Adam : Adaptive Moment Estimation
CNN : Convolution Neural Network
RNN : Récurrent Neural Network
LSTM : Long Short-Term Memory
KNN: K plus Proche Voisin
RF : Forêts Aléatoires
DT : Arbres de décision
LR : Régression Logistique
ReLU : Rectified Linear Units
FC : fully-connected

Sommaire

Introduction générale	25
Contexte.....	25
Organisation du mémoire.....	27
Chapitre I Contexte Médical.....	28
Introduction.....	29
I. Anatomie de l'œil	30
I.1 La tunique externe	30
I.2 La tunique moyenne	30
I.3La tunique interne.....	31
I.3.1 La rétine	31
I.3.2 Le nerf optique.....	31
I.3.3 La macula.....	32
I.3.4 Les photorécepteurs	32

I.3.5 Les bâtonnets	32
II. Pathologies rétiniennes.....	33
II.1 Rétinopathie diabétique (RD)	33
II.1.1 Définition	33
II.1.2 Diagnostique	34
II.2 Dégénérescence Maculaire lie à l'âge.....	35
II.2.1 Définition	35
II.2.2 Diagnostique	36
II.3 Glaucome	36
II.3.1 Définition	36
II.3.2 Diagnostique	37
III. Les techniques d'analyse d'imagerie médicale	38
III.1 L'examen du fond d'œil	38
III.2 Techniques d'acquisition des images du fond d'œil.....	38
III.2.1 Rétinographie	38

III.2.2	Angiographie à la fluorescence.....	38
III.3	La tomographie de cohérence optique (OCT) :.....	38
III.3.1	Définition	38
III.3.2	Principes de L'OCT	39
III.3.1	L'OCT-Angiographie	40
	Conclusion.....	41
	Chapitre II: Deep Learning.....	42
	Introduction.....	43
I.	Intelligence Artificielle (IA).....	43
II.	Apprentissage Automatique (Machine Learning).....	44
II.1	Types d'apprentissage automatique	44
II.1.1	Apprentissage automatique supervisée	44
II.1.2	Apprentissage Automatique Non-Supervisé.....	50
II.1.3	Apprentissage par Renforcement	51
II.1.4	Sur et sous apprentissage	51

III. Apprentissage profond (Deep Learning)	52
III.1 Réseaux de Neurones Artificiels.....	53
III.1.1 Le neurone	53
III.1.2 Perceptron simple d'un neurone	54
III.1.3 Perceptron multicouches (PMC).....	55
III.1.4 La descente de gradient.....	56
III.1.5 Les fonctions d'activation.....	57
IV. Les architectures du Deep Learning	58
IV.1 Les réseaux entièrement connectés	58
IV.2 Les réseaux de neurones convolutifs.....	59
IV.3 Réseau de neurones récurrent.....	60
IV.4 Long Short-Term Memory (LSTM).....	61
IV.5 Modèle génératif	61
Conclusion.....	62

Chapitre III : Étude des Travaux utilisant les techniques de Machine Learning pour le diagnostic des maladies oculaires

Introduction.....	64
I. Critères d'évaluation du modèle d'apprentissage.....	64
I.1 Ensemble de données d'apprentissage	65
I.2 Choix de l'algorithme d'apprentissage.....	65
I.3 Les mesures de performance	66
II. Applications de L'Intelligence Artificielle IA en ophtalmologie	68
II.1 Machine Learning pour le diagnostique de(RD, DMLA ,Glaucome)	68
II.1.1 Travaux machine Learning obtenant meilleurs résultats avec la méthode support vector machine (SVM).....	70
II.1.2 Travaux Machine Learning obtenant meilleurs résultats avec la méthode forêt aléatoire (RF).....	71
II.1.3 Travaux Machine Learning obtenant meilleurs résultats avec la méthode <i>k</i> voisin le plus proche (KNN) et GMM.....	72
II.1.4 Discussions et conclusions.....	72
II.2 Applications de diagnostique avec Deep Learning.....	73

II.2.1 Travaux ont utilisé le model VGG pour le diagnostic des maladies ophtalmique.....	75
II.2.2 Travaux ont utilisé leur propre modèle CNN pour le diagnostic des maladies ophtalmique.....	76
II.2.3 Travaux ont utilisé DeepSeeNet et Inception V3 pour le diagnostique des maladies ophtalmique:.....	77
II.2.4 Discussions et conclusions :.....	77
II.3 Applications de diagnostic combiner DeepLearning et machine Learning	77
II.3.1 Travaux combiner un modèle DeepLearning et une Méthode de Machine Learning dans le diagnostic des maladies	78
II.3.2 Discussion et conclusions	79
III. Défis potentiels	79
Conclusion.....	82
Chapitre IV Solution proposée.....	83
Introduction.....	84
I. Architecture du système	84

II. Data set.....	86
II.1 Ensemble d'entraînement (Training Set)	86
II.2 Ensemble de test (Test set)	87
II.3 Ensemble de validation (Validation set)	87
II.4 Préparation des données	88
III. Phase d'entraînement du modèle	88
III.1 Modèles CNN.....	89
III.1.1 CNN à trois couches :	90
III.1.2 Model à cinq couches :	91
III.1.3 Model à sept couches :	93
III.1.4 Apprentissage par transfert VGG16.....	95
III.1.5 Apprentissage de transfert ResNet50	97
III.1.6 Apprentissage de transfert DenseNet121	98
III.1.7 Apprentissage de transfert MobileNet	99
IV. Phase de test.....	100

Conclusion.....	100
Chapitre V.. Implémentation et Test.....	101
Introduction	102
I. Environnement de développement	102
I.1 Le hardware	102
I.2 Le software	103
I.2.1 Python	103
I.2.2 Bibliothèques utilisées	103
II. Aperçu du système	104
II.1 Module 1	104
II.1.1 Jeux de données	104
II.1.2 Implémentation et Mise en œuvre.....	106
II.1.3 Comparaison des performances des modèles	115
II.1.4 Phase de test.....	116
II.2 Module 2	117

II.2.1 Jeux de données	117
II.2.2 Implémentation et Mise en œuvre.....	120
Conclusion.....	124
Conclusion générale et Perspectives	125
Annexe.....	1
III. Fonction d'activations	2
III.1 Fonction d'activation linéaire	2
III.2 Fonction d'activation non linéaire	2
IV. Réseaux de Neurones Convolutifs(CNN).....	5
IV.1 Hyper-paramètres	8
IV.1.1 Hyper-paramètresliés à la structure du réseau	8
IV.1.2 Hyper-paramètres liés à l'algorithme d'entraînement	10
IV.2 Apprentissage par transfert.....	13

Introduction générale

Contexte

La rétinopathie diabétique (RD), la dégénérescence maculaire liée à l'âge (DMLA) et le glaucome sont les trois causes les plus courantes de cécité irréversible à l'échelle mondiale. L'imagerie ophtalmique fournit un moyen de diagnostiquer et de détecter objectivement la progression d'un certain nombre de pathologies, notamment la RD, la DMLA, le glaucome et d'autres troubles ophtalmiques.

Il existe trois méthodes d'imagerie utilisées comme moyen de diagnostic dans la pratique ophtalmique: la photographie numérique du fond d'œil, la tomographie par cohérence optique (OCT) et l'Angiographie rétinienne. Il convient de noter que l'OCT est devenue la modalité d'imagerie la plus utilisée en ophtalmologie dans les pays développés.

La découverte précoce et le diagnostic précis de ces maladies joue un rôle crucial dans la Prévention de la perte de la vision des patients atteints de ces pathologies. Les méthodes conventionnelles de diagnostic ophtalmique dépendent profondément de l'expérience et les connaissances professionnelles des médecins, ce qui peut conduire à un taux élevé d'erreurs de diagnostic et le gaspillage d'une grande quantité de l'énergie et de données médicales.

Par conséquent, un outil d'aide au diagnostic ne sera que bénéfique pour les ophtalmologistes. En effet, l'ophtalmologie est un domaine où l'imagerie et les données mesurables sont nombreuses, ce qui en fait un domaine idéal pour l'application de l'IA et du Deep Learning (DL).

Problématique et objectifs

L'OCT (Tomographie par Cohérence Optique), souvent appelé "scanner de la rétine ou des yeux" par abus de langage, est au un examen d'imagerie médicale permettant la visualisation du fond d'œil. L'OCT permet donc de détecter très précocement des pathologies de l'œil, afin de permettre la mise en place de traitements appropriés et d'éviter l'apparition de Dommages irréversibles. Récemment, les ophtalmologistes peuvent réaliser un nouvel

examen, grâce à l'OCT : c'est l'OCT angiographie ou OCTA. Il s'agit d'un programme spécial qui permet de bien visualiser les vaisseaux.

Ces différentes techniques d'imagerie médicale utilisées en ophtalmologie produisent des images. Les ophtalmologistes interprètent ces images afin de poser un diagnostic. L'expérience et la compétence de l'ophtalmologiste ainsi que la subjectivité humaine sont la source de beaucoup d'erreurs de diagnostic.

Plusieurs travaux [1][2][3] ont montré la performance et la robustesse des modèles du Deep Learning quand ils sont appliqués aux données de type image. C'est dans ce contexte que s'intègre notre travail qui consiste à proposer une méthode qui se base sur une ou plusieurs architectures de Deep Learning afin de bien interpréter les images de l'OCT et l'ANGIO, et par conséquent aider l'ophtalmologiste à faire son diagnostic d'une façon plus précise.

Organisation du mémoire

Le mémoire se repartit en cinq chapitres :

Chapitre 1 : « Contexte Médical »

Ce chapitre contient une présentation sur l'anatomie de l'œil en se basant sur sa composante la plus importante : la rétine et ses éléments principaux. Et une présentation de quelques pathologies rétiniennes, définition de l'examen du fond d'œil, l'Angiographie et l'OCT..

Chapitre 2 : « Deep Learning »

Dans ce chapitre nous allons présenter c'est quoi le Deep Learning ainsi que les différentes architectures utilisées le domaine utilisé.

Chapitre 3 : «Machine pour le diagnostic des maladies oculaires : RD, DMLA, et Glaucome»

Dans ce chapitre nous présenterons quelques revues et techniques antécédentes qui se rapproches le plus de notre objectif.

Chapitre 4 : « Solution proposée »

Ce chapitre présente l'approche proposé pour le diagnostic des maladies ophtalmique cité auparavant, les modèles utilisés, présentation des différentes phases du système et les paramètres lié à l'entraînement du modèle, terminant par la phase du test.

Chapitre 5 : « Implémentation et tests » Ce chapitre, sera réservé pour exposer notre solution proposée et ces résultats.

Chapitre I

Contexte Médical

Introduction

La Dégénérescence Maculaire Liée à l'Âge (DMLA), le glaucome et la rétinopathie diabétique sont les trois premières maladies oculaires responsables de la cécité et la perte de la vision dans le monde.

Dans le diagnostic des pathologies ophtalmologiques, les nouvelles machines d'imagerie révèlent un grand intérêt thérapeutique elles permettent notamment d'évaluer l'efficacité de certains médicaments tels que les Anti VEGF, utilisés dans la prise en charge des maladies de la rétine, et aident le praticien à prendre la bonne décision concernant le traitement.

Les ophtalmologistes se base sur des techniques d'imagerie médicale, l'Angiographie et l'OCT en grande partie pour le diagnostic et le suivi de l'état des malades pendant la période de traitement afin de surveiller l'évolution de la maladie et la réponse des patients aux différents protocoles de traitement.

Dans ce chapitre Nous allons présenter l'anatomie de l'œil en se basant sur sa composante la plus importante : la rétine et ses éléments principaux. En plus, nous allons présenter quelques pathologies rétiniennes. Nous allons terminer par la définition de l'examen du fond d'œil, l'Angiographie et l'OCT.

I. Anatomie de l'œil

L'œil est l'organe principal du système visuel, qui capte les images et les transforme en signal électrique vers le nerf optique. Ce signal est ensuite « traduit » par le cerveau, au niveau du cortex visuel, qui nous renvoie l'image traitée et permet ainsi l'interprétation de notre environnement.[4]

L'œil peut se décomposer en trois principales tuniques (Figure 1) :

I.1 La tunique externe

Regroupe la cornée et la sclère :

La sclérotique (sclère ou sclérotique) La sclère est une tunique résistante, blanchâtre et inextensible constituant l'enveloppe externe du globe oculaire. Elle est formée d'un tissu conjonctif peu innervé et peu vascularisé sur lequel s'insèrent les 6 muscles oculomoteurs.[5]

La sclère est perforée d'un orifice : la papille par où sortent les fibres optiques du nerf optique, la veine centrale de la rétine et par où entre l'artère centrale de la rétine, Le rôle de la sclérotique est de maintenir le volume, les formes et le tonus du globe et de supporter la traction des muscles oculomoteurs.[6]

La cornée c'est une structure transparente qui n'est pas vascularisée, afin d'assurer la transmission de la lumière. Elle est formée de cinq couches : épithélium, membrane de Bowman, stroma, membrane de D'Écémment et endothélium. Les cellules épithéliales sont renouvelées tous les sept jours.

La cornée est séparée de la sclère par le limbe, structure richement innervée et vascularisée qui adhère à la conjonctive. La conjonctive est une fine membrane qui recouvre la face interne de la paupière et une partie de la sclère.[7]

I.2 La tunique moyenne

Aussi appelée uvée, se compose de l'iris, des corps ciliaires et de la choroïde :

L'iris, partie qui donne sa couleur à l'œil, est formé de fibres pigmentées qui entourent la pupille. la pupille s'adapte à la lumière en se contractant et en se dilatant par l'intermédiaire du sphincter de l'iris[7]

Les corps ciliaires créent la jonction entre l'iris et la choroïde. des corps ciliaires sont également en contact avec la rétine au niveau de la zone appelée "oraserrata". Les corps ciliaires assurent la sécrétion d'une substance liquide, l'humeur aqueuse.[7]

La choroïde : c'est une structure qui se trouve entre la rétine et la sclère .C'est un tissu vascularisé qui nourrit les corps ciliaires, l'iris et la partie externe de la rétine.[7]

I.3 La tunique interne

Correspond au tissu nerveux, la rétine.

Ces différentes tuniques, enferment trois milieux transparents formants une lentille convergente. [7]

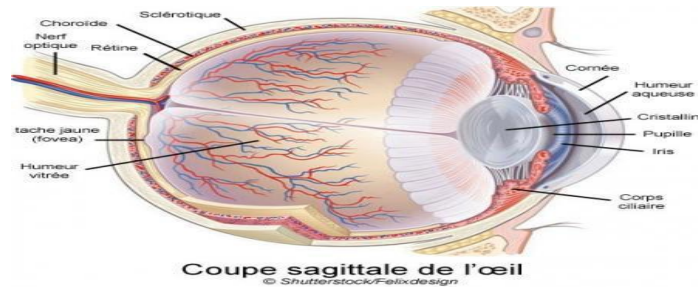


Figure 1: Anatomie de l'œil.[8]

I.3.1 La rétine

La rétine est la première étape importante du traitement des stimuli lumineux, grâce à plusieurs couches de neurones. Cette étape est ensuite suivie d'un message électrique qui va traverser le nerf optique pour finalement atteindre les voies optiques jusqu'au lobe occipital du cerveau, où ils sont perçus comme une image. La rétine tapisse la surface intérieure du fond de l'œil et se prolonge pour former le nerf optique. Au fond de l'œil à travers la pupille, nous pouvons observer les principaux éléments de la rétine qui sont le disque du nerf optique (ou papille optique), les vaisseaux sanguins de la rétine et la macula (Figure 2). [7]Les principaux éléments constituant la rétine sont :

La neurorétine, qui comprend l'ensemble des cellules nerveuses.

Les cellules gliales.

Le système vasculaire. [5]

I.3.2 Le nerf optique

Est le second nerf crânien, débute au niveau de la papille optique [5].

Le nerf optique transmet les informations reçues par les yeux vers le cerveau au niveau du cortex visuel, C'est lui qui permet au cerveau d'enregistrer, d'interpréter et de traduire les images [4].

I.3.3 La macula

Est la zone dont dépend la vision centrale. Le reste de la rétine sert à la vision périphérique. Au centre de la macula se trouve la fovéa, qui n'est pas vascularisée. [7]

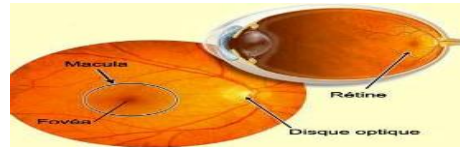


Figure 2: Fond d'œil.[5]

I.3.4 Les photorécepteurs

Sont les cellules les plus profondes, par rapport à l'arrivée de la lumière. Ils sont subdivisés en deux types cellulaires qui sont: **les bâtonnets** et **les cônes**. [7](Figure 3)

I.3.5 Les bâtonnets

Permettent surtout une vision de nuit ou lorsque l'environnement est sombre, alors que les cônes permettent d'avoir une vision nette lorsque la lumière est vive, de voir les couleurs et la définition des images. Chez l'homme, la rétine centrale comprend la macula, une zone pratiquement vasculaire qui contient en son centre la fovéa, une petite dépression qui est responsable de la vision des détails. Cette région spécialisée, uniquement peuplée de cônes, est l'endroit de la rétine qui permet la meilleure acuité visuelle. [7]

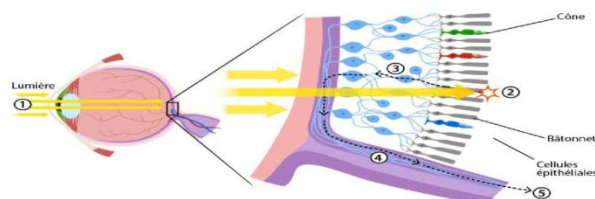


Figure 3: Schéma de la rétine. [6]

Ces différentes tuniques, enferment trois milieux transparents formants une lentille convergente. [4]

- **L'humeur aqueuse :**

C'est un liquide transparent produit par les corps ciliaires. La pression intraoculaire dépend entre autre de cette production. L'humeur aqueuse circule de la chambre postérieure vers la chambre antérieure, d'où elle s'écoule vers le canal de Schlem par le réseau tabulaire. [8]

- **Le cristallin**

C'est une lentille biconvexe incolore, transparente et non vascularisée. Sa structure lui permet de dévier les rayons lumineux sur la rétine. Sa partie antérieure est en contact avec l'humeur aqueuse et la partie postérieure avec l'humeur vitrée.[8]

- **L'humeur vitrée (ou corps vitré)**

C'est une substance gélatineuse transparente composée principalement d'eau, qui occupe les deux tiers du volume de l'œil. La voie visuelle comprend la rétine, le nerf optique, le chiasma optique, les tractus optique. Les corps géniculés latéraux, les radiations optiques et l'aire visuelle primaire du cerveau. [7]

II. Pathologies rétiniennes

Les maladies de la rétine constituent une cause majeure de malvoyance, leurs origines sont variées telles que: le vieillissement, les complications du diabète, etc. nous présentons dans cette section quelques pathologies qui touchent la rétine et qui peuvent mener à la cécité s'ils ne sont pas diagnostiqués tôt.

II.1 Rétinopathie diabétique (RD)

II.1.1 Définition

La RD est une maladie qui est la principale cause de malvoyance et de cécité chez les patients diabétiques .[9]Le diabète est un agent robuste de cette pathologie, il se définit par une augmentation de sucre dans le sang résultant de facteurs génétiques et environnementaux .[10]

On estime qu'environ 30% des diabétiques sont porteurs d'une rétinopathie. [11]

La dilatation et l'occlusion des capillaires rétiniens sont les premières lésions cliniquement décelables de la rétinopathie diabétique, aboutissant à deux phénomènes intriqués : hyperperméabilité capillaire menant à l'œdème rétinien, et parallèlement occlusion capillaire menant à l'ischémie rétinienne.[11]

L'occlusion étendue des capillaires rétiniens crée une ischémie rétinienne. Lorsque celle-ci est étendue, il se produit une prolifération réactionnelle de néovaisseaux par production de facteurs de croissance, dont le VEGF : c'est la rétinopathie diabétique proliférante. Des complications peuvent alors survenir :

- Hémorragie intra-vitréenne par saignement des néovaisseaux pré-rétiniens ou pré-papillaires,

- Décollement de la rétine dû à la traction exercés sur la rétine par du tissu fibreux de soutien des néovaisseaux (prolifération fibro-vasculaire) : on parle de décollement de rétine par traction.
- voire une prolifération de néovaisseaux sur l'iris (néo vascularisationirienne)et dans l'angle iridocornéen, pouvant provoquer comme au cours des occlusions de la veine centrale de la rétine un glaucome neovasculaire par blocage de l'écoulement de l'humeur aqueuse.
- Au niveau de la macula L'accumulation de liquide dans le compartiment extracellulaire de la rétine. [11]

II.1.2 Diagnostique

Le diagnostic de la RD repose sur l'examen bio microscopique du fond d'œil après dilatation pupillaire, complété par des photographies du fond d'œil. Cet examen permet d'identifier les différents signes de la RD [11] ,On site:

- Les **micro anévrismes rétiniens** sont les premiers signes ophtalmo scopiques de la RD. Ils apparaissent sous forme de lésions punctiformes rouges de petite taille. Ils prédominent au pôle postérieur. Les micros anévrismes peuvent se thrombose et disparaître spontanément. Mais l'augmentation du nombre des micro anévrismes est un bon indice de progression de la RD.

- Les hémorragies **rétiniennes punctiformes** peuvent être associées aux micros anévrismes.

D'autres signes sont évocateurs d'ischémie rétinienne sévère compliquant la RD :

- Des **néovaisseaux pré rétiniens et pré papillaires** caractérisent la RD proliférante. Ils prolifèrent à la limite postérieure des territoires ischémiques, ou sur la papille du nerf optique lorsque la surface de rétine non perfusée est très étendue,

D'autres signes sont évocateurs de l'hyperperméabilité capillaire au niveau de la macula :

- un œdème maculaire cystoïde (OMC) qui se traduit par un épaissement micro kystique de la rétine maculaire.
- des exsudats lipidiques qui sont des accumulations de lipoprotéines dans l'épaisseur de la rétine oedématiée. Ils forment de dépôts jaunes souvent disposés en couronne autour des anomalies microvasculiaires dont ils proviennent (exsudats

circinés). [11]



Figure 4: Rétinopathie diabétique minime.[11]

Présence de micro anévrysmes et hémorragies punctiformes au pôle postérieur : à l'examen du fond d'œil les micros anévrysmes apparaissent comme de petites lésions punctiformes, rouges, à la limite de la visibilité. [11]

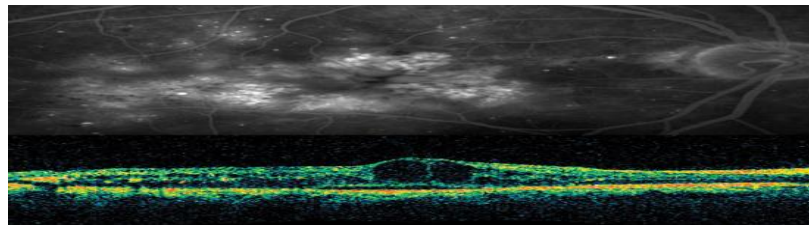


Figure 5 : Œdème maculaire cystoïde sévère.[11]

En angiographie à la fluorescéine, accumulation de colorant dans des logettes cystoïdes étendues à toute la surface de la macula. L'OCT montre l'épaississement micro kystique rétinien maculaire. [11]

II.2 Dégénérescence Maculaire lie à l'âge

II.2.1 Définition

Le terme de dégénérescence maculaire liée à l'âge (DMLA) correspond a un ensemble de lésions dégénératives secondaires a des facteurs de susceptibilité génétique et/ou environnementaux, qui altèrent l'aire centrale de la rétine chez les personnes âgées de plus de 50 ans. [11]

Le terme de maculopathie liée à l'âge (MLA) est la forme précoce de la maladie. Elle précède la DMLA. Elle est caractérisée par la présence de drusen[11]. La DMLA proprement dite peut prendre deux formes :

La forme atrophique, dite aussi "forme sèche", caractérisée par des altérations de l'épithélium pigmentaire et un amincissement de la macula consécutif a l'évolution des drusen ; son évolution est lente sur des années.

La forme exsudative, appelée "forme humide" ou "forme néovasculaire", caractérisée par le développement de néovaisseaux choroïdiens sous la macula ; l'évolution de cette forme peut être très rapide, faisant perdre la vision centrale (acuité visuelle (AV) < 1/10) en quelques semaines ou mois. [11]La DMLA est la première cause de malvoyance après 50 ans dans les pays industrialisés

La prévalence globale de la maladie est de 8% après 50 ans, Cette prévalence globale augmente avec l'âge : de 1% avant 55 ans, elle passe progressivement à 25% après 75 ans [11].

II.2.2 Diagnostique

La DMLA est suspectée devant une baisse progressive de l'acuité visuelle de loin et de près, correspondant à l'aggravation lente des lésions (drusen, forme atrophique), Une baisse brutale de l'acuité visuelle associée a des métamorphoses (sensation de déformation des objets : le patient décrit souvent la vision ondulée des lignes droites) qui sont liées le plus souvent à l'apparition de neovaisseaux choroïdiens, responsables d'une exsudation et de L'accumulation de liquide au niveau de la macula, Un scotome central, correspondant aux stades très évolués des formes atrophiques et exsudatives. L'examen biomicroscopie du fond d'œil pratique à la lampe a fente, après dilatation pupillaire, avec un verre de contact recherche:

- La présence de drusen (MLA) ;

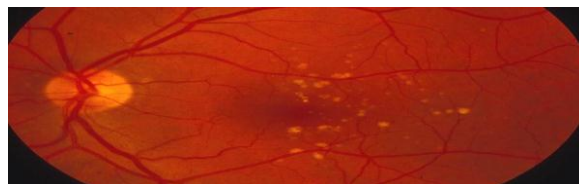


Figure 6 :Forme précoce -drusens maculaires-.[12]

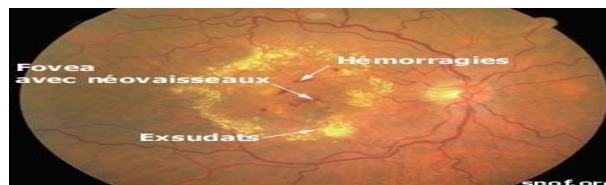


Figure 7: Forme exsudative de DMLA.[12]

II.3 Glaucome

II.3.1 Définition

Le glaucome est une neuropathie optique dégénérative asymptomatique, qui associe des changements morphologiques au niveau de la tête du nerf optique et une altération du

champ visuel. En effet, une excavation de la papille et une modification de la couche des fibres nerveuses sont observées et entraînent un déficit au niveau du champ visuel.

[13] Il existe deux types principaux de glaucome :

Le glaucome à angle ouvert :

Qui est le plus fréquent. Il apparaît lorsque l'humeur aqueuse (liquide contenu dans le globe oculaire) ne circule plus correctement à travers la membrane de l'œil, ce qui entraîne une augmentation de la pression intraoculaire et une détérioration du nerf optique.

Le glaucome à angle fermé :

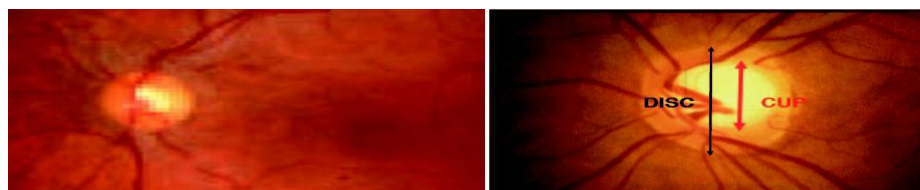
Qui survient lorsque le drainage des fluides n'est plus possible car la zone entre l'iris et la cornée se referme complètement.

Le glaucome à angle fermé peut être :

- Un glaucome primaire aigu à angle fermé, pour lequel l'obstruction est rapide et survient en quelques heures.
- Un glaucome chronique par fermeture de l'angle, pour lequel l'obstruction peut survenir en quelques semaines ou années. [19]

II.3.2 Diagnostique

Un angle irido-cornéen ouvert en gonioscopie, une papille excavée, c'est-à-dire creusée en son centre par raréfaction progressive des fibres nerveuses qui forment le nerf optique : cette excavation pathologique s'apprécie en mesurant le «rapport cup-disc» (rapport entre la surface de l'excavation et la surface de la papille). [11]



a) Papille normale b) Excavation papillaire

Figure 8 : glaucome[11]

III. Les techniques d'analyse d'imagerie médicale

III.1 L'examen du fond d'œil

Le fond d'œil est un examen simple qui permet à l'ophtalmologue d'observer la rétine avec ses vaisseaux sanguins, la tête du nerf optique (la papille) et la macula. Il existe plusieurs manières d'observer le fond d'œil du patient. Parmi tous les appareils qui existent nous allons citer deux appareils importants qui permettent aux médecins d'observer la rétine et d'offrir au même temps des photographies du fond d'œil.[14]

III.2 Techniques d'acquisition des images du fond d'œil

III.2.1 Rétinographie

Cette technique consiste à l'utilisation d'une rétinographie qui permet de prendre plusieurs photographies en couleur du fond d'œil[14]. Il peut être réalisé avec la dilatation de la pupille (rétinographie mydriatique) ou sans dilatation (rétinographie non mydriatique)[15]

C'est un examen indolore et les images peuvent être archivées ce qui est un élément important pour le bon suivi du patient et la surveillance de sa maladie [14](figure 9.a.)

III.2.2 Angiographie à la fluorescence

Cet examen consiste en la prise de photographies du fond d'œil après injection intraveineuse d'un colorant fluorescent (fluorescéine, vert d'indocyanine[6]) [15]. Il permet une étude détaillée des vaisseaux sanguins pour établir le diagnostic de certaines maladies ou guider un traitement, par exemple, lorsqu'une photo-coagulation au laser est envisagée[6]. Les indications essentielles sont la rétinopathie diabétique et les dégénérescences musculaires[6](Figure 9 b)



(a): Rétinographie; (b): Angiographie à la fluorescence.

Figure 9: Techniques d'acquisition des images du fond d'œil.[6]

III.3 La tomographie de cohérence optique (OCT) :

III.3.1 Définition

La Tomographie par Cohérence Optique (OCT) est un examen permettant d'obtenir des coupes anatomiques fines par l'application d'un balayage d'une lumière laser sur la structure à examiner. Il permet alors d'avoir des coupes très précises des différentes

structures de l'œil, de la cornée à la rétine et au nerf optique. L'examen n'est pas douloureux car sans contact avec l'œil, et il n'est pas éblouissant. Le patient devra uniquement suivre une cible de fixation.

L'examen OCT permet d'analyser l'angle entre la cornée et le cristallin, d'évaluer l'épaisseur des fibres nerveuses, et de mesurer la taille du nerf optique.

La figure (Figure) qui suit présente l'appareil d'OCT. Tel que :

Appareil d'OCT, B. Examen automatisé de la tête du nerf optique, La couleur vert indique les cellules saines, le jaune affectées, le rouge mortes.[16]

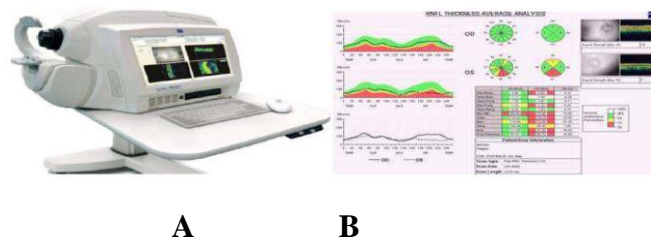


Figure 10: L'appareil d'OCT et résultat de l'examen.[6]

III.3.2 Principes de L'OCT

Sa principale application est l'étude des affections musculaires :

- confirmation du diagnostic d'une membrane pré maculaire ou d'un trou maculaire.
- quantification d'un œdème rétinien maculaire, et évolution sous traitement par la mesure de l'épaisseur maculaire.
- Visualisation et localisation de néovaisseaux choroïdiens dans la DMLA.[11]
- Elle est également utilisée dans le dépistage et le suivi du glaucome chronique car elle permet de mettre en évidence la perte de fibres optiques, premier signe du glaucome.[11]

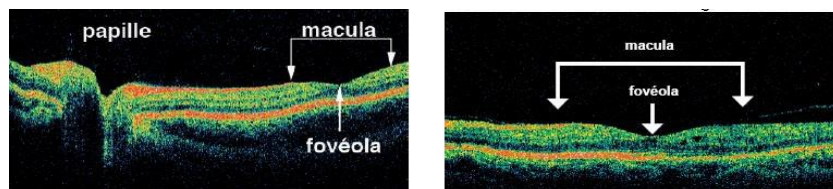


Figure 11 :OCT normal.[11]

A gauche, OCT passant par la papille et la macula,A droite, OCT maculaire

III.3.1 L'OCT-Angiographie

III.3.1.1 Définition

Angiographie-OCT (Angio-OCT) est une nouvelle technique performante et novatrice, offrant une visualisation quasi instantanée des flux vasculaires chorio-rétiniens. Examen non invasif, l'Angio-OCT repose sur l'acquisition d'une imagerie OCT "en face", c'est-à-dire avec des coupes frontales, à la détection des mouvements du flux sanguin. La vascularisation normale ou pathologique peut ainsi être analysée en trois dimensions au sein de la fenêtre d'acquisition. Les éléments figurés du sang peuvent être observés en réalisant des coupes répétées et consécutives d'une même région, puis en calculant la "dé-corrélation" de l'amplitude du signal. L'Angio-OCT ne nécessite pas d'injection intraveineuse de produit de contraste évitant ainsi tout risque allergique potentiel.

Le temps d'acquisition est très rapide (quelques secondes) ainsi que le diagnostic pour l'ophtalmologiste et le patient. [15] Comme en OCT "en face", il est possible de choisir des plans de référence prédéfinis pour étudier la zone d'intérêt à différentes profondeurs. [15] Les principaux plans d'analyse sont (Figure 12):

- le plexus vasculaire superficiel.
- le plexus vasculaire profond.
- la rétine externe et le plan choroïdien avec la choriocapillaire bien visible. [15]

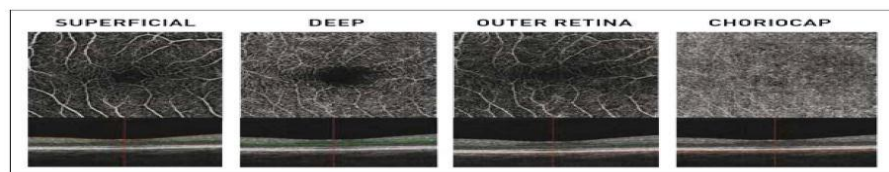


Figure 12 : Les différents plans d'analyse OCT. [15]

Conclusion

Nous avons présenté dans ce chapitre l'anatomie de l'œil, les pathologies rétinienne et les techniques d'analyse d'imagerie médicale, dans le chapitre suivant nous allons présenter quelques notions de base sur le Machine Learning (ML) et l'Apprentissage Automatique et définir l'apprentissage profond (Deep Learning), ses méthodes et ses architectures.

Deep Learning

Introduction

L'objectif de la recherche en intelligence artificielle (IA) est de doter un système informatique de capacités de réflexion similaires à celles des humains. Il y a donc un défi dans la compréhension du raisonnement humain, mais surtout dans la modélisation et dans la reproduction de celui-ci .

L'IA est devenue un sujet en vogue dans les médias et magazines scientifiques en raison des nombreuses réalisations, dont beaucoup sont le fruit des progrès accomplis dans le domaine de l'apprentissage automatique. De grandes entreprises dont Google, Facebook, IBM, Microsoft mais aussi des constructeurs automobiles à l'instar de Toyota, Volvo et Renault, sont très actifs dans la recherche en IA et prévoient d'y investir davantage encore dans le futur. Plusieurs scientifiques spécialisés dans l'IA dirigent désormais les laboratoires de recherche de ces grandes entreprises et de nombreuses autres. La recherche en IA a permis de réaliser d'importants progrès dans la dernière décennie, et ce dans différents secteurs. Les avancées les plus connues sont celles réalisées dans l'apprentissage automatique, grâce notamment au développement d'architectures d'apprentissage profond, des réseaux de neurones convolutifs multicouche dont l'apprentissage s'opère à partir de gros volumes de données sur des architectures de calcul intensif. Parmi les réalisations de l'apprentissage automatique, il convient de citer la résolution de jeux Atari (Bricks, Spaceinvaders, etc.) par Google DeepMind, utilisant les pixels images affichés à l'écran comme données d'entrée afin de décider quelle action adopter pour atteindre le plus haut score possible à la fin de la partie[17]

Le but de ce chapitre est d'introduire **l'intelligence artificiels (AI : Artificiel Intelligence)**et **L'Apprentissage Automatique (Machine Learning (ML) en langue anglaise)**. Une grande partie de ce chapitre sera consacrée à l'apprentissage profond (Deep Learning, DL) et ses différentes architectures.

I. Intelligence Artificielle (IA)

Bien que le terme Intelligence artificielle puisse sembler explicite, il existe de nombreuses définitions. L'intelligence computationnelle est l'étude de la conception d'agents intelligents [17].

Le terme « agent » peut désigner tout ce qui agit dans un environnement particulier, tout ce qui fait quelque chose, que ce soit les humains, les animaux, les machines... etc.

Un agent intelligent d'autre part est un système qui agit intelligemment. Comme l'expliquent les chercheurs : il est approprié pour son objectif, il est flexible face à des environnements changeants, des objectifs changeants, il apprend de l'expérience, et il fait des choix appropriés compte tenu des limitations et des calculs finis. [17]

II. Apprentissage Automatique (Machine Learning)

Le Machine Learning est un sous-domaine de l'IA.[17], qui donne à un système une capacité de compréhension grâce à ses algorithmes. Il est basé sur l'idée de faire apprendre des algorithmes à partir de données et de faire des prédictions avec ces données et par cela les ordinateurs apprennent à résoudre des tâches spécifiques, sans avoir besoin de les programmer.

"Un programme informatique est dit capable d'apprentissage à partir d'une expérience E dans le respect d'une classe de tâche T avec la mesure de performance P s'il accomplit la tâche T, mesurée par P, et améliorée par l'expérience E". [18]

II.1 Types d'apprentissage automatique

II.1.1 Apprentissage automatique supervisé

Cette approche a pour objectif la conception d'un modèle reliant des données d'apprentissage à un ensemble de valeurs de sortie (un comportement)[19]. Un expert est employé pour étiqueter correctement des exemples. L'apprenant doit alors trouver ou approximer la fonction qui permet d'affecter la bonne étiquette à ces exemples (figure 14).[19]

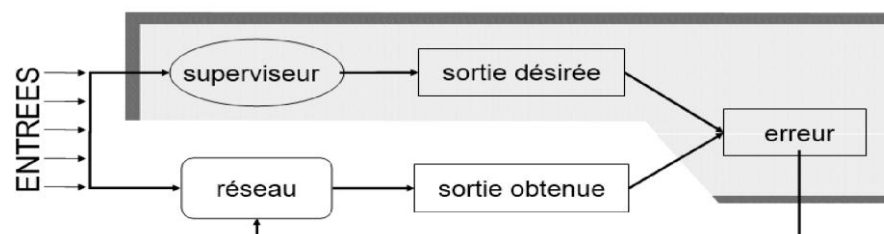


Figure 13: Schéma d'un modèle supervisé.[20]

II.1.1.1 La classification

Consiste à identifier les classes d'appartenance de nouveaux objets à partir d'exemples antérieurs connus, la variable à prédire peut donc prendre des valeurs discrètes appelées classes. Il existe des problèmes en multi-classes, c'est-à-dire que le changement de note peut-être représenté par plusieurs classes, par opposition à la classification binaire représentée par une variable de sortie à deux classes[21].(figure 15)

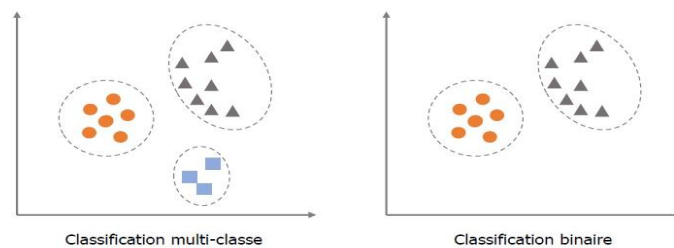


Figure 14 : Représentation des classifications.[21]

Les méthodes de classification supervisée peuvent être basées sur :

- **Classifieurs à mémoire** : des *hypothèses probabilistes* (cas du classifieur naïf bayésien),
- **Classifieurs Construisant des Hyperplans Séparateurs** : des *notions de proximité* (exemple, *k* plus proches voisins, **SVM**)
- **Classifieurs basés sur des Modèles des recherches dans des espaces d'hypothèses** (exemple, arbres de décisions). [22]

II.1.1.1 Naïf Bayésien

La classification naïve bayésienne repose sur l'hypothèse que les **attributs** sont **fortement** (ou **naïvement**) **indépendants**. Elle est basée sur le *théorème de Bayes* qui ne s'applique que sous cette hypothèse.

Théorème de Bayes est donné par :

$$P(X \setminus Y) = \frac{P(Y \setminus X)P(X)}{P(Y)} \dots \dots \dots \mathbf{1}$$

avec $P(X \setminus Y)$ est la probabilité conditionnelle d'un événement X sachant qu'un autre événement Y de probabilité non nulle s'est réalisé.[22]

Cependant, cette probabilité pourrait être beaucoup trop compliquée à estimer, si l'on considère le nombre de descriptions possibles. [22] C'est alors que l'on utilise l'hypothèse d'indépendance, qui nous permet de décomposer la probabilité conditionnelle en un produit de probabilités conditionnelles.

Le classifieur devient alors :

$$\text{Classe } (x) = \arg \max \prod_{i=1}^n P(X_i | H) * P(H) \text{-----2}$$

II.1.1.2 Machine à Vecteurs de Support (SVM)

Les machines à vecteurs de support ou séparateurs à vastes marges (notées **SVM** pour *Support Vector Machines*). Elles s'appliquent aussi bien à des **problèmes linéairement séparables** que **non séparables** [22].

L'idée principale des SVM est de reconsidérer le problème dans un espace infini. Dans ce nouvel espace, il est alors probable qu'il existe un hyperplan séparateur linéaire. Si c'est le cas, les SVM cherchent parmi l'infinité des hyperplans séparateurs celui qui maximise la marge entre les classes (Figure 17)[22].

On applique au vecteur d'entrée x une transformation non-linéaire Φ , afin de décrire les données dans un autre espace. L'espace d'arrivée $\Phi(x)$ est appelé **espace de redescription**. On cherche alors l'**hyperplan séparateur optimal**, d'équation dans l'espace de redescription [22].

$$h(x) = \alpha \Phi(x) + \beta \text{-----3}$$

On cherche *les équations des hyperplans parallèles* qui passent par **les vecteurs supports**, c'est-à-dire les attributs les plus proches de la frontière interclasse. On en déduit l'équation de l'hyperplan optimal, équidistant de ces hyperplans.

En pratique, on ne connaît pas la transformation Φ , on construit donc plutôt directement la fonction h , appelée *fonction noyau*. Le *théorème de Mercer* explicite les conditions que h doit satisfaire pour être une fonction noyau : elle doit être *symétrique* et *semi-définie positive*. [22]

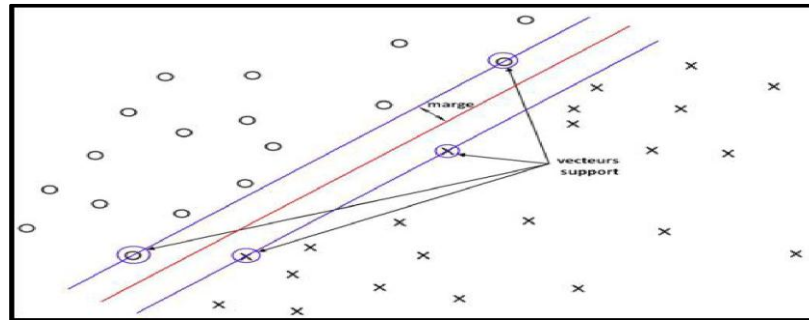


Figure 15 : Principe du Séparateur à Vaste Marge (SVM).[22]

II.1.1.3 La régression

Elle est utilisée lorsqu'il s'agit de prédire une variable continue, qui peut donc prendre n'importe quelle valeur. Les classes représentant le changement de note ont donc été considérées comme continues et l'algorithme de régression a permis d'obtenir des valeurs décimales que nous avons finalement arrondies pour obtenir un vecteur de classes prédites.

II.1.1.4 Les méthodes de régression

- **Modèle linéaire généralisé (GLM)**

Afin d'établir une comparaison avec les modèles statistiques classiques, un modèle linéaire généralisé est implémenté. [21]

Nous rappelons que les modèles linéaires généralisés sont une extension de la régression linéaire ordinaire dans laquelle la variable réponse y peut être vue comme une réalisation d'une distribution particulière de la famille exponentielle. En régression linéaire classique, une variation constante d'un prédicteur entraîne une variation constante de la variable expliquée, peu importe le point de l'ensemble de définition considéré. La relation de dépendance entre la sortie et les régresseurs est donc exclusivement linéaire. [21] A contrario, les GLM permettent de modéliser une dépendance non linéaire dans le sens où c'est l'image de la variable réponse par une fonction arbitraire g (appelée fonction de lien) qui dépend linéairement des variables explicatives :

$$(E[Y_i|X_i]) = X_i \beta \text{-----}4$$

où désigne la transposée du vecteur d'observations X_i .

Un modèle GLM est donc composé de 3 éléments clés :

- une composante aléatoire Y à laquelle est associée une loi de probabilité $(y_i, \theta_i, \varphi, \omega_i)$, cette loi appartenant à la famille des exponentielles :

$$f(y_i, \theta_i, \varphi, \omega_i) = \exp[y_i \theta_i - b(\theta_i) a(\varphi) \omega_i + c(y_i, \varphi, \omega_i)] \text{ -----5}$$

- a est une fonction non nulle définie sur \mathbb{R} ;
- b est une fonction définie sur \mathbb{R} et deux fois dérivable ; c est une fonction définie sur \mathbb{R}^2 ;
- θ_i est un paramètre réel appelé paramètre canonique ou paramètre de la moyenne ;
- φ est un paramètre réel appelé paramètre de dispersion
- ω_i représente la pondération liée à une observation.
- φ est un paramètre réel appelé paramètre de dispersion
- ω_i représente la pondération liée à une observation.
- une composante déterministe η , combinaison linéaire des X_1, \dots, X_k ... X_k appelée prédicteur linéaire : $\eta = X^t \beta = \beta_0 + \beta_1 X_1 + \dots + \beta_k X_k$
- une fonction de lien g monotone et différentiable entre la composante aléatoire Y et la composante déterministe η , spécifiant comment l'espérance mathématique de Y : $[Y] = \mu$ est liée au prédicteur linéaire η : $g(\mu) = \eta$.

Il s'agit de calculer les estimateurs β par une méthode telle que celle du maximum de vraisemblance pour obtenir le GLM.[21]

❖ Les arbres de classification et de régression

Les méthodes dites de partitionnement récursif ou de segmentation ont été formalisées dans un cadre générique de sélection de modèle sous l'acronyme de CART (Classification and Regression Tree). L'acronyme CART correspond à deux situations bien distinctes selon la variable à expliquer, modéliser ou prévoir est discrète (classification) ou continue (régression). L'avantage de cet algorithme est son pouvoir explicatif relativement simple, puisque les prédictions obtenues sont présentées sous une forme graphique facile d'interprétation et efficace pour

l'aide à la décision. Elles sont basées sur une séquence récursive de règles de division.

La construction d'un arbre consiste à déterminer une séquence de nœuds.

L'algorithme considéré nécessite :

- la définition d'un critère permettant de sélectionner la meilleure division parmi toutes celles admissibles pour les différentes variables,
- une règle permettant de décider qu'un nœud est terminal et ne se divise plus : il devient ainsi une feuille,[21]
- L'affectation de chaque feuille à l'une des classes ou à une valeur de la variable à expliquer.

Pour de nouvelles observations, l'arbre de décision renverra :

- La classe majoritaire de la feuille à laquelle les observations appartiennent s'il s'agit d'un problème de classification,
- La moyenne des valeurs de la variable à prédire pour les observations situées dans la même feuille s'il s'agit d'un problème de régression.

Dans cet exemple fictif, les conditions « *Chiffred'affaire* > 60M ? » et « *Opinioninvestisseurs* > 0,5 ? » permettent à l'algorithme d'effectuer deux divisions successives et d'arriver au niveau des feuilles de l'arbre pour identifier les types de migration (changement de note positif =1, négatif=-1, ou pas de changement=0).

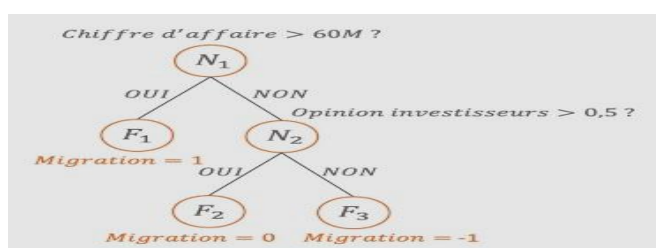


Figure 16: Arbre de classification déterminant la classe de migration des observations.[21]

Chaque nœud père engendre deux fils et à chaque pavé de l'espace ainsi découpé, est finalement associée une classe de migration.

Après chaque subdivision possible de l'arbre, l'algorithme regarde l'hétérogénéité de la partition, c'est à dire la partition qui « sépare » de manière la plus efficace la base, celle pour laquelle les données seront regroupées de la façon la plus homogène dans un nœud, d'après le critère de Gini. Cette hétérogénéité est calculée de la manière suivante en posant t un nœud de l'arbre :

$$G(t) = \sum (Y_e - \mu_t)^2$$

Où e est une observation présente dans le nœud t , Y_e correspond à l'ensemble des valeurs de migrations Y prises par les observations e dans t , et μ_t représente la moyenne empirique des Y_e .

II.1.2 Apprentissage Automatique Non-Supervisé

Aucun expert n'est disponible. Il vise à concevoir un modèle structurant l'information. La différence ici est que les comportements (ou catégories ou encore les classes) des données d'apprentissage ne sont pas connus, c'est ce que l'on cherche à trouver. [20](figure19)

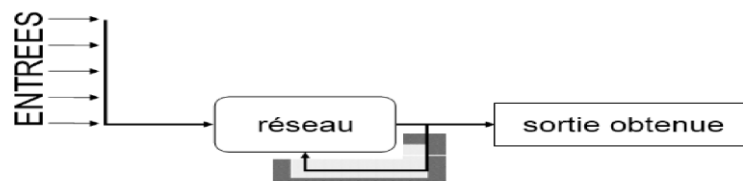


Figure 17 : Schéma d'un modèle non supervisé. [20]

II.1.2.1 Clustering

Entre dans la catégorie de l'apprentissage non-supervisé, où le but est plutôt de chercher à faire émerger de l'information qui n'était pas présente au départ. Les valeurs du changement de note sont prédéterminées, d'où le cadre de l'apprentissage supervisé.

Le principe est de regrouper des données par similarité sans avoir d'information au préalable. La différence avec la classification est que les classes ne sont pas identifiées en amont mais émergent à partir de l'exploration de la structure des données.[21]

II.1.3 Apprentissage par Renforcement

Les données en entrée sont les mêmes que pour l'apprentissage supervisé, cependant l'apprentissage est guidé par l'environnement sous la forme de récompenses ou de pénalités données en fonction de l'erreur commise lors de l'apprentissage.[23]

Il s'agira en général d'apprendre un comportement maximisant une mesure de performance, à moins que cet objectif soit seulement lié à la compréhension de l'environnement (**figure 18**)

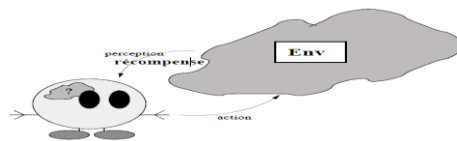


Figure 18 : Boucle (perception+récompense)/action en apprentissage par renforcement[22]

II.1.4 Sur et sous apprentissage

Il ne suffit pas de mettre en place un modèle pour obtenir des résultats. Une phase préalable d'optimisation et de paramétrage est primordiale pour éviter les erreurs dues au :

- **Sous-apprentissage** : qui est le fait de ne pas ajuster suffisamment le modèle aux données. Ainsi, le modèle apprend « mal » sur les données d'apprentissage et ne peut donc pas non plus modéliser le phénomène en question sur de nouvelles données.
- **Sur-apprentissage** : qui consiste au contraire à trop ajuster le modèle aux données. Le modèle apprend même le bruit présent dans les données d'apprentissage et cela a un impact négatif sur la prédiction face à de nouvelles données dont le bruit sera aléatoirement différent. La capacité de généralisation est réduite. [21]

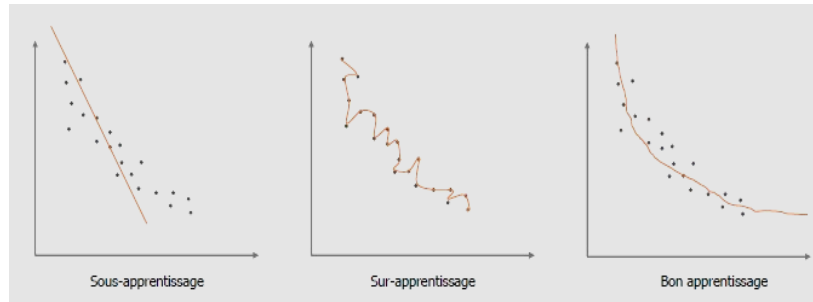


Figure 19: Différentes entre apprentissages mal estimés et bon apprentissage. [21]

Il est important de trouver un compromis « biais-variance » du modèle et de l'optimiser. Le biais d'un modèle correspond à la différence asymptotique entre la moyenne des prédictions et la moyenne réelle des données d'apprentissage. La variance du modèle quant à elle correspond au bruit présent dans le jeu de données qui modifie radicalement la prédiction d'un jeu de données à l'autre. Le sous-apprentissage contient un biais trop important tandis qu'une variance trop forte est le signe d'un sur-apprentissage. Toutes les méthodes algorithmiques sont concernées par ce compromis. Plus un modèle est complexe, plus il intègre de paramètres, et plus il est flexible donc capable de s'ajuster aux données engendrant ainsi une erreur faible d'ajustement. En revanche, un tel modèle peut s'avérer défaillant lorsqu'il s'agira de prévoir ou de généraliser, c'est-à-dire de s'appliquer à des données qui n'ont pas pris part à sa construction. L'objectif consiste donc à trouver le meilleur compromis entre le biais et la variance en contrôlant l'ajustement aux données et la complexité du modèle.[21]

III. Apprentissage profond (Deep Learning)

Le Deep Learning est un nouveau domaine de recherche de la machine Learning (ML), qui a été introduit dans le but de rapprocher le ML de son objectif principal à savoir : l'intelligence artificielle. Il concerne les algorithmes inspirés par la structure et du fonctionnement du cerveau. Ils peuvent apprendre plusieurs niveaux de représentation dans le but de modéliser des relations complexes entre les données. Les modèles de Deep Learning sont bâtis sur le même modèle que les perceptrons multicouches qu'on va décrire ensuite. Cependant, il convient de souligner que les différentes couches intermédiaires sont

plus nombreuses. Chacune des couches intermédiaires va être subdivisée en sous partie, traitant un sous problème, plus simple et fournissant le résultat à la couche suivante, et ainsi de suite. [24]

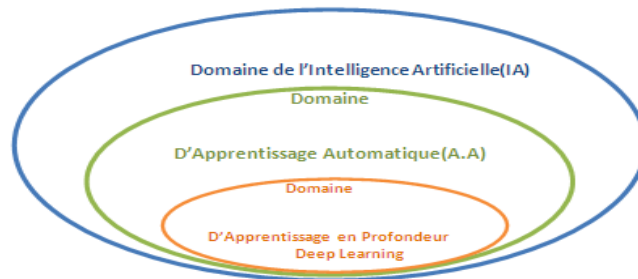


Figure 20 : La relation entre l'intelligence artificielle, le ML et le Deep Learning

L'apprentissage profond est basé sur ce qui a été appelé, par analogie, des « réseaux de neurones artificiels », composés de milliers d'unités (les « neurones ») qui effectuent chacune

De petites opérations simples. Les résultats d'une première couche de « neurones » servent d'entrée aux calculs d'une deuxième couche et ainsi de suite.

L'apprentissage approfondi permet de découvrir des structures complexes dans de grands ensembles de données en utilisant l'algorithme de micro propagation pour indiquer comment une machine doit modifier ses paramètres internes qui sont utilisés pour calculer la représentation dans chaque couche par rapport à la représentation de la couche précédente. [24]

III.1 Réseaux de Neurones Artificiels

De nos jours, on ne peut pas parler de machine Learning sans parler de réseau de neurones artificiels, un ensemble d'algorithmes dont le fonctionnement est inspiré des neurones biologiques et qui s'appuie également sur les méthodes statistiques. L'usage de réseaux de neurones est une technique de l'Intelligence Artificielle, optimisant les méthodes d'apprentissage automatiques [23].

III.1.1 Le neurone

Ce qui forme les réseaux de neurones, ce sont les neurones artificiels inspirés du vrai neurone qui existe dans notre cerveau. Les 2 figures suivantes montrent une représentation d'un neurone réel et d'un neurone artificiel :

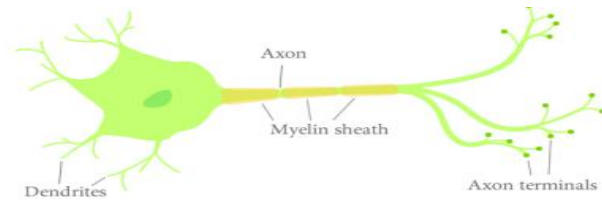


Figure 21 : Un neurone réel[23].

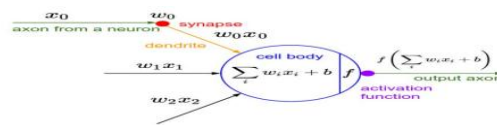


Figure 22 : Un neurone artificiel. [25]

Les X_i sont des valeurs numériques qui représentent soit les données d'entrée, soit les valeurs sorties d'autres neurones. Les poids W_i sont des valeurs numériques qui représentent soit la valeur de puissance des entrées, soit la valeur de puissance des connexions entre les neurones. Il existe des opérations qui se passent au niveau du neurone artificiel. Le neurone artificiel fera un produit entre le poids (w) et la valeur d'entrée (x), puis ajoutera un biais (b), le résultat est transmis à une fonction d'activation (f) qui ajoutera une certaine non-linéarité.

Les réseaux de neurones fonctionnent en répartissant les valeurs des variables dans des automates (les *neurones*). Ces unités sont chargées de combiner entre elles leurs informations pour déterminer la valeur du paramètre de discrimination. C'est de la connexion de ces unités entre elles qu'émerge la capacité de discrimination du RN. Chaque neurone reçoit des informations numériques en provenance de neurones voisins ; à chacune de ces valeurs est associé un *poids* représentatif de la force de la connexion. Chaque neurone effectue localement un calcul dont le résultat est transmis ensuite aux neurones avals. [25]

III.1.2 Perceptron simple d'un neurone

Il s'agit d'un neurone binaire, c'est-à-dire dont la sortie vaut 0 ou 1. Pour calculer cette sortie, le neurone effectue une somme pondérée de ses entrées (chaque entrée possède un poids) :

$$Y = (W_1 \times X_1 + W_2 \times X_2) \dots \dots \dots -6$$

Puis applique une fonction d'activation à seuil : si la somme pondérée dépasse une certaine valeur, la sortie du neurone est 1, sinon elle vaut 0. Il ne peut donc faire que de la classification, puis de la prédiction. [26]

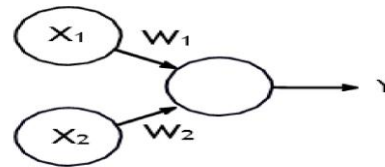


Figure 23 : : schéma du perceptron simple.[26]

III.1.3 Perceptron multicouches (PMC)

La famille de réseau de neurones majoritairement employée est le perceptron multicouche. À lui seul ce type de réseau recouvre plus de 95 % des applications scientifiques et industrielles. Il comporte quelques dizaines quelques centaines de neurones dans les cas usuels, voire plusieurs milliers pour les applications graphiques.

Le PMC est un modèle de réseau à propagation par couche. Les neurones y sont organisés en couches successives : une couche d'entrée, une couche de sortie et entre les deux une ou plusieurs couches intermédiaires, appelées aussi couches cachées. Il n'existe pas de connexion entre les neurones d'une même couche, en revanche tout neurone d'une couche est connecté à tous les neurones de la couche suivante. La « couche » d'entrée n'est pas une réelle couche de neurones car elle se contente de coder les variables d'observation. La couche de sortie code la variable de discrimination. Les valeurs d'activité des neurones sont propagées dans le réseau, de l'entrée vers la sortie, sans retour arrière..[26]

La (Figure 25) illustre l'estimation de l'âge au décès à partir de l'observation de critères osseux de la surface sacro-pelvienne iliaque. Les entrées correspondent à l'observation de critères morphologiques sur la surface sacro.

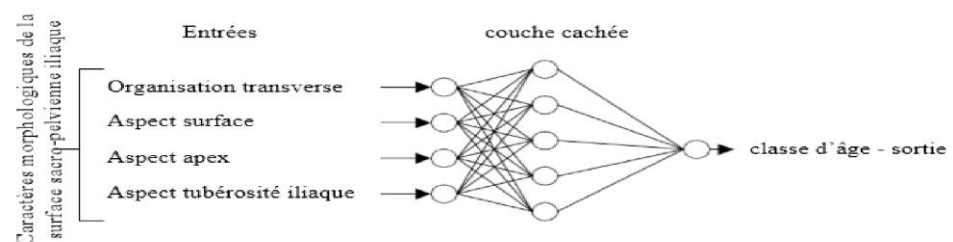


Figure 24 : Perceptron multicouche. [26]

III.1.4 La descente de gradient

Le processus d'apprentissage dans le DL revient à l'entraînement du réseau neuronal en utilisant des optimiseurs itératifs, qui ne font que conduire la fonction de coût à une très faible valeur[27]. Nous pouvons utiliser différents algorithmes pour effectuer l'apprentissage, mais l'algorithme le plus utilisé est l'algorithme itératif d'optimisation par **la descente de gradient** qu'est la méthode la plus utilisée presque sur tous les réseaux neuronaux avec ses différents modèles [27]. Le processus d'apprentissage revient au problème d'optimisation où il s'agit de minimiser ou de maximiser une fonction $f(x)$. Cette fonction est appelée la fonction objective, les auteurs dans [27].l'ont appelée aussi la fonction de coût, la fonction de perte et la fonction d'erreur. Pendant l'entraînement, l'algorithme tente d'identifier le minimum global sans tomber dans le piège du minimum local. Le schéma suivant réalisé par les auteurs en [27].est une illustration de ce concept :

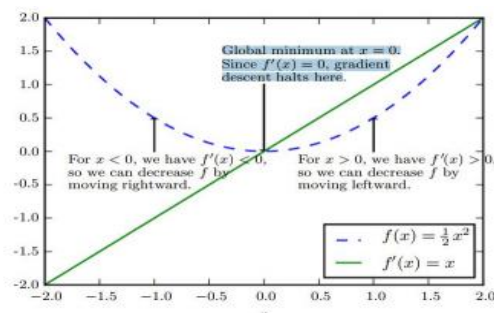


Figure 25 :Une illustration du processus de recherche de l'optimum.[27]

III.1.4.1 Les Algorithmes d'optimisation de la descente de gradient

Il existe trois principaux types de variantes de l'algorithme de descente de gradient. La principale différence entre eux est la quantité de données que nous utilisons lorsque nous calculons le gradient pour chaque étape d'apprentissage[28][27]. Les algorithmes d'optimisation qui utilisent tout l'ensemble de l'apprentissage sont appelés les méthodes de gradients déterministes ou batch descente de gradient, car ce sont des

Méthodes où tous les exemples d'apprentissage sont traités simultanément dans un grand batch, tandis que le terme "batch" pour décrire un groupe d'exemples.

Adam : Adam est un algorithme d'optimisation présenté en 2015 [29]. Le nom de cet algorithme est dérivé d'Adaptive Moment Estimation. Lors de l'introduction de cet algorithme, les auteurs ont présenté les avantages de l'utilisation d'Adam sur des problèmes d'optimisation non convexes, comme suit :

- Simplicité de mise en œuvre.
- Efficacité du calcul.
- Peu de mémoire requise.
- Bien adapté aux problèmes volumineux en termes de données et/ou de paramètres.
- Les hyper-paramètres nécessitent généralement peu de réglages.

Il existe d'autres optimiseurs avec différents mécanismes de fonctionnement, comme :

- Adagrad
- RMS Prop
- Adadelta

III.1.5 Les fonctions d'activation

Les fonctions d'activation sont une composante extrêmement importante des réseaux de neurones artificiels. Ils décident si un neurone doit être activé ou non. Si l'information que le neurone reçoit est pertinente pour l'information donnée ou si elle doit être ignorée .[30]

$$Y = \text{Activation} (\sum (w \times x) + b) \text{-----}(7)$$

W : poids , **X** : entrée , **B** : biais

Les fonctions d'activation peuvent être divisées en 2 types : **(voir l'annexe)**

Les différents algorithmes du deep Learning ne sont apparus qu'à l'échec de l'apprentissage automatique tentant de résoudre une grande variété de problèmes de l'intelligence artificielle (l'IA), parmi eux :

- ✓ Améliorer le développement des algorithmes traditionnels dans de telles tâches de l'IA.

- ✓ Développer une grande quantité de données telle que les big data.
- ✓ S'adapter à n'importe quel type de problème.
- ✓ Extraire les caractéristiques de façon automatique [22].

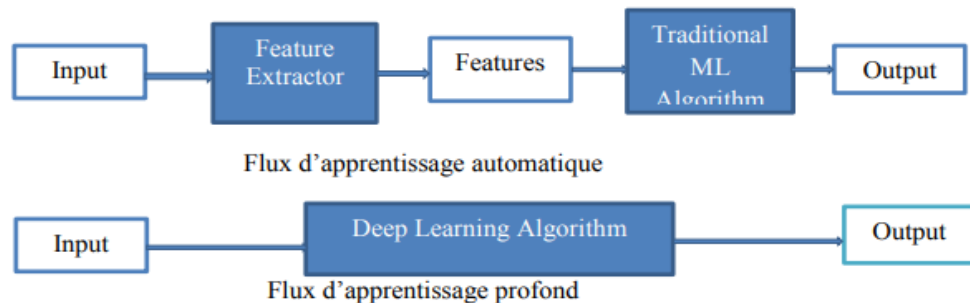


Figure 26: Comparaison entre la machine Learning et le DeepLearning.[22]

IV. Les architectures du Deep Learning

Bien qu'il existe un grand nombre de variantes d'architectures profondes. Il n'est pas toujours possible de comparer les performances de toutes les architectures, car elles ne sont pas toutes évaluées sur les mêmes ensembles de données. Le Deep Learning est un domaine à croissance rapide, et de nouvelles architectures, variantes ou algorithmes apparaissent toutes les semaines.

La plupart des architectures profondes sont réalisées en combinant et recombinaison un ensemble limité de primitives architecturales (des couches de réseaux neuronaux) [23]., Nous disons que notre réseau de neurones est profond lorsque le nombre des couches cachées est supérieur à 1. [34]

IV.1 Les réseaux entièrement connectés

Un réseau entièrement connecté (FullyConnected en Anglais) permet de transformer une liste d'entrées en une liste de sorties. La transformation est appelée totalement connectée car toute valeur d'entrée peut affecter toute valeur de sortie. Ces couches auront de nombreux paramètres apprenables, même pour des entrées relativement petites[22], mais elles ont le grand avantage de n'assumer aucune structure dans les entrées. Les calculs sont une série de transformations qui changent les similarités entre les cas. Dans ce

type de réseaux, les activités des neurones de chaque couche sont une fonction non-linéaire des activités de la couche inférieure [22]

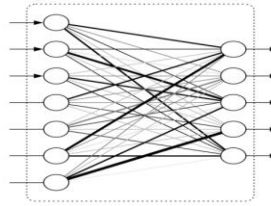


Figure 27: : Un réseau entièrement connecté.[22]

IV.2 Les réseaux de neurones convolutifs

Les réseaux de neurones convolutifs (CNN, ou ConvNet) est l'un des algorithmes les plus populaires dans le domaine de l'apprentissage en profondeur. Comme les autres réseaux de neurones, un CNN est composé d'une couche d'entrée, d'une couche de sortie et de nombreuses couches cachées (Figure 32)(Annexe). La distribution hiérarchique la plus courante des couches CNN contient :

- **Couche d'entrée:** le volume d'entrée est une image dont les dimensions sont les suivantes: [largeur x hauteur x profondeur]. Il s'agit d'une matrice de valeurs de pixels;
- **Les couches convolutifs:** l'objectif d'une couche convolutifs est d'extraire les caractéristiques du volume d'entrée;
- **Couches non linéaires:** la plupart du temps utilisant une fonction d'activation ReLU qui applique une activation élémentaire par seuillage à zéro;
- **Couches Pooling**(Figure 33): dans cette couche un sous-échantillonnage spatial est appliqué;
- **Couche entièrement connectée (FC):** Les couches entièrement connectées connectent chaque neurone d'une couche à chaque neurone d'une autre couche. [35]

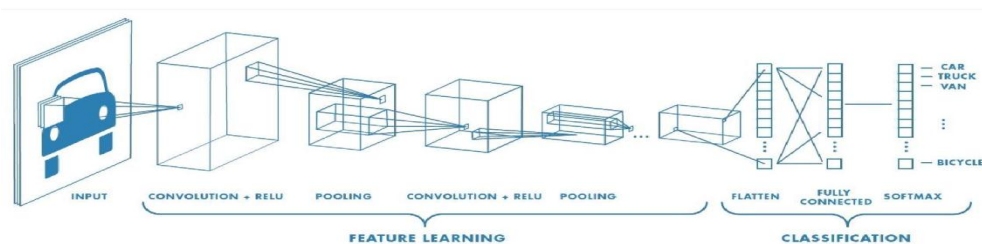


Figure 28 : Architecture d'un réseau de neurones convolutifs.[35]

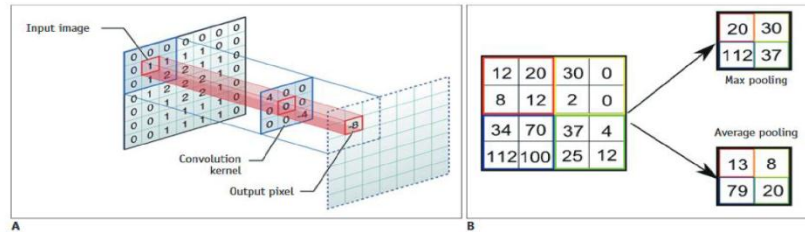


Figure 29: Illustration des méthodes de convolution et de Pooling.[35]

IV.3 Réseau de neurones récurrent

L'idée derrière les RNN est d'utiliser des informations séquentielles. Dans un réseau neuronal traditionnel, nous supposons que toutes les entrées (et les sorties) sont indépendantes les unes des autres. Mais pour de nombreuses tâches, c'est une très mauvaise idée. Si on veut prédire le prochain mot dans une phrase, il faut connaître les mots qui sont venus avant. Les RNN sont appelés récurrents, car ils exécutent la même tâche pour chaque élément d'une séquence, la sortie étant dépendante des calculs précédents.(figure 34)

Une autre façon de penser les RNN est qu'ils ont une « mémoire » qui capture l'information sur ce qui a été calculé jusqu'ici. En théorie, les RNN peuvent utiliser des informations dans des séquences arbitrairement longues, mais dans la pratique, on les limite à regarder seulement quelques étapes en arrière..[36]. Il est utilisé pour :

- La modélisation du langage et génération de texte
- La traduction automatique
- La reconnaissance vocale
- Et la description des images

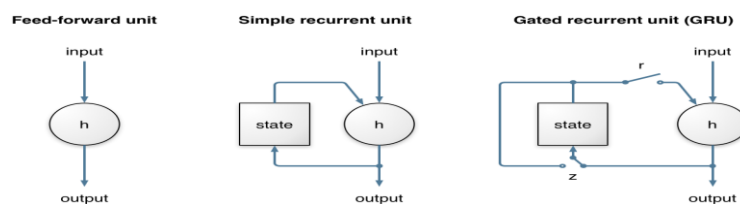
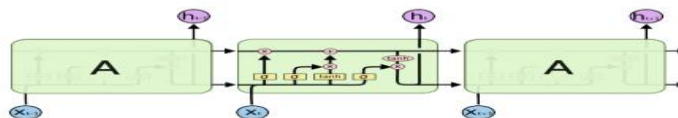


Figure 30 : Réseau récurrent à portes .[36]**IV.4 Long Short-Term Memory (LSTM)**

Les réseaux de mémoire à long terme à court terme généralement appelés simplement (LSTM : Long Short Term Memory) sont un type spécial de RNN. Ils ont été introduits par HochreiterSchmidhuber (1997). Les Réseaux neuronaux récurrents présentés dans la section précédente sont capables d'apprendre des règles de mise à jour de séquence arbitraire en théorie. Dans la pratique, cependant, ces modèles oublient généralement rapidement le passé. C'est ce qu'on appelle le problème de la disparition de gradient et c'est pourquoi ils ont inventé le LSTM. La cellule LSTM est une adaptation de la couche récurrente qui permet aux signaux plus anciens des couches profondes de se déplacer vers la cellule du présent[37]. La figure suivante représente une chaîne de trois cellules LSTM :

**Figure 31 : Une chaîne de cellules LSTM.[36]****IV.5 Modèle génératif**

Si les modèles discriminatifs comme (CNN, RNN) sont utilisés pour prédire les données du label et de l'entrée, tant disque le modèle génératif décrit comment générer les données, il apprend et fait des prédictions en utilisant la loi de Bayes .[38] Cependant les modèles génératifs sont capables de bien plus que la simple classification comme par exemple générer de nouvelles observations.

Voici quelques exemples de modèle génératif :

- Boltzmann Machines
- Restricted Boltzmann Machines
- DeepBeliefNetworks
- Deep Boltzmann Machines
- GenerativeAdversarial Networks
- GenerativeStochasticNetworks
- Adversarial Autoencoders

Conclusion

Nous avons présenté au cours de ce chapitre l'apprentissage automatique avec ses différents types, supervisé, non- supervisé et avec renforcement, les réseaux de neurones tel que le perceptron simple et le perceptron multicouches, ensuite nous avons présenté l'apprentissage profond et ses différentes méthodes et quelques domaines d'applications dont il est utilisé.

**Étude des Travaux utilisant les techniques de
Machine Learning pour le diagnostic des
maladies oculaires**

Introduction

Comparativement aux techniques traditionnelles, il a été démontré que le DL atteint des précisions considérablement plus élevées dans de nombreux domaines, notamment le traitement du langage naturel, la vision par ordinateur et la reconnaissance vocale.

En médecine et en soins de santé, le DL a été principalement appliqué à l'analyse d'imagerie médicale des différentes pathologies, dans lesquelles les systèmes de DL ont montré des performances de diagnostics robustes. En effet, le DL a été appliqué dans la détection des principales maladies ophtalmiques comme **la rétinopathie diabétique(RD), le glaucome, la dégénérescence maculaire liée à l'âge (DMLA)**.

L'un des principaux avantages du DL en ophtalmologie pourrait être le **diagnostique**, comme pour la RD, pour lesquelles il existe des lignes directrices bien établies. Autres conditions, comme le glaucome et la DMLA, peut également exiger le dépistage et le suivi à long terme. Toutefois, le diagnostique exige une main-d'œuvre considérable et des sources financières des systèmes de soins de santé, dans des pays développés et des pays à faible revenu et pays à revenu intermédiaire [39].

L'utilisation du DL, couplée avec la télémédecine, peut être une solution à long terme pour **dépister** et **surveiller** les patients soins oculo-visuels.

Ce chapitre commence par des définitions liées aux critères d'évaluation d'un modèle en intelligence artificiel (IA) ensuite nous présentons quelques travaux antécédentes récentes et les mieux cités en Google Scholar pour les applications de diagnostic ophtalmologiques avec différentes méthodes et modèles ainsi que les résultats obtenus et les défis potentiels liés au déploiement clinique,

I. Critères d'évaluation du modèle d'apprentissage

Pour évaluer la qualité d'un modèle prédictif, il y a trois éléments à prendre en compte :

- La qualité des données d'apprentissage,
- La pertinence et la robustesse de l'algorithme d'apprentissage,

- L'erreur de prédiction du modèle généré.

I.1 Ensemble de données d'apprentissage

Un ensemble de données d'apprentissage doit au minimum comporter les caractéristiques suivantes :

- ✓ Les données doivent être en nombre suffisant,
- ✓ Les données doivent être représentatives de conditions réelles,
 - Complétude des cas (classes de réponse),
 - Représentativité des cas et des situations,
 - Sans biais à priori.

Dans le cas de l'apprentissage supervisé, les étiquettes doivent être standardisées.

I.2 Choix de l'algorithme d'apprentissage

Un algorithme d'apprentissage possède deux failles potentielles qu'il faut savoir détecter. La première faille est un mauvais compromis biais/variance. Cette dernière engendre un sous- ou un sur-apprentissage. Cette erreur très connue est aujourd'hui absente des modèles générés par apprentissage, mais une vérification

La seconde faille est moins connue, c'est le problème de robustesse de l'algorithme d'apprentissage. En effet, les algorithmes d'apprentissage prennent des paramètres initiaux fixés aléatoirement. Il se peut qu'en changeant ces paramètres initiaux, le modèle apprenne différemment sur le même ensemble de données. On dit alors que le modèle ne converge pas de manière stable. Afin de détecter cela, il faut itérer des dizaines de fois l'apprentissage avec des paramètres initiaux variant et un ensemble de données d'apprentissage qui ne varie pas. Si le modèle généré diffère de manière significative, c'est que l'algorithme d'apprentissage n'est pas robuste. N'est pas à exclure.

Tableau 1: Algorithmes d'apprentissage automatique et leurs catégories.[40]

Le tableau suivant résume l'ensemble des algorithmes d'apprentissage automatique utilisés dans les travaux qui vont être explicités dans la section suivante de ce chapitre.

Techniques d'IA	Classification	Algorithmes
Apprentissage automatique conventionnel	Apprentissage supervisé	SVM, régression linéaire, régression logistique, RF, KNN, bayésienne naïve, arbre de décision, AdaBoost, méthodes de réseau neuronal
	Apprentissage non supervisé	Analyse en composantes principales, K-moyennes, maximisation des attentes, décalage moyen, clustering hiérarchique, propagation d'affinité, données itératives auto-organisées, systèmes de moyennes C floues
	Apprentissage par renforcement	Q-learning, apprentissage des différences temporelles, état-action-récompense-état-action, systèmes Teaching-Box, systèmes Maja
L'apprentissage profond	DBN	Réseau de croyances profondes convolutifs, machine Boltzmann restreinte conditionnelle
	CNN	AlexNet, GoogleNet, Réseau de groupes de géométrie visuelle (VGG), DeepResidual Learning, Inception v4 (v2, v3), Restnet-152 (34,50,101), LeNet
	RNN	RNN bidirectionnel, mémoire à court terme longue

I.3 Les mesures de performance

La précision, la sensibilité, la spécificité, le rapport de vraisemblance et le ROC / AUC ont été largement utilisés comme critères pour évaluer un modèle de diagnostic. Les termes

suivants sont fondamentaux pour comprendre leur utilités .[40]

- Vrai positif (TP): le patient a une maladie et la prédiction est positive.
- Faux positif (FP): le patient n'a pas de maladie mais la prédiction est positive.
- Vrai négatif (TN): le patient n'a pas de maladie et la prédiction est négative
- Faux négatif (FN): le patient a une maladie mais la prédiction est négative.

La précision d'un modèle de diagnostic fait référence à la capacité du modèle à identifier correctement les patients atteints de la maladie et sans la maladie:

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \text{-----8}$$

La sensibilité d'un modèle de diagnostic fait référence à la capacité du modèle à identifier correctement les patients atteints de la maladie:

$$\text{Sensitivity} = \frac{TP}{TP+FN} \text{-----9}$$

La spécificité d'un modèle de diagnostic fait référence à la capacité du test à identifier correctement les patients sans maladie:

$$\text{Sepecificity} = \frac{TN}{FP+TN} \text{-----10}$$

Le rapport de vraisemblance est défini comme le rapport entre les résultats attendus des tests chez les sujets atteints d'une certaine maladie et les sujets non atteints. Le rapport de vraisemblance pour des résultats de test positifs (LR +) nous indique dans quelle mesure le résultat du test positif est plus probable chez les sujets atteints de la maladie que chez ceux non atteints:

$$LR+ = \frac{\text{Sensitivity}}{1-\text{Specificity}} \text{-----11}$$

Le rapport de vraisemblance pour des résultats de test négatifs (LR-) représente le rapport entre la probabilité qu'un résultat négatif se produise chez les sujets atteints de la maladie et la probabilité que le même résultat se produise chez des sujets non atteints:

$$LR- = \frac{1-\text{Sensitivity}}{\text{Specificity}} \text{-----12}$$

Le graphique des caractéristiques de fonctionnement du récepteur (ROC) exprime la relation entre la sensibilité et 1 –Spécificité. Plus la courbe ROC est proche du coin supérieur gauche, meilleur est le modèle. L'aire sous la courbe (AUC) peut

avoir n'importe quelle valeur entre 0 et 1 et c'est un bon indicateur de la qualité du modèle.

II. Applications de L'Intelligence Artificielle IA en ophtalmologie

Nous avons recherché des revues et des études de recherche récentes utilisant plusieurs bases de données en plusieurs types (tomographie en cohérence optique OCT et Fond d'œil FD) dans les établissements de soins de santé ophtalmiques pour le diagnostic des maladies en mettant l'accent sur les maladies à forte prévalence et incidence, telles que la RD «rétinopathie diabétique », la DMLA «Dégénérescence maculaire liée à l'âge » et le glaucome qui se basent sur les techniques de machine Learning. La figure 1 présente une classification des travaux étudiés.

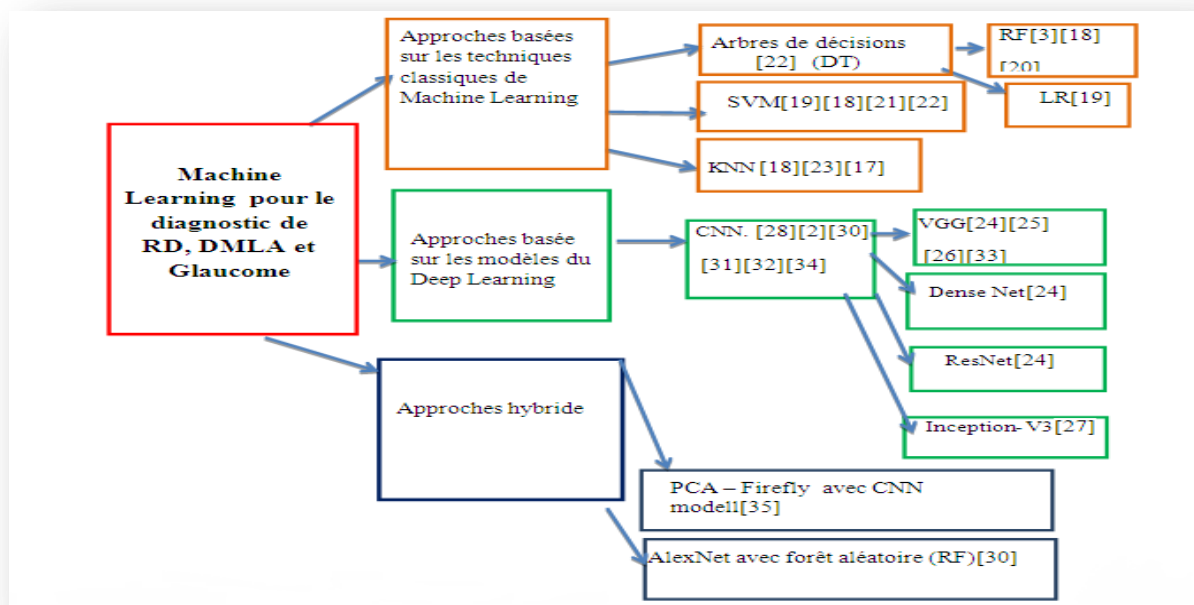


Figure 32 : Classification des travaux de diagnostic des maladies : RD, DMLA, Glaucome

II.1 Machine Learning pour le diagnostic de(RD, DMLA ,Glaucome)

Le machine Learning avec supervision peut être utilisé pour faire des prédictions sur des données indisponibles ou futures L'algorithme essaie de développer une fonction qui prédit avec précision la sortie à partir des variables d'entrée.

Les principaux algorithmes du machine Learning avec supervision sont :forêts aléatoires(RF), les arbres de décision (DT) , méthode du k plus proche voisin (k-NN),régression linéaire, classification naïve bayésienne, machine à vecteurs de support (SVM), régression logistique (LR) et boosting des gradients.

Nous présentons quelques travaux qui ont utilisé l’approche Machine Learning pour le diagnostic des trois maladies (Rétinopathie diabétique DR, glaucome, DMLA), avec différentes méthodes et bases de données. Dans le **tableau 3**

Tableau 2:Travaux basés sur les techniques de machine Learning classique pour la détection des maladies oculaires (Rétinopathie Diabétique, Glaucome, DMLA)

Approche	Année	Maladie	Donnés	Méthode	Performances
Tsao, H et all[41]	2018	RD	536FD private« DM shared care”	DT	(SVM)
				SVM	AUC 0.839
				LR	Sensibilité:0,933
				ANN	Spécificité :0,724
Schmidt-Erfurth, U et all[3]	2018	DMLA	614 (SD) OCT	RF	AUC 0.70
Kim SJ, Cho KJ[40]	2017	Glaucome	499 FD et Champ Visuel	RF	AUC 0.979
				SVM	AUC 0.967
				KNN	AUC0,971
Bogunović, Hrvoje, et al [42]	2017	DMLA	317 images OCT	RF	AUC0.7et 0.77
Carrera, E. V.et all[43]	2017	RD	400 images FD	SVM	AUC0.94 Sensibilité 0.95

Salam, A. A .et all[44]	2016	Glaucome	100 images FD	SVM	AUC 0.92 Sensibilité0.87
chowdhury, S.et all[45]	2013	RD	1 200 images FD Ensemble MESSIDOR DIARETDB1 89 images(FD)	GMM K NN SVM	KNN et gmm Sensibilité 1 Spécificité 0.531 SVM Sensibilité0.847 Spécificité 0.697 SVM+GMM Sensibilité0.991 Spécificité 0.5492
Niemeijer, M.et all[46]	2007	RD	430 images FD	KNN	Sensibilité 0.95 Spécificité 0.88

II.1.1 Travaux machine Learning obtenant meilleurs résultats avec la méthode support vector machine (SVM)

Tsao, H et all [4] ont appliqué différents algorithmes d'apprentissage automatique pour réaliser des prédictions précises et identifier de nouveaux facteurs de risque de la rétinopathie diabétique RD. Les résultats expérimentaux démontrent que les

machines à vecteurs de support ont obtenu les meilleures performances et précision, avec un ensemble de données de 536 photographies fond d'œil a obtenu une précision de 0.839, sensibilité de 0,933 et une spécificité de 0,724.

Carrera, E. V. et all [7] a utilisé la méthode SVM pour classifier automatiquement le grade de la rétinopathie diabétique non proliférative à n'importe quelle image rétinienne. Extraire des caractéristiques pour déterminer le grade de rétinopathie de chaque image rétinienne. Cette proposition a été testée sur une base de données de 400 images rétiniennes étiquetées selon une échelle à 4 niveaux de rétinopathie diabétique non proliférative. En conséquence, ils ont obtenu une sensibilité maximale de 95% et une capacité prédictive de 94%.

Salam, A. A. and all [8] ont fait le diagnostique du glaucome, sur un ensemble de 100 images FDLa sélection d'entités est effectuée à l'aide de l'analyse en composantes principales (ACP) pour réduire la dimension du vecteur d'entités en sélectionnant les vecteurs d'entités les plus significatifs pour la classification. La machine vectorielle de support (SVM) avec noyau linéaire est utilisée pour la formation et les tests. SVM recherche le meilleur hyper plan qui sépare les deux classes avec la distance maximale. résultant en une précision de 97% avec une spécificité de 0,98 et une sensibilité de 0,92

Le travail de Roychohury, S. [9] avec la base MESSIDOR de 1200 images ,il ont eu avec la méthode svm une sensibilité d sens 0.8474 et une spécificité de 0.6975

II.1.2 Travaux Machine Learning obtenant meilleurs résultats avec la méthode forêt aléatoire (RF)

Kim SJ, Cho KJ [3] ont plusieurs algorithmes de classifications pour le diagnostic du glaucome en fonction de l'épaisseur de la couche de fibres nerveuses rétiniennes (RNFL) et du champ visuel (FV), Le modèle de forêt aléatoire présente les meilleures performances et les modèles C5.0, SVM et KNN montrent une précision similaire. Dans le modèle de forêt aléatoire, la précision de classification est de 0,98, la sensibilité de 0,983, la spécificité de 0,975 et l'ASC de 0,979. sur un ensemble de données de 499 images FD

Schmidt-Erfurth, U et al[5] et Bogunović, Hrvoje, et al [6] ont utilisé la méthode forêt aléatoire (RF) pour le diagnostic du DMLA sur un ensemble des images OCT le premier avec 614 images et le deuxième avec 317 images les résultats de précision sont presque les mêmes pour les deux est 0.7 et 0,77 respectivement.

II.1.3 Travaux Machine Learning obtenant meilleurs résultats avec la méthode *k*voisin le plus proche (KNN) et GMM

Niemeijer, M. et al[10] a utilisé l'algorithme *k*voisinLe système a atteint une aire sous la courbe des caractéristiques de fonctionnement du récepteur (ROC) de 0,95 et des paires sensibilité / spécificité de 0,95 / 0,88 pour la détection de lésions lumineuses de tout type, et de 0,95 / 0,86, 0,70 / 0,93 et 0,77 / 0,88 pour la détection des exsudats, des tâches de coton et des drusens, respectivement. Le troisième spécialiste de la rétine a obtenu des paires de 0,95 / 0,74 pour les lésions claires et de 0,90 / 0,98, 0,87 / 0,98 et 0,92 / 0,79 par type de lésion.

Le travail de Roychowhury, S.[9] ont fait un système de diagnostic de la rétinopathie diabétique qui analyse les images du fond d'œil avec l'éclairage et les champs de vision, et génère un degré de gravité pour la rétinopathie diabétique (DR). Il ont utilisé des classificateurs comme le modèle de mélange gaussien (MGM), le voisin le plus proche

(kNN), la machine à vecteur de support (SVM) et AdaBoost sont analysés pour classer les lésions de la rétinopathie des non lésions. GMM et les classificateurs kNN sont considérés comme les meilleurs classificateurs pour des lésions rouges, les résultats obtenus sont 100% de sensibilité, 53,16% de spécificité et 0,904 AUC, par rapport à la meilleure sensibilité signalée de 96 %, 51% de spécificité et 0,875 CUA sur 1 200 images FD Ensemble MESSIDOR

II.1.4 Discussions et conclusions

Les méthodes support vecteur machine SVM, KNN ont montré les meilleurs résultats dans la détection de Rétinopathie Diabétique et glaucome : une précision atteinte avec la méthode SVM : AUC 0.967 et AUC 0.95 .

Comparant avec à la méthode forêt aléatoire (RF) une précision de AUC 0.7 et 0.77 dans la détection du DMLA.

Par contre la méthode forêt aléatoire (RF) a obtenu de bons résultats (RF) AUC 0.979 dans la détection du glaucome. ce qui montre qu'un algorithme de classification peut être meilleurs pour un problème qu'un autre.

Les travaux cité utilisant l'approche machine Learning travaille sur un ensemble limités de données maximum 536 FD pour Tsao, H et all [41] ce qui limite ses résultats et performances de leurs modèles.

II.2 Applications de diagnostique avec Deep Learning

Tableau 3: Tableau présente quelque systèmes utilisant Deeplearning pour détecter les maladies ophtalmiques (Rétinopathie Diabétique RD, Glaucome, Dégénérescence maculaire lié à l'âge DMLA)

Auteur	Année	Maladie	Donnés	Le modèle	Performances
Phan, S. all[47]	2019	Glaucome	3312 images Fond d'œil	VGG-19 ResNet152 DenseNet-201	VGG19 et le DenseNet201 AUC +0,97 DenseNet-201 AUC 0,9
Gómez-Valverde, J. J. et al[48]	2019	Glaucome	4316 images OCT (RS3000)	VGG-19, ResNet50, DenseNet et GoogLeN et	VGG-19 the best with AUC of 0.94, Sensibilité 0.87 Spécificité 0.89
Nagasawa, T. all [49]	2019	RD	378 images Fond d'œil	VGG-16	Sensibilité 0.94 Spécificité 0.97 AUC: 0,969
C. Tan. et al [50]	2019	RD	2500 fundus images	Inception V3	AUC90.9%
Burlina, P. M. and all. [51]	2019	DMLA	Local: 133821 PF	DCNN	AUC: 0,9706 (modèle formé sur des données réelles) AUC :0,9235

					(modèle formé sur des données synthétiques)
Peng, Yifan, et al.[52]	2018	DMLA	Public: AREDS, 59302 FPs	DeepSee Net	Précision: 0.94 (gros drusens) 0,93 (anomalies pigmentaire) 0,97(DMLA tardive)
Raghavendra, U. et al[2]	2018	Glaucome	1426 images de fond d'œil	CNN	Précision 0.98 Sensibilité 0.98 Spécificité 0.983
Muhammad, H and al[53]	2017	Glaucome	102 images OCT	CNN	AUC 0.931
Gargeya, R. et al [54]	2017	RD	Public: EyePACS 75137 PF	DCNN	AUC: 0.97 Sensibilité 0.94 Spécificité 0.98

Burlina, P. M., Joshi, N., Pekala, M & all[55]	2017	DMLA	Public: AREDS, plus de 130000 PF	DCNN	AUC 0,96
Abràmoff, M. D. and al[56]	2016	RD	Messidor-2 1748 Images FD	AlexNet/VGG	AUC : 0,98 Sensibilité 96,80 Spécificité 87,00 pour la détection de lésions lumineuses
Abbas, Q.[57]	2017	Glaucome	1200 Images FD	CNN DBN	AUC 0.99 Sensibilité0.845 Spécificité0.98

II.2.1 Travaux ont utilisé le model VGG pour le diagnostic des maladies ophtalmique

Une études récente de Phan, S. all[47] a utilisé trois modèles de DCNN : VGG-19 ,ResNet152 DenseNet-201 pour diagnostique du glaucome VGG19 et DenseNet201 ont obtenu une meilleur résultat plus de 0,97 dans l'accuracy sur un ensemble de donnés de 3312 images fond d'œil.

Une autre étude récente de Gómez-Valverde, J. J. et al[48] pour détecter le glaucome ce groupe de chercheurs a utilisé plusieurs modèles de cnn ,VGG-19, ResNet50, Dense Net et GoogleNet ,il ont eu une meilleure précision avec le model VGG -19 avec précision de 0,94 sensibilité de 87.01% spécificité de 89.01% sur un ensembles de 4316 images OCT.

Une autre étude de Nagasawa, T. all [49] et Abràmoff, M. D. and al[56] pour le diagnostique de la rétinopathie diabétique sur un ensemble de données de 378 images Fond d'œil et 1748imagesFD de Messidor-2 respectivement utilisant le model cnn VGG-16 le premier a atteint une accuracy de 0,969 et une Sensibilité de 94,7% et

une Spécificité: 97,2% et le deuxième a atteint AUC : 0,98 , sensibilité :96,80 et spécificité :87,00pour la détection de lésions lumineuses

II.2.2 Travaux ont utilisé leur propre modèle CNN pour le diagnostic des maladies ophtalmique

Raghavendra, U.et all[2] et Muhammad, H and all[53] ont fait un diagnostique du glaucome utilisant le model **CNN** le premier travail a eu une précision de 98,13%,sensibilité de 98% etspécificité de 98,3%. Avec un ensemble de donnés de **1426** images de fond d'œil et le deuxièmes une précision de 93.1% avec un ensemble de **102** images OCT

Un total de 75 137 images de fond de l'œil accessibles au public de patients diabétiques ont été utilisées pour former et tester R.et all [54]avec leur propre model cnn, a atteint une zone sous la courbe de 0,97 (AUC) avec une sensibilité et une spécificité de 94% et 98% sur une validation croisée de 5 fois .

Burlina, P. M. and all. [51] fait une étude pour le diagnostiqueLa dégénérescence maculaire liée à l'âge (**DMLA**) sur un ensemble de données de 133821 images du fond d'œil utilisant le model DCNN a obtenu une précision de AUC: 0,9706 sur un modèle formé sur des données réelles et AUC :0,9235sur un modèle formé sur des données synthétiques.

Abbas, Q.[57] a aussi fait le diagnostique du glaucome avec l'architecture non supervisée du réseau neuronal convolutif (CNN) pour extraire les caractéristiques par multicouches à partir des intensités brutes des pixels. Ensuite, le modèle du réseau de croyances profondes (DBN) a été utilisé pour sélectionner les caractéristiques profondes les plus discriminantes sur la base de l'ensemble de données d'entraînement de 1200 images fD. Enfin, la décision finale est effectuée par le classificateur linéaire softmax pour différencier l'image du fond d'oeil de la rétine glaucomateuse de celle de la rétine non glaucomateuseles résultat obtenu dans la sensibilité, spécificité et l'accuracy sont de 84,50 %, 98,01 %,99 %.

II.2.3 Travaux ont utilisé DeepSeeNet et Inception V3 pour le diagnostic des maladies ophtalmique:

Avec le modèle DeepSeeNet Burlina, P. M. and all. [51] ont fait un diagnostic de DMLA tardive avec une base de données Public: AREDS, 59302 FP Il ont obtenu une précision de 0,94 pour gros drusens et 0,93 pour anomalies pigmentaires 0,97.

Avec un ensemble de 2500 fundus images C. Tan. et al[50] ont utilisé le model Inception V3 il ont obtenu une précision de 90.9%

II.2.4 Discussions et conclusions :

Le modèle VGG a les meilleurs résultats de précision, la précision est atteinte AUC : 0,98

Sensibilité :96,80 Spécificité :87,00 pour la détection de lésions lumineuses dans la rétinopathie Diabétique pour le travail de Abràmoff, M. D.and al[56] et auc 0.97 pour Phan, S. all[47] pour le diagnostic de du glaucome par rapport aux autre modèles de transfère learning exemple AUC 90.9% a eu le travail de C. Tan. et al[50] avec le modèle inception-V3pour prédire la rétinopathie Diabétique avec un nombre limité de données.

Le modelDeepSeeNet aussi a atteint une haut précision 0,97 pour DMLA tardive avec le travail de Peng, Yifan, et al.[52].avec un grand ensembles de données Public: AREDS, 59302 FPs

Abràmoff, M. D.and al[56] a eu de bonne résultats avec un ensemble limité de données Messidor-2 1748 photographies FD ce qui montre que la taille de la base de données n'a pas toujours d'influence sur l'efficacité du modèle ce qui dévoile un autre facteur important c'est la qualité de la base et la préparation de données. qui joue un rôle important avant de faire la classification du model donc

Les données volumineux et bien traité évaluent la capacité d'un modèle d'apprentissage

II.3 Applications de diagnostic combiner DeepLearning et machine Learning

Tableau 4 : Tableau présente deux systèmes utilisant Deep Learning combiner avec machine Learning pour prédire les maladies ophtalmiques(Rétinopathie Diabétique, Glaucome)

II.3.1 Travaux combiner un modèle DeepLearning et une Méthode de Machine Learning dans le diagnostic des maladies

Gadekallu, T. R. and all[58] a améliorer la précision à 0.95 dans la détection de la

Auteur	Année	Maladie	Donnés	Méthode/model	Performances
Gadekallu, T. R. and all[58]	2020	RD	1151 images FD	PCA And Firefly CNN model	DNN AUC 0.96 Précision 0.95 DNN-PCA- Firefly AUC 0.97 Précision 0.96
Muhammad, H. et all[53]	2017	Glaucome	100 images OCT RNFL	AlexNet RF	Précision 0.931

rétinopathie diabétique utilisant un model CNN entraîné en combinant avec l'algorithmes d'analyse en composantes principales (ACP) pour la sélection des caractéristiques des données ,l'algorithmme Firefly a été utilisé pour la réduction de dimensionnalité. Cet ensemble de données réduits été introduit dans le réseau neuronal profond (DNN) qui a généré des résultats de classification le résultat de précision AUC est de 0.96% sur un ensemble de données de 1151 photographies fond d'œil.

Muhammad, H.[53] ont utilisé pour le diagnostic du glaucome un modèle CNN pré-entraîné pour l'extraction des caractéristiques et d'un modèle de forêt aléatoire un classificateur basé sur des arbres de décision. Les poids des nœuds dans les couches entièrement connectées du réseau neuronal sont utilisés comme entrée pour la forêt aléatoire. Combiner les deux approches Alex Net et forêt aléatoire, il a eu une précision de 93.1% sur un ensemble de 100 OCT de type RNFL

II.3.2 Discussion et conclusions

Les travaux cités précédemment ont eu des résultats perforantes jusqu'à AUC de 0.96% dans la rétinopathie diabétique et une précision de 93.1% dans la détection du glaucome ce qui montre qu'on peut intégrer les algorithmes d'apprentissage automatique dans les modèles d'apprentissage profond et combiner les points forts des deux approches pour augmenter la performance du modèle. Afin d'obtenir un bon diagnostic de la maladie.

III. Défis potentiels

Malgré le haut niveau de précision des modèles basés sur l'IA dans de nombreuses maladies en ophtalmologie, il y a encore beaucoup les défis cliniques et techniques de la mise en œuvre clinique et le déploiement en temps réel de ces modèles dans la pratique clinique (tableau 6). Ces défis pourraient se poser à différents stades, à la fois dans la recherche et le cadre clinique. Tout d'abord, de nombreuses études ont utilisé des ensembles de données de formation provenant de populations relativement homogènes. La formation et les tests d'IA à l'aide d'images rétiniennes sont souvent soumis à de nombreuses variabilités, notamment la largeur du champ, le champ de vue, l'agrandissement de l'image, la qualité de l'image et le participant ethnicités. Diversifier l'ensemble des données, en termes d'ethnicités, et Le matériel de capture d'images pourrait aider à relever ce défi.[59]

Tableau 5 :Tableau récapitulatif des différents défis potentiels d'un système de l'IA dans la détection des maladies ophtalmique.[59]

Étape	Défit potentiels
1-Identification des données relatives à la formation établit	<ul style="list-style-type: none"> • Le consentement des patients et les questions de confidentialité. • Des normes et des réglementations variables entre les différents comités de révision institutionnels • Petits ensembles de données de formation pour les maladies rares (par exemple, les tumeurs oculaires) ou les

	maladies courantes qui ne sont pas saisies en routine (par exemple, la cataracte).
2. Données de validation et d'essai Etablit	<ul style="list-style-type: none"> • Manque de taille des échantillons - pas assez de puissance. • Manque de généralisation - non testé à grande échelle dans différentes populations ou sur des données collectées à partir de différents appareils.
3-Explicabilité des résultats	<ul style="list-style-type: none"> • Démonstration des régions "jugées" anormales par DL. • Méthodes de génération de cartes thermiques - tests d'occlusion, activation de classe, méthode de gradient intégré, carte d'attention douce, etc.
4-Déploiement clinique du système de DL	<ul style="list-style-type: none"> • Recommandation des sites potentiels de déploiement clinique. • Application de l'approbation réglementaire des autorités sanitaires (par exemple, la Food and Drug Administration américaine, le marquage CE européen, etc.) • Application de l'approbation réglementaire des autorités sanitaires (par exemple, la Food and Drug Administration américaine, le marquage CE européen, etc.) • Conduite d'essais cliniques prospectifs. • Régime de remboursement des frais médicaux et exigence médicolégale. • Les enjeux éthiques.

Il est également important de souligner que l'"interopérabilité" des systèmes de DL ,ne peut pas signifier la même chose pour un professionnel de la santé que pour une machine expert en apprentissage. Bien qu'il semble probable qu'une interprétation des algorithmes seront plus facilement acceptés par les ophtalmologistes, la future recherche clinique appliquée

sera nécessaire pour déterminer si c'est le cas et si cela entraîne des avantages tangibles pour les patients en termes d'efficacité clinique.[39]

Conclusion

Les méthodes de diagnostic conventionnelles dépendent fortement de l'expérience et des connaissances professionnelles des médecins, ce qui peut conduire à un taux élevé d'erreurs de diagnostic et de gaspillage de données médicales

Le diagnostic d'image ne doit être utilisé que pour confirmer le diagnostic d'un médecin. Cependant, dans les pays en développement où le nombre de médecins peut également être limité. Pour traiter les problèmes médicaux les plus urgents dans le monde de manière efficace, rapide et rentable, toutes les ressources disponibles sont nécessaires. Par conséquent, l'introduction de l'intelligence artificielle dans le domaine médical est opportune, bienvenue et nécessaire.

Néanmoins, les réseaux de neurones à convolution profonde (DCNN) nécessitent des ensembles de données d'apprentissage d'image volumineux, diversifiés et bien équilibrés avec des étiquettes définissant les structures

Solution proposée

Introduction

L'analyse d'imagerie médicale et le diagnostic dans le domaine d'ophtalmologie ont attiré l'attention d'introduire des nouvelles techniques de vision par ordinateur. Après avoir parcouru la littérature et voir quelques travaux réalisés dans ce domaine, on a proposé un modèle pour le diagnostic des maladies ophtalmiques cités auparavant, Nous commençons ce chapitre par décrire l'architecture du notre système, ensuite la base de donnée utilisé, la phase d'entraînement et les paramètres lié à l'entraînement du modèle, une description détaillée des modèles CNN utilisés. Et on clôture ce chapitre par explicité la phase de test.

I. Architecture du système

Le système proposé permet de diagnostiquer trois maladies qui sont, DLMA, glaucome et RD en se basant sur des modèles de DeepLearning. Le diagnostic d'une maladie est un problème de classification. Par conséquent, la solution proposée est de construire un classificateur qui prédira une classe parmi quatre classes (Normal, Glaucome, Rétinopathie diabétique, DMLA). Pour ce faire :

On doit d'abord préparer le jeu de données. Ensuite on va choisir l'architecture de DeepLearning adéquate. Les réseaux de neurones convolutifs (CNN) sont très performants en classification binaire et multi-classes.

Etant un problème médical, on doit minimiser le taux d'erreur du modèle le plus que possible, c'est-à-dire une précision élevée et un rappel élevé. Donc, nous avons pris f1-micro avg . Comme métrique.

Différents travaux ont utilisé les méthodes CNN pour le diagnostic des maladies d'ophtalmologie, mais selon nos recherches, aucun n'a utilisé ces méthodes d'apprentissage en profondeur dans le même but que la nôtre. La combinaison des trois maladies d'ophtalmologie (DMLA, Glaucome, Rétinopathie diabétique) et la réutilisation de model CNN pré entraîné pour les maladies de la rétine (Module 1) constituent la contribution majeure de notre modèle.

Notre système consiste en deux modules principaux (voir figure 38):

Module 1 : Entrainement des modèles CNN pour les maladies de la rétine, avec des images de homographie par cohérence optique (OCT).

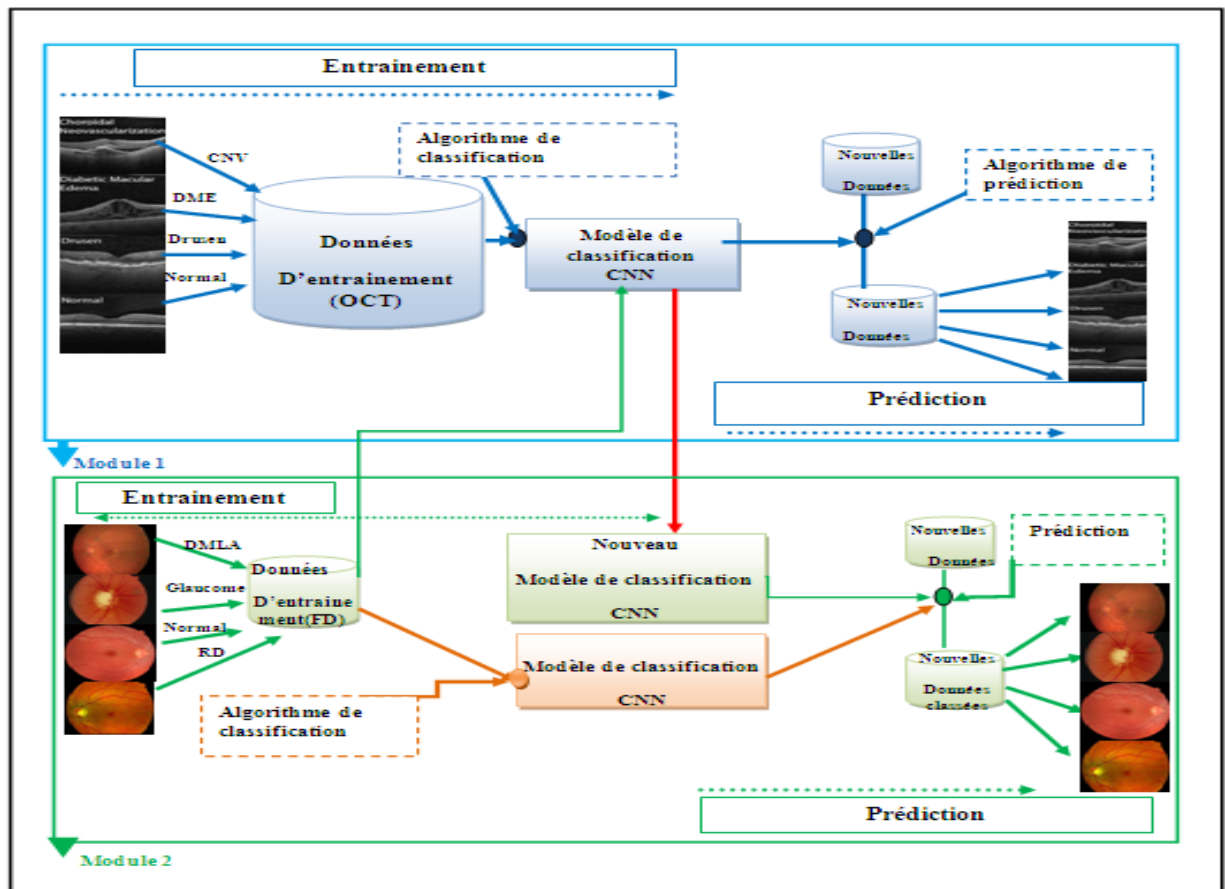


Figure 33 : Schéma global de la solution proposée.

Module2 : Entrainement du modèle CNN pour les trois pathologies d’ophtalmologies (Glaucome, DMLA, et la rétinopathie diabétique). On utilise les modèles du **Module 1** (Grande base d’entraînement) pour faire l’entraînement avec des images du **fond œil** (Petite base d’entraînement) en se basant sur le transfert Learning.

Chaque module comporte deux phases :

- **Phase d’entraînement** : La phase de l’entraînement est menée sur des réseaux neuronaux convolutifs profonds.
- **Phase de test** : La phase de test est la phase où le modèle proposé prouve ses performances et son efficacité. Les principaux objectifs du modèle proposé sont de classer

correctement les maladies d’ophtalmologies pour aider le médecin à prendre une décision finale concernant le diagnostic (Aide à la décision).

Avant de passer aux détails des deux phases, le DataSet que nous utilisons doivent d’abord être prétraité, ce qui va être détaillé dans la section suivante.

II. Data set

Le choix du bon ensemble de données pour une tâche de classification de donnée est d’une importance cruciale pour la performance et la généralisation des résultats.

Pour un problème de classification d’image, l’entrée X du réseau consiste en une variété d’images en niveaux de gris ou en couleur et leurs étiquettes de vérité de fond correspondantes Y. Pour toute tâche d’apprentissage automatique supervisé, il est courant de diviser l’ensemble de données en trois sous-ensembles appelés ensemble d’apprentissage, ensemble de test et ensemble de validation. Figure 38 fournit un aperçu schématique de ces divisions de jeux de données. Ils ont chacun un objectif différent et sont motivés comme suit (voir figure 38):

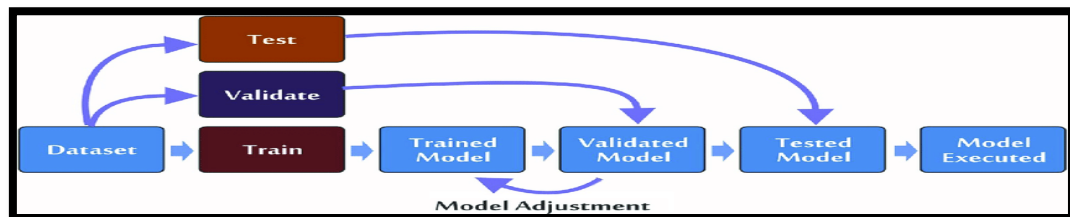


Figure 34 : Représentation des processus de formation (phase 1), de validation (phase 2) et de test (phase 3) du modèle de classification [60]

Détermination des images de l'application

Les images de l'application étaient celles que nous avons l'intention d'utiliser pour la classification des maladies d’ophtalmologies (**Module1, Module2**) après le développement du modèle. Cela a été fait pour que le modèle puisse être utilisé par n'importe qui, et pas seulement par un expert du domaine, qui est capable d’enter une nouvelle image de nouveau patient puis d’exécuter le modèle pour classer la maladie.

II.1 Ensemble d’entraînement (Training Set)

L'ensemble d’entraînement est généralement le plus grand des trois et est utilisé pour trouver les paramètres du modèle qui expliquent le mieux la relation prédictive sous-jacente entre les

données et leurs étiquettes. La plupart des approches qui adaptent les paramètres basés sur des relations empiriques à L'ensemble d'entraînement seul ont tendance à surcharger les données, ce qui signifie qu'elles peuvent identifier des relations apparentes dans les données de formation mais ne le font pas en général. C'est ce qui motive l'utilisation d'un ensemble de tests.

II.2 Ensemble de test (Test set)

L'ensemble de test est un ensemble de données qui n'est pas utilisé pendant l'entraînement, mais qui suit la même distribution de probabilité et la même relation prédictive. Si un modèle fonctionne bien sur l'ensemble d'apprentissage et s'adapte également bien à l'ensemble de test, c'est-à-dire prédit l'étiquette correcte pour un grand nombre de données d'entrée invisibles, un over-fitting minimal a eu lieu. Il est important de noter que l'ensemble de test n'est généralement utilisé qu'une seule fois dès que les paramètres et les hyper-paramètres du modèle sont entièrement spécifiés afin d'évaluer les performances de généralisation du modèle. Cependant, pour estimer les performances prédictives d'un modèle pendant l'entraînement, un jeu de validation est utilisé.

II.3 Ensemble de validation (Validation set)

L'ensemble de validation est créé en divisant l'ensemble d'entraînement en deux parties, la plus petite étant couramment utilisée pour l'optimisation des hyper-paramètres et pour éviter l'over-fitting grâce à un arrêt précoce (Earllystopping). Les hyper-paramètres sont décrits plus en détail dans la section 2.8 et font référence aux paramètres du modèle qui ne peuvent pas être ajustés avec la rétro-propagation. Afin de choisir des hyper-paramètres favorables et de maximiser les performances prédictives du modèle, un ensemble de validation est utilisé comme ensemble de données supplémentaire pour deux raisons. Si l'ensemble d'apprentissage a été utilisé pour ajuster simultanément les hyper-paramètres et les paramètres, le modèle a tendance à sur-ajuster l'ensemble d'apprentissage. Au contraire, si les hyper-paramètres étaient choisis par rapport à l'ensemble de tests, le modèle correspond implicitement aux hypothèses des données de test, ce qui atténuerait l'évaluation non biaisée de ses performances prédictives et donc de sa généralité.

Ces limitations motivent suffisamment l'utilisation d'un troisième ensemble pour la validation. L'arrêt précoce (Earlystopping) est une tentative universellement applicable pour trouver le moment optimal pour arrêter le processus de formation et pour décider quand un modèle est entièrement spécifié avec respect à ses paramètres et hyper-paramètres. Une technique courante consiste à surveiller en permanence la soi-disant précision de la validation et à arrêter l'entraînement lorsqu'il cesse de s'améliorer. La précision de la validation peut être considérée comme une approximation des performances de classification finale qui impose des hypothèses minimales sur la relation prédictive des données, à la fois en théorie et en pratique.

II.4 Préparation des données

Le prétraitement des images est une étape importante pour la suite car il permet d'uniformiser les images à traiter pour ne pas faire des traitements spécifiques pour certaines images. En effet les images étaient sur différentes formes, différentes résolutions, des images en RGB et des images en niveau de gris. Les images sont toutes redimensionnées. Les images à haute résolution ne peuvent pas être traitées en un seul passage par CNN en raison de leur grande taille. Alors que la majorité des CNN sont limités à une entrée de 256×256 pixels. Compte tenu des limites de la quantité de mémoire GPU.

Pour créer des modèles d'apprentissage profond utiles, l'erreur de validation doit continuer à diminuer avec l'erreur d'entraînement. L'augmentation des données est une méthode très puissante pour y parvenir. Les données augmentées représenteront un ensemble plus complet de points de données possibles, minimisant ainsi la distance entre l'ensemble d'apprentissage et de validation, ainsi que tout ensemble de tests futurs. Donc Nous avons effectué une augmentation de données en appliquant des déformations géométriques aux images : symétrie horizontale et verticale. Ceci permet aux modèles de mieux distinguer les caractéristiques invariantes et d'augmenter le nombre d'images ayant les mêmes concepts et légendes.

III. Phase d'entraînement du modèle

Dans cette phase, nous créons des modèles CNN et nous les entraînons sur un dataset des maladies d'ophtalmologie, jusqu'à ce que nous obtenions un modèle entraîné plus performant qui est prêt à être utilisé pour extraire les classes prédire. Cette phase est représentée dans la figure

ci-dessous. Au cours de nos expérimentations, nous avons créé 7 modèles CNN parmi lesquels les 4 derniers modèles utilisant la technique d'apprentissage par transfert (fine tuning le modèle) avec différentes architectures. Dans ce qui suit on présente l'architecture des sept modèles proposés.

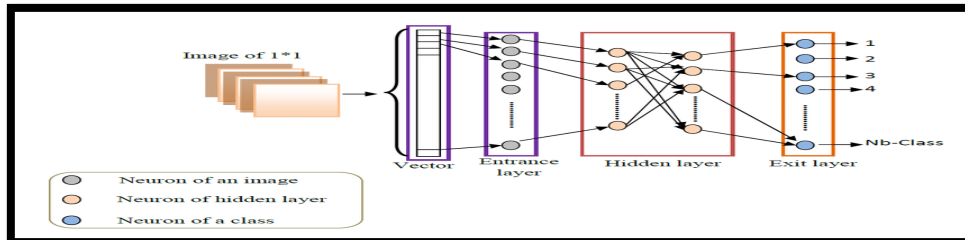


Figure 35 : Phase de classification par CNN. [61]

III.1 Modèles CNN

- Keras nous permet de spécifier le nombre de filtres souhaités et la taille des filtres. Donc, dans notre première couche, nous spécifions la forme de l'entrée et le nombre de filtres.
- La deuxième couche est la couche d'activation. Nous avons utilisé le **ReLU** (unité linéaire redressée) comme fonction d'activation. Elle fixe toutes les valeurs négatives de la matrice "x" à 0 et maintient toutes les autres valeurs constantes. C'est la fonction d'activation la plus utilisée car elle réduit le temps d'entraînement et évite le problème de la disparition des gradients.
- La troisième couche est la couche **MaxPooling**. La couche MaxPooling est utilisée pour sous-échantillonner les données d'entrée afin de permettre au modèle de faire des hypothèses sur les caractéristiques de manière à réduire le over-fitting. Elle permet également de réduire le nombre de paramètres à apprendre, ce qui réduit le temps de formation.
- Nous pouvons répéter les couches d'activation si nous voulons créer un CNN profond.
- C'est une bonne pratique de toujours faire de la normalisation par lots. La normalisation par lots normalise la matrice après qu'elle ait traversé une couche de convolution, de sorte que l'échelle de chaque dimension reste la même. Il réduit considérablement le temps de formation.
- Il est recommandé de toujours effectuer la normalisation par lots. **Batch Normalization** normalise la matrice après avoir traversé une couche de convolution

afin que l'échelle de chaque dimension reste la même. Cela réduit considérablement le temps de formation.

- **LeDropout** est la méthode utilisée pour réduire leover-fitting. Cela force le modèle à apprendre plusieurs représentations indépendantes des mêmes données en désactivant de manière aléatoire les neurones dans la phase d'apprentissage. Dans notre modèle, le décrochage désactivera aléatoirement 20% des neurones.
- Après avoir créé toutes les couches convolutionnelles, nous devons les aplatir, afin qu'elles puissent servir d'entrée aux couches denses. Les couches denses sont l'alias de Keras pour les couches entièrement connectées. Ces couches donnent la possibilité de classer les caractéristiques apprises par la CNN.
- La dernière couche est la couche d'activation **Softmax**. L'activation Softmax nous permet de calculer la sortie en fonction des probabilités. Chaque classe se voit attribuer une probabilité et la classe avec la probabilité maximale est la sortie du modèle pour l'entrée.

III.1.1 CNN à trois couches :

Nous allons commencer par entraîner un réseau de neurones convolutifs, Nous avons utilisée un noyau de 3*3 avec le même remplissage dans les couches de convolutions.

Le réseau possède 94,212 paramètres entraînales et il consiste en 3 couches de convolutions, une couche de MaxPooling2Det une GlobalAveragePooling2Dsont placés après les couches de convolutions,Nous avons utilisées BatchNormalizationpour normaliser les entrées dans une couchePourunapprentissageplus rapides et plus stables et une couche Dropoutpour la régularisation.

- **ReLu** sera la fonction d'activation.

Après ces trois couches de convolution, notre réseau de neurones compose d'une couche entièrement connectéetout en faisant des prédictions multi-classes ou la fonction d'activation utilisée est le softmaxqui permet de calculer la distribution de probabilité des 4 classes (nombre de classe dans la base d'images).

L'image en entrée est de taille 224*224.

Nous avons formé le modèle à l'aide de l'optimiseur Adam.Nous avons utilisé EarlyStopping pour empêcher le modèle de se surajouter, ReduceLROnPlateau pour

réduire le taux d'apprentissage après une perte de validation non améliorée à une époque et ModelCheckpoint pour enregistrer le modèle.

La figure suivante représente les différentes couches que contient notre modèle :

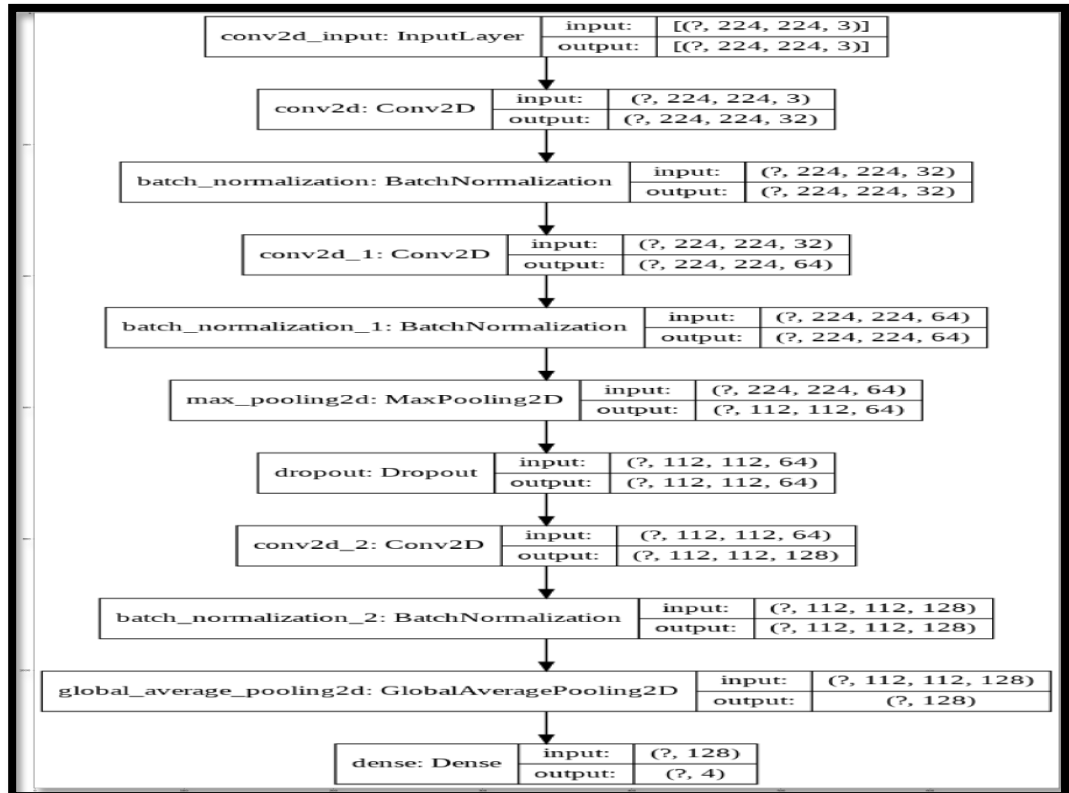


Figure 36– L’architecture du modèle CNN 3 couches.

III.1.2 Model à cinq couches :

Nous allons commencer par entraîner un réseau de neurones convolutifs, Nous avons utilisée un noyau de 3*3 avec le même remplissage dans les couches de convolutions.

Le réseau possède 557,636 paramètres entraînaibles et il consiste en 5 couches de convolutions, deux couches de MaxPooling2D et une GlobalAveragePooling2D sont placés après les couches de convolutions.

Nous avons utilisées BatchNormalization (à améliorer la précision et la vitesse Les formations de haut niveau) pour normaliser les entrées dans une couche Pour

unapprentissage plus rapides et plus stables et deux couches **Dropout** pour la régularisation.

ReLU sera la fonction d'activation pour chaque couche.

Après ces cinq couches de convolution, notre réseau de neurones compose d'une couche entièrement connectée ou la fonction d'activation utilisée est le softmax qui permet de calculer la distribution de probabilité des 4 classes (nombre de classe dans la base d'images).

L'image en entrée est de taille 224*224.

Nous avons formé le modèle à l'aide de l'optimiseur Adam. Nous avons utilisé EarlyStopping pour empêcher le modèle de se surajouter, ReduceLRonPlateau pour réduire le taux d'apprentissage après une perte de validation non améliorée à une époque et ModelCheckpoint pour enregistrer le modèle.

La figure suivante représente les différentes couches que contient notre modèle :

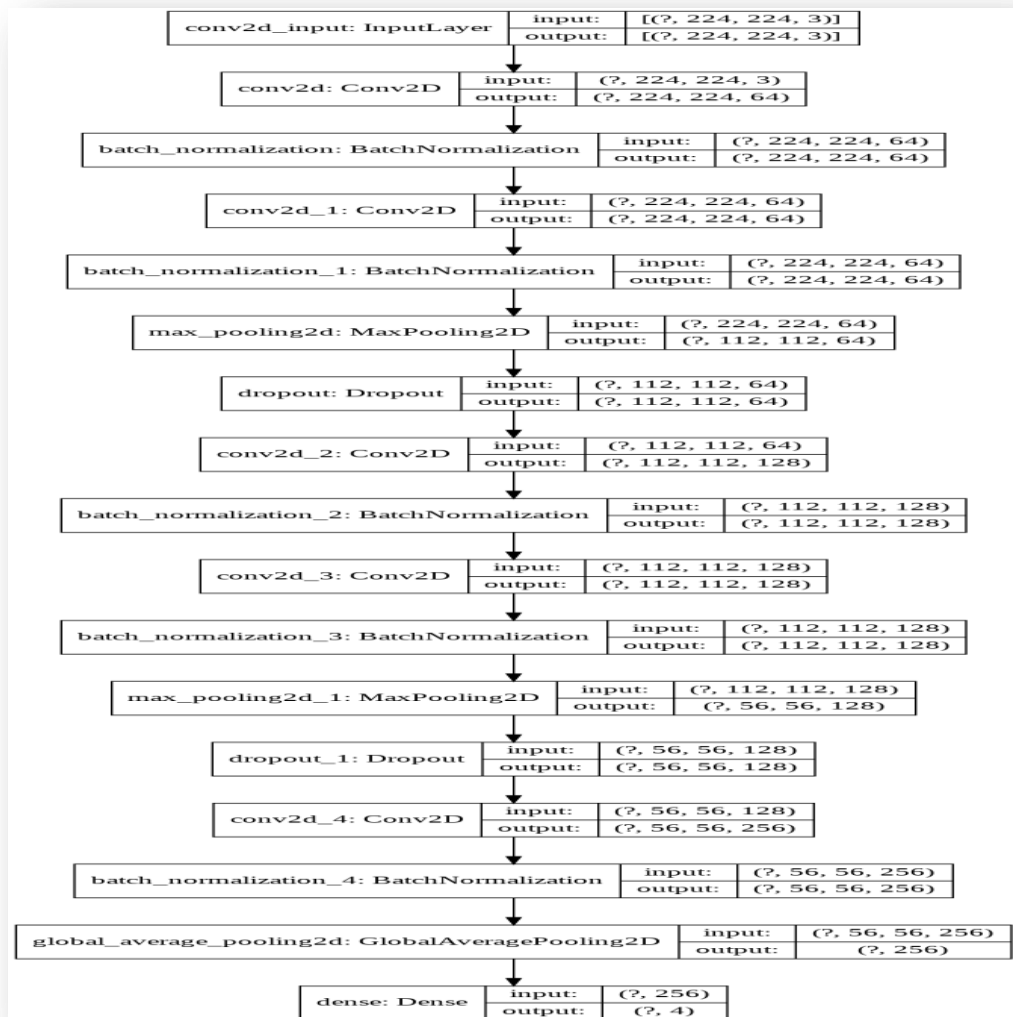


Figure 37 : L'architecture du modèle CNN 5 couches.

III.1.3 Model à sept couches :

Nous allons commencer par entraîner un réseau de neurones convolutifs, Nous avons utilisée un noyau de 3*3 avec le même remplissage dans les couches de convolutions. Le réseau possède **557,636** paramètres entraînaibles et il consiste en **7** couches de convolutions, trois couches de **MaxPooling2D** et une **GlobalAveragePooling2D** sont placés après les couches de convolutions, Nous avons utilisées **BatchNormalization** pour normaliser les entrées dans une couche Pour un apprentissage plus rapides et plus stables et trois couches **Dropout** pour la régularisation, **ReLU** sera la fonction

d'activation. Après ces cinq couches de convolution, notre réseau de neurones compose d'une couche entièrement connectée où la fonction d'activation utilisée est le **softmax** qui permet de calculer la distribution de probabilité des 4 classes (nombre de classe dans la base d'images). L'image en entrée est de taille 224*224.

Nous avons formé le modèle à l'aide de l'optimiseur **Adam**. Nous avons utilisé **EarlyStopping** pour empêcher le modèle de se surajouter, **ReduceLROnPlateau** pour réduire le taux d'apprentissage après une perte de validation non améliorée à une époque et **ModelCheckpoint** pour enregistrer le modèle.

La figure suivante représente les différentes couches que contient notre modèle :

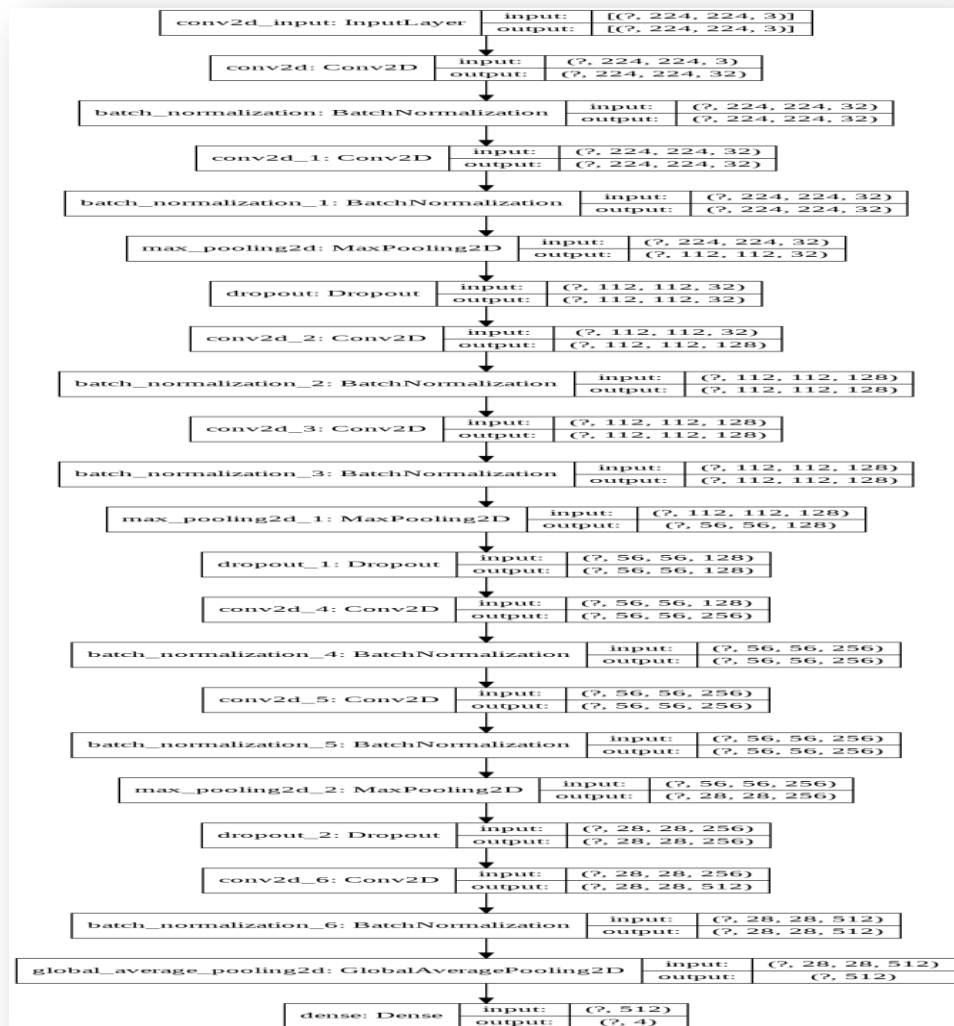


Figure 38 – L’architecture du modèle CNN 7 couches.

III.1.4 Apprentissage par transfert VGG16

VGG16 est un modèle de réseau neuronal convolusionnel proposé par K. Simonyan et A. Zisserman de l'Université d'Oxford dans l'article intitulé «Verydeepconvolutional networks for large-scale image recognition »[74].

Le modèle atteint une précision de 92,7% sur la base de données ImageNet [AKR12] qui contient plus de 14 millions d’images appartenant à 1000 classes.

C’est un modèle de 16 couches qui reçoit a l’entrée une taille d’image RVB de taille 224 x 224 VGG16 utilise les filtres de noyaux 3X3 dans les couche de convolution (qui est

la plus petite taille pour capturer la notion de gauche / droite, haut / bas, centre) et un filtre de taille 2X2 en maxpooling, toutes les couches cachées sont équipées de la non-linéarité de rectification (ReLU).

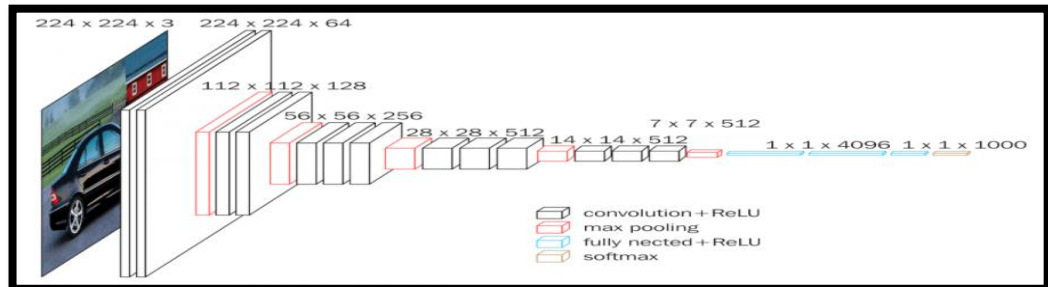


Figure 39: Architecture de VGG [75]

Et à la fin du réseau, trois couches entièrement connectées qui calcule le score de chaque classe des caractéristiques extraites de la convolution dans les étapes précédentes. Les cartes de caractéristiques de la couche finale sont représentées sous forme de vecteurs avec des valeurs scalaires, la fonction de classification soft-max. qui formule le score final entre 0 et 1 pour présenter la classification sous forme un pourcentage partagé entre les différentes classes, Les couches entièrement connectées sont coûteuses en termes de calcul, des approches alternatives ont été proposées au cours des dernières années. Par exemple la mise en œuvre d'une couche Global average pooling qui aide à réduire le nombre de paramètres dans le réseau de manière significative.

Dans ce modèle nous n'allons pas utiliser la couche de classification du VGG16 mais nous allons ajouter et entraîner notre propre classifieur, spécifique à la réponse au problème de nos propres données. Donc nous avons ajouté une couche personnalisée entièrement connectée pour le problème de classification en 4 classes (Module 1 : Normal, CNV, DME, DRUSEN, Module 2 : Normal, Glaucoma, Diabetic Retinopathy, AMD). Nous avons gelé les poids du modèle prétraité pendant l'entraînement.

Nous avons formé le modèle à l'aide de l'optimiseur Adam. Nous avons utilisé EarlyStopping pour empêcher le modèle de se surajuster, ReduceLRonPlateau pour réduire le taux d'apprentissage après une perte de validation non améliorée à une époque et ModelCheckpoint pour enregistrer le modèle.

III.1.5 Apprentissage de transfert ResNet50

Les réseaux neuronaux résiduels (ResNet) ont été développés par [76], ont obtenu la première place dans le défi de reconnaissance visuelle à grande échelle d'ImageNet (ILSVRC 2015) avec 3,57% de taux d'erreur. Les auteurs ont utilisé une architecture CNN de 152 couches dans le calcul. Le réseau ResNet a popularisé l'idée d'utiliser des réseaux plus profonds par rapport à AlexNet, qui a huit couches, et VggNet avec jusqu'à 19 couches. L'architecture ResNet a introduit le saut des connexions, également appelées connexions résiduelles pour éviter la perte d'informations pendant la formation des réseaux. La technique de saut de connexion permet de former des réseaux très profonds et peut stimuler la performance du modèle. Les auteurs ont pu former un modèle de 1001 couches profondes en utilisant des résidus connexions. L'architecture ResNet est principalement composée de blocs résiduels. Dans les connexions neuronales peu profondes.

Toutefois, dans l'architecture ResNet, des couches cachées consécutives sont reliées entre elles, il existe également des liens entre les blocs résiduels. Les principaux avantages des blocs résiduels les connexions dans l'architecture ResNet ; les connexions préservent les connaissances acquises pendant la formation et accélèrent le temps de formation du modèle en augmentant la capacité du réseau. ResNet-50 est un réseau neuronal convolutif de 50 couches de profondeur est présenté à la Figure 50.

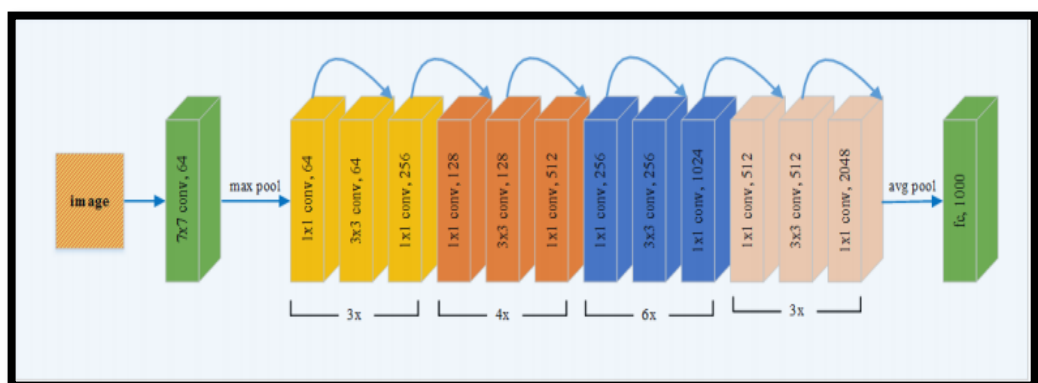


Figure 40: Architecture ResNet-50 préformée

Dans notre modèle, Nous avons utilisé le modèle préformé ResNet50 sans couche entièrement connectée. Nous avons ajouté une couche personnalisée entièrement connectée pour le problème de classification en 4 classes (Module1 : Normal, CNV, DME, DRUSEN, Module2 : Normal, Glaucoma, DiabeticRetinopathy, AMD). Nous avons gelé les poids du modèle préformé à l'exception des 3 dernières couches pendant la formation. Nous avons affiné les 3 dernières couches et les couches entièrement connectées.

Nous avons formé le modèle à l'aide de l'optimiseur Adam. Nous avons utilisé EarlyStopping pour empêcher le modèle de se surajouter, ReduceLRonPlateau pour réduire le taux d'apprentissage après une perte de validation non améliorée à une époque et ModelCheckpoint pour enregistrer le modèle.

III.1.6 Apprentissage de transfert DenseNet121

Les réseaux convolutionnels denses (DenseNet) développés par Huang, Liu et Maaten.[77] ont obtenu les meilleures performances de classification sur les ensembles de données d'images accessibles au public, tels que CIFAR-10 et ImageNet en 2017. L'architecture DenseNet utilise des connexions denses dans la construction du modèle, autres que des connexions directes entre les couches cachées du réseau, comme dans le modèle ResNet l'architecture. L'architecture DenseNet utilise moins de paramètres que les architectures CNN similaires pour la formation du réseau en raison de la transmission directe des cartes de caractéristiques à toutes les couches suivantes.

De nombreux modèles DenseNet sont utilisés, ici, DenseNet121 est utilisé comme l'un des modèles à assembler dans le modèle proposé, où 121 est le nombre total de couches présentes dans l'architecture. Le flux de tâches dans le modèle DenseNet121 peut être visualisé comme indiqué dans la figure 51.

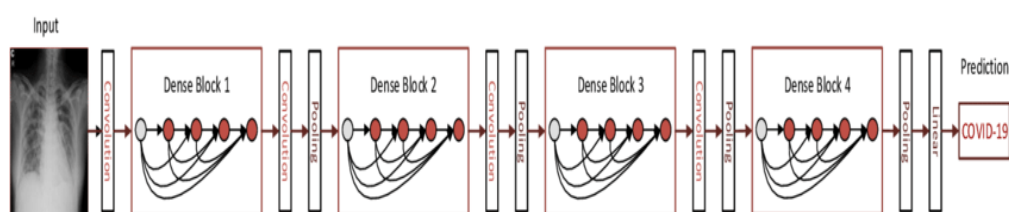


Figure 41 : Une architecture du modèle DenseNet121[77]

Dans notre modèle, Nous avons utilisé le modèle préformé DenseNet121 sans couche entièrement connectée. Nous avons ajouté une couche personnalisée

entièrement connectée pour le problème de classification en 4 classes (Module 1 : « Normal, CNV, DME, DRUSEN », Module 2 : « Normal, Glaucoma, Diabetic Retinopathy, AMD »).. Nous avons gelé les poids du modèle préformé sauf les 3 dernières couches pendant l'entraînement. Nous avons affiné les trois dernières couches et les couches entièrement connectées.

Nous avons formé le modèle à l'aide de l'optimiseur Adam. Nous avons utilisé EarlyStopping pour empêcher le modèle de se surajouter, ReduceLRonPlateau pour réduire le taux d'apprentissage après une perte de validation non améliorée à une époque et ModelCheckpoint pour enregistrer le modèle.

III.1.7 Apprentissage de transfert MobileNet

MobileNet est un modèle pré-formé populaire qui a été formé sur l'ensemble de données ImageNet. L'architecture MobileNet est définie dans la figure au-dessus. Toutes les couches sont suivies par une non-linéarité batchnorm.[64] et ReLU, à l'exception de la dernière couche entièrement connectée qui ne présente aucune non-linéarité et qui alimente une couche softmax pour la classification. MobileNet a 28 couches.

Table 1. MobileNet Body Architecture

Type / Stride	Filter Shape	Input Size	
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$	
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$	
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$	
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$	
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$	
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$	
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$	
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$	
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$	
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$	
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$	
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$	
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$	
5×	Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
	Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$	
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$	
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$	
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$	
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$	
FC / s1	1024×1000	$1 \times 1 \times 1024$	
Softmax / s1	Classifier	$1 \times 1 \times 1000$	

Figure 42. Architecture MobileNet

La formation d'un modèle semblable à MobileNet, qui a des millions de paramètres, est coûteuse en calcul et prend beaucoup de temps et pour gagner du temps et augmenter la précision, nous nous fierons à l'apprentissage par transfert pour notre objectif.

Dans notre modèle, nous avons rechargé le modèle MobileNet sans les couches supérieures ou FC, nous avons gelé les 4 dernières couches. Nous avons créé la tête du modèle qui sera placée sur les couches inférieures et nous avons ajouté notre FC Head à MobileNet.

Nous avons formé le modèle à l'aide de l'optimiseur **Adam**. Nous avons utilisé **EarlyStopping** pour empêcher le modèle de se surajuster, **ReduceLRonPlateau** pour réduire le taux d'apprentissage après une perte de validation non améliorée à une époque et **ModelCheckpoint** pour enregistrer le modèle.

IV. Phase de test

La phase de test est la phase où le modèle proposé prouve ses performances et son efficacité. Pour cela nous avons utilisé un ensemble de test qui n'est pas utilisé pendant l'entraînement, et qui a la même distribution de probabilité. L'objectif est d'obtenir un ensemble de prédictions correctes par rapport à un grand ensemble (données) d'entraînement. On va détailler cette phase dans le chapitre suivant.

Conclusion

Pour conclure, nous avons passé en revue les différentes phases et les différentes couches de nos modèles proposées. Plus important encore, nous avons creusé profondément dans chaque partie pour donner une meilleure compréhension des bases de tous les modèles utilisés. Dans le chapitre suivant, nous décrirons le processus que nous avons suivi pour mettre en œuvre ce modèle et les résultats que nous avons obtenus à la fin.

Implémentation et Test

Introduction

Maintenant que notre modèle a été conçu, il est temps de mettre la théorie en pratique et d'évaluer l'architecture proposé. Nous commencerons par présenter la configuration expérimentale que nous avons utilisée pour développer notre modèle et réaliser nos tests, puis nous partagerons plus de détails sur les données utilisées dans nos expériences et le processus suivi, et à la fin, nous partageons les résultats que nous avons obtenus.

I. Environnement de développement

I.1 Le hardware

Colaboratory[78] est un outil de recherche open source pour l'enseignement et la recherche en apprentissage machine. Il s'agit d'un environnement pour ordinateur portable Jupyter qui ne nécessite aucune installation ou bien une configuration pour être utilisé, il s'exécute entièrement dans le cloud. Cet environnement supporte Python 2.0 et ses versions supérieures. Il nous permet d'écrire et d'exécuter du code, de sauvegarder et partager, et d'accéder à de puissantes ressources informatiques depuis notre navigateur. Les expériences ont été réalisées sur une machine virtuelle dans l'environnement Google Colab qui offre une bonne performance dont les caractéristiques sont les suivantes :

Tableau 6: configuration du colab

CPU	GPU	RAM
Intel Xeon CPU 2.30GHz	NVIDIA Tesla T4	12 GB

Pourquoi les GPU ?

Pour la plupart des approches de l'apprentissage en machine ou l'apprentissage en profondeur, les GPU sont essentiels en raison de la quantité de données sur laquelle vous devez opérer. Le calcul mathématique de Deep Learning sur un CPU peut prendre des mois ! Mais ces calculs peuvent être assignés à des GPU pour un calcul plus rapide. Il n'est pas recommandé de former un réseau de neurones sur un processeur, un GPU est nécessaire pour un calcul élevé. Comme nous le savons tous, les GPU sont chers, de même que les plates-formes cloud telles qu'AWS et GCP. Alors, où pouvons-nous obtenir un GPU ? Colab vient à la rescousse ! Colab vous fournit un GPU Nvidia Tesla K80 gratuit dont le prix est d'environ 1,7 lakh Roupie et qui est

une véritable bête dans la performance. Vous pouvez utiliser ce GPU pour former votre réseau de neurones sur le cloud et profiter des performances rapides et sans tracas.[79]

I.2 Le software

I.2.1 Python

Python est un langage de programmation de haut niveau utilisé pour la programmation générale. Créé par Guido van Rossum et sorti en 1991, Python a une philosophie de conception qui met l'accent sur la lisibilité du code, notamment en utilisant des espaces importants. Il fournit des constructions qui permettent une programmation claire à petite et à grande échelle. Python dispose d'un système de type dynamique et d'une gestion automatique de la mémoire. Il prend en charge de multiples paradigmes de programmation, y compris orientés objet, impératifs, fonctionnels et procéduraux, et dispose d'une bibliothèque standard vaste et complète. Les interpréteurs de Python sont disponibles pour de nombreux systèmes d'exploitation.



Figure 43 : Le logo de python.

I.2.2 Bibliothèques utilisées

- **TensorFlow**

Un Framework de programmation pour le calcul numérique créé par Google et devenu un Framework Open Source en novembre 2015. Depuis, TensorFlow a continué à gagner une grande importance et popularité et rapidement devenu l'un des Frameworks les plus utilisés pour le Deep Learning. Son nom s'inspire notamment du fait que les opérations actuelles des réseaux de neurones s'effectuent principalement via une table multidimensionnelle de données, appelée Tenseurs (Tensor) qui est l'équivalent d'une matrice. Aujourd'hui, ce framework est utilisé par les grandes entreprises [79] comme Intel, Twitter, Contribution 58 Lenovo, Nvidia, etc.

- **Keras**

Un framework open source d'apprentissage profond pour le Python, capable de s'exécuter sur TensorFlow. Il est écrit par Francis Chollet, membre de l'équipe Google. Keras est utilisé dans un grand nombre de startups, de laboratoires de recherche [80](dont le CERN, Microsoft Research et la NASA) et de grandes

entreprises telles que Netflix, Yelp, Square, Uber, Google, etc. Il a été développé dans le cadre de l'effort de recherche du projet ONEIROS (Open-ended Neuro Electronic Intelligent Robot Operating System). En 2017, l'équipe TensorFlow de Google a pris la décision de fournir un support pour Keras et de l'intégrer dans la bibliothèque principale de TensorFlow. Il présente un ensemble d'abstractions de plus haut niveau et plus intuitives qui facilitent la configuration des réseaux neuronaux .[81]

▪ **Pandas**

Pandas est un package Python qui offre des structures de données rapides, flexibles et expressives conçues pour rendre le travail avec des données "relationnelles" ou "étiquetées" à la fois facile et intuitif.[82]

II. Aperçu du système

Notre système est composé de deux Modules:

II.1 Module 1

Un classifieur qui prédira une classe parmi les quatre classes (CNV, DME, DRUSEN, NORMAL).

II.1.1 Jeux de données

L'ensemble de données contient des sous-dossiers pour chaque catégorie d'image (NORMAL, CNV, DME, DRUSEN). Il existe 84 495 images de rayons X (JPEG) et 4 catégories (NORMAL, CNV, DME, DRUSEN). Il existe quatre classes dans les images OCT rétiniennes. Ils sont listés ci-dessous.). Les images sont étiquetées comme (maladie) - (ID patient randomisé) - (numéro d'image par ce patient) et divisées en 4 répertoires: CNV, DME, DRUSEN et NORMAL. Les images de tomographie par cohérence optique (OCT) (Spectralis OCT, Heidelberg Engineering, Allemagne) ont été sélectionnées à partir de cohortes rétrospectives de patients adultes du ShileyEye Institute de l'Université de Californie à San Diego, de la CaliforniaRetinalResearchFoundation, du Medical Center Ophthalmology Associates, de Shanghai First People'sHospital et Beijing TongrenEye Center entre le 1er juillet 2013 et le 1er mars 2017. Dans le jeu de données lié également, nous avons une structure de répertoires et donc ImageDataGenerator déduira les étiquettes. Une vue de la structure du répertoire du jeu de données est présentée ci-dessous:



Figure 44: (Extrême gauche) Néovascularisation choroïdienne (CNV) .[83]

La figure 2 représente (Extrême gauche) Néovascularisation choroïdienne (CNV) avec membrane néovasculaire (têtes de flèches blanches) et liquide sous-rétinien associé (flèches). (Milieu gauche) Œdème maculaire diabétique (EMD) avec liquide intrarétinien associé à l'épaississement de la rétine (flèches). (Au milieu à droite) Plusieurs drusen (pointes de flèches) présentes au début de la DMLA. (Extrême droite) Rétine normale avec contour fovéal préservé et absence de liquide rétinien / œdème. Nous utiliserons un ensemble de données des OCT mis sur **kaggle**. [83]

II.1.1.1 L'analyse des données

Après avoir analysé les données et à partir de l'image de la Figure 2, nous pouvons clairement dire qu'il s'agit d'un problème de jeu de données déséquilibré. Ici, les classes DME et DRUSEN ont très moins d'images.

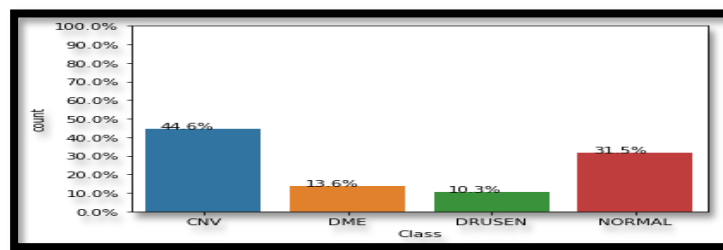


Figure 45. Distribution de classes de l'ensemble de données

Comme nous n'avons pas les ressources et le temps souhaités pour former 84k images. Donc, nous prenons un échantillon de 20000 images (Train) ,968 images (Val) et 32 images (Test) avec presque la même distribution de classe que dans le DataSet entières (**Nous avons 45%(CNV) ,30%(Normal) ,15%(DME) et10%(DRUSEN)**).

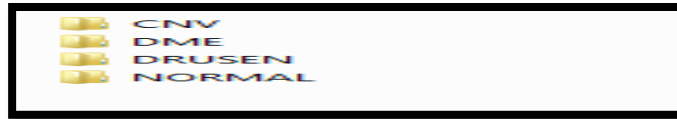


Figure 46: Structure de répertoire.

Le Dataset est composé de :

- Train (Contient 4 dossiers « classes » : CNV, NORMAL, DME, DRUSEN).
- val (Contient 4 dossiers « classes » : CNV, NORMAL, DME, DRUSEN).
- Test (Contient 4 dossiers « classes » : CNV, NORMAL, DME, DRUSEN).

II.1.1.2 Prétraitement et visualisation des données

Nous devons visualiser nos données images et prétraiter les images avant de les transmettre aux modèles CNN. Nous avons utilisé la technique de l'augmentation de données. **L'augmentation des données augmente artificiellement la taille de l'ensemble d'entraînement** en générant une nouvelle variante de chaque instance d'entraînement. Keras offre la possibilité d'ajuster des modèles à l'aide de l'augmentation des données d'image via la classe *ImageDataGenerator*.

II.1.2 Implémentation et Mise en œuvre

Afin de montrer les architectures pour les modèles CNN proposées dans le chapitre précédant, on illustre dans ce qui suit les résultats obtenus en termes de précision et d'erreur ainsi que les matrices de confusion par rapport au nombre d'époques (Une époque décrit le nombre de fois que l'algorithme passe sur l'ensemble de données) pour chaque modèle.

II.1.2.1 Mesures et performances

Maintenant que notre architecture est prête, avant de la mettre en œuvre, nous devons définir les paramètres qui nous permettront de mesurer les résultats de notre travail et de le comparer à d'autres travaux. Pour ce faire, nous devons introduire ce qu'on appelle la matrice de confusion et les différentes variables que nous utiliserons pour calculer toutes nos mesures de rendement.

- **Précision**

La précision est le rapport entre les observations positives correctement prédites et le total des observations positives prédites

$$\text{Précision} = \text{TP} / \text{TP} + \text{FP} \text{-----} 13$$

- **Recall**

Le rappel est le rapport entre les observations positives correctement prédites et toutes les observations de la classe réelle

$$\text{Recall} = \text{TP} / \text{TP} + \text{FN} \text{-----} 14$$

- **F1-Score**

La moyenne pondérée de la précision et du rappel. Par conséquent, ce score prend en compte à la fois les faux positifs et les faux négatifs.

$$\text{Score F1} = 2 * (\text{Rappel} * \text{Précision}) / (\text{Rappel} + \text{Précision}) \text{---} 15$$

- **Accuracy**

Ratio d'observations correctement prédites par rapport au nombre total d'observations.

$$\text{Accuracy} = \text{TP} + \text{TN} / \text{TP} + \text{FP} + \text{FN} + \text{TN} \text{-----} 16$$

II.1.2.2 Evaluation des performances

Chaque modèle a été appliqué sur l'ensemble de données mentionnée précédemment. L'évaluation de nos modèles s'est faite en termes de métriques de performances. Tous les résultats obtenus sont résumés sous forme de graphes. Après plusieurs essais en modifiant dans les valeurs Epoch et Batch size, Nous avons eu les meilleurs résultats avec Epoch = 20, Batch size = 64.

❖ **CNN Trois Couches**

- Accuracy&Loss :

```
31/31 [=====] - 4s 120ms/step - loss: 0.2782 - accuracy: 0.9019
Test Loss: 0.2782209813594818
Test accuracy: 0.9018595218658447
```

Figure 47: Test et accuracy du modèle CNN 3 couches

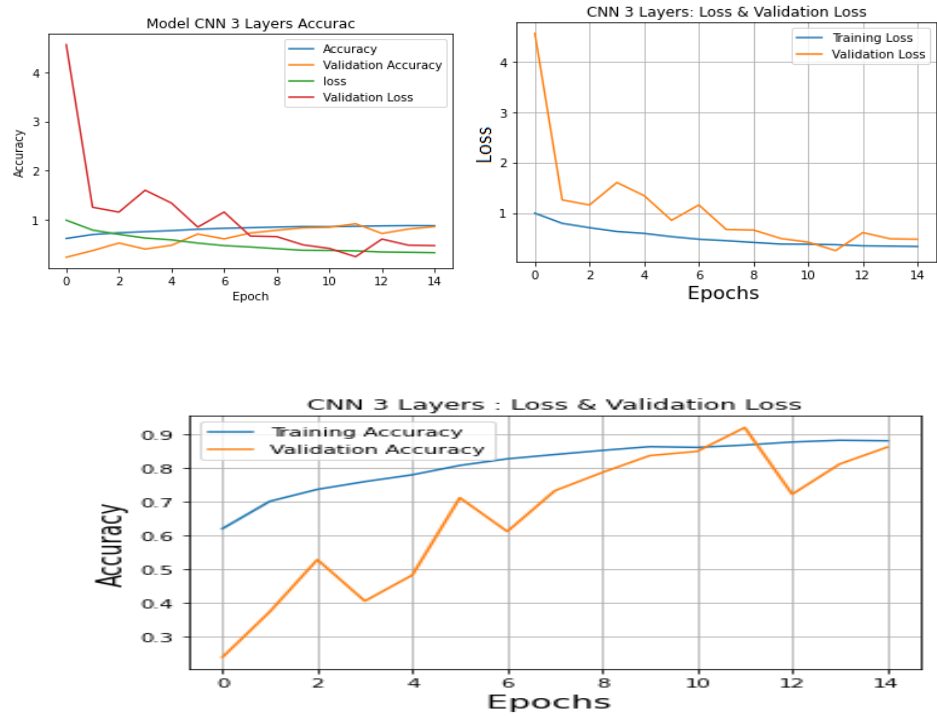


Figure 48 : Représentations graphiques de précision et de perte CNN 3 couches

- M
a
t
r
i
c
e
d
e
C
onfusion :

```

Test Confusion Matrix CNN 3 Layers
[[218  21  2  1]
 [ 19 201  1 21]
 [  6  9 222  5]
 [  0 10  0 232]]
Test Classification Report CNN 3 Layers
              precision    recall  f1-score   support

   Cnv         0.90         0.90         0.90         242
   DME         0.83         0.83         0.83         242
  DRUSEN       0.99         0.92         0.95         242
   NORMAL      0.90         0.96         0.93         242

 accuracy         0.90
macro avg         0.90         0.90         0.90         968
weighted avg         0.90         0.90         0.90         968

```

Figure 49 : Matrice de confusion CNN 3 Couches

- Observations :

1. Nous avons 3 couches cachées dans ce modèle CNN.

2. Test Loss: 0, 2782.
3. Test Accuracy: 0, 9018.
4. Precision = 0, 90 || Recall (rappel) = 0,90 || f1-score = 0,90.

❖ CNN Cinq Couches

▪ Accuracy&Loss

```

31/31 [=====] - 5s 170ms/step - loss: 0.1709 - accuracy: 0.9390
Test Loss: 0.17090530693531036
Test accuracy: 0.9390496015548706

```

Figure 50 : Test et Accuracy CNN 5 couches

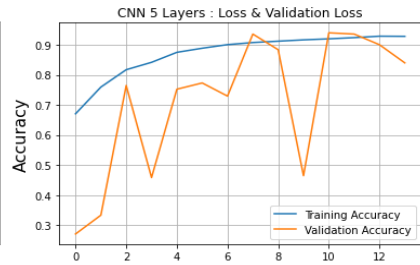
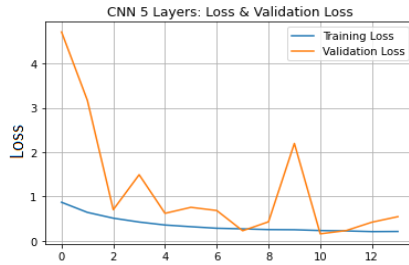
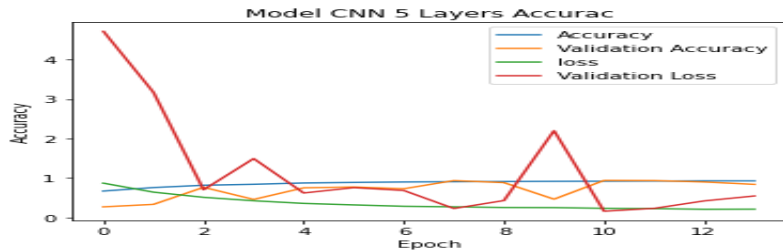


Figure 51 : Représentations graphiques de précision et de perte CNN 5 couches

▪ Matrice de confusion

```

Test Confusion Matrix Retino_CNN 5 Layers
[[226  3 11  2]
 [ 10 200  0 32]
 [  0  0 241  1]
 [  0  0  0 242]]
Test Classification Report Retino_CNN 5 Layers
              precision    recall  f1-score   support

   Cnv         0.96     0.93     0.95         242
   DME         0.99     0.83     0.90         242
  DRUSEN        0.96     1.00     0.98         242
  NORMAL        0.87     1.00     0.93         242

 accuracy          0.94         968
 macro avg         0.94     0.94     0.94         968
 weighted avg      0.94     0.94     0.94         968

```

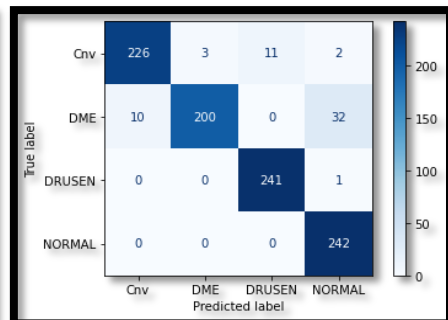


Figure 52 : Matrice de Confusion CNN 5 Couches

▪ Observations :

1. Nous avons 5 couches cachées dans ce modèle CNN.
2. Test Loss: 0, 1709
3. Test Accuracy: 0, 9390
4. Precision = 0, 94 || Recall (rappel) = 0, 94 || f1-score = 0,94

❖ **CNN Sept Couches**

▪ Accuracy&Loss :

```
31/31 [=====] - 4s 135ms/step - loss: 0.1936 - accuracy: 0.9329
Test Loss: 0.1936475932598114
Test accuracy: 0.932851254940033
```

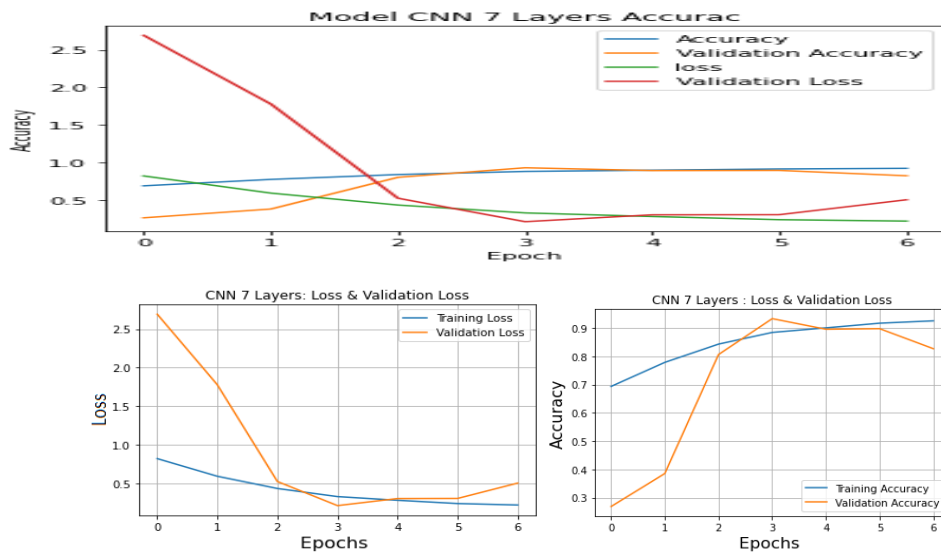


Figure 54 : Représentations graphiques de précision et de perte CNN 7 couches.

▪ Matrice de confusion

```
Test Confusion Matrix Retino_CNN 7 Layers
[[240  2  0  0]
 [ 20 216  0  6]
 [  4  0 238  0]
 [  0  4  29 209]]

Test Classification Report Retino_CNN 7 Layers
precision  recall  f1-score  support
Cnv        0.91    0.99    0.95    242
DME        0.97    0.89    0.93    242
DRUSEN     0.89    0.98    0.94    242
NORMAL     0.97    0.86    0.91    242

accuracy   0.93
macro avg  0.94    0.93    0.93    968
weighted avg 0.94    0.93    0.93    968
```

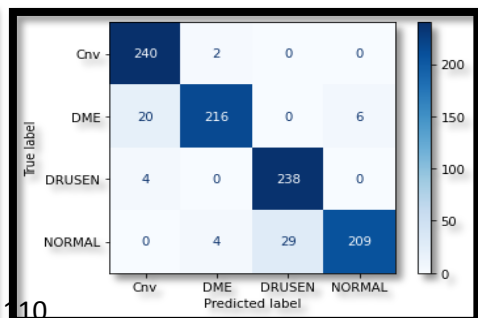


Figure 55 : Matrice de confusion CNN 7 Couches

- Observations :
 1. Nous avons 5 couches cachées dans ce modèle CNN.
 2. Test Loss: 0, 1936
 3. Test Accuracy: 0, 9328
 4. Precision = 0, 94 || Recall (rappel) = 0, 93 || f1-score = 0, 93

❖ **Apprentissage par transfert VGG16**

- Accuracy&Loss :

```

31/31 [=====] - 5s 151ms/step - loss: 0.2188 - accuracy: 0.9298
Test Loss: 0.21884028613567352
Test accuracy: 0.9297520518302917
    
```

Figure 56 : Test et Accuracy VGG16

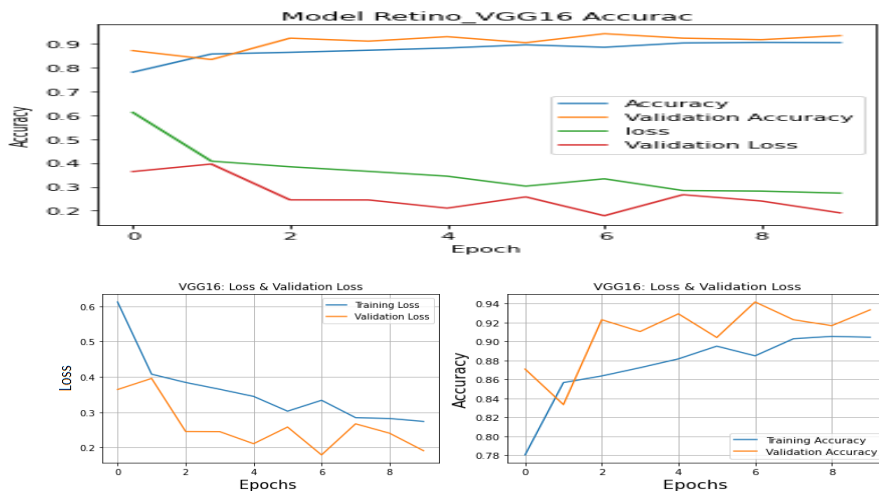


Figure 57: Représentations graphiques de précision et de perte VGG16

- Matrice de confusion

```

Test Confusion Matrix Retino_VGG16
[[221  9 12  0]
 [ 5 209  7 21]
 [ 7  0 229  6]
 [ 0  0  1 241]]
Test Classification Report Retino_VGG16
precision recall f1-score support
 Cnv      0.95  0.91  0.93   242
 DME      0.96  0.86  0.91   242
 DRUSEN   0.92  0.95  0.93   242
 NORMAL   0.90  1.00  0.95   242
accuracy          0.93   968
macro avg         0.93  0.93  0.93   968
weighted avg     0.93  0.93  0.93   968
    
```

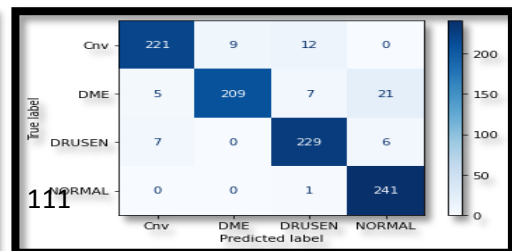


Figure 58 : Matrice de Confusion VGG16

- Observations :
 1. Test Loss: 0, 2188
 2. Test Accuracy: 0, 9297
 3. Precision = 0, 93 || Recall (rappel) = 0, 93 || f1-score = 0,93.

▪ **Apprentissage par Transfert ResNet50**

Accuracy&Loss :

```
31/31 [=====] - 5s 155ms/step - loss: 0.0527 - accuracy: 0.9814
Test Loss: 0.05273180082440376
Test accuracy: 0.9814049601554871
```

Figure 59: Test et Accuracy ResNet50

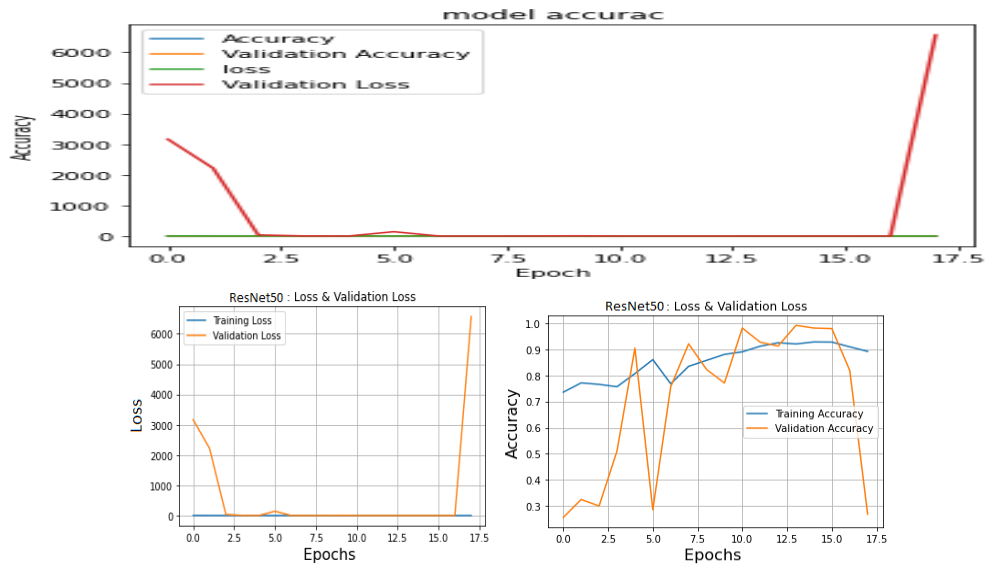


Figure 60: Représentations graphiques de précision et de perte ResNet50

▪ Matrice de confusion

```
Test Confusion Matrix
[[228  5  9  0]
 [ 1 239  1  1]
 [ 1  0 241  0]
 [ 0  0  0 242]]
Test Classification Report
precision  recall  f1-score  support
Cnv       0.99    0.94    0.97    242
DME       0.98    0.99    0.98    242
DRUSEN    0.96    1.00    0.98    242
NORMAL    1.00    1.00    1.00    242
accuracy          0.98
macro avg         0.98
weighted avg      0.98
```

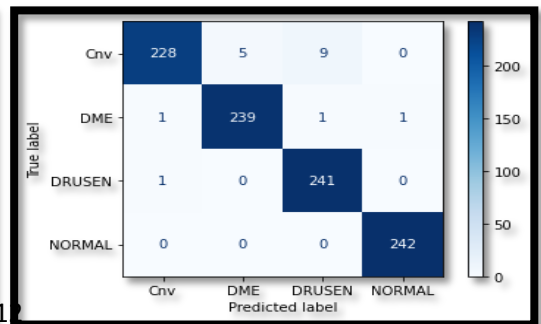


Figure 61: Matrice de Confusion ResNet50

- Observations:
 1. Test Loss: 0, 0527
 2. Test Accuracy: 0, 9814
 3. Precision = 0, 98 || Recall (rappel) = 0, 98 || f1-score = 0,98

❖ **Apprentissage par transfert MobileNet**

- Accuracy&Loss :

```
31/31 [=====] - 5s 161ms/step - loss: 0.1334 - accuracy: 0.9597
Test Loss: 0.1334446370601654
Test accuracy: 0.9597107172012329
```

Figure 62: Test et Accuracy Mobile Net (Module1)

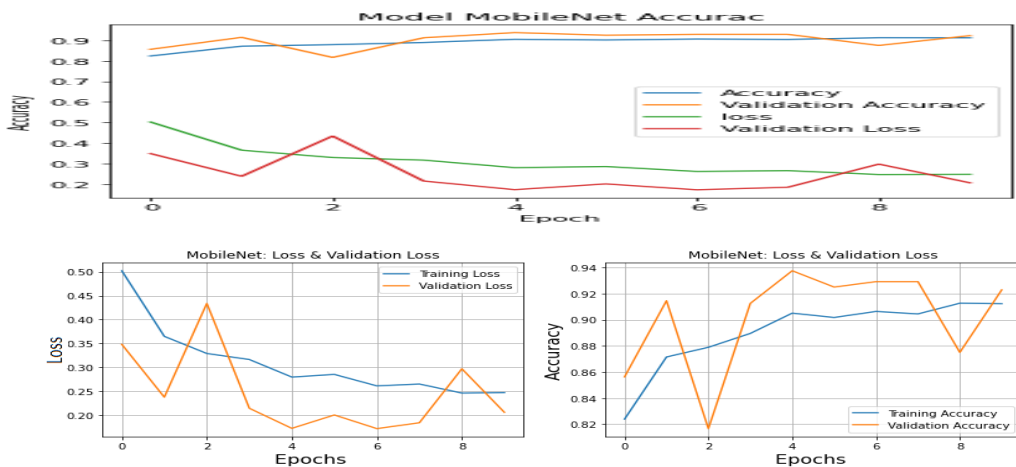


Figure 63 : Représentations graphiques de précision et de perte MobileNet(Module1)

- Matrice de confusion

```
Test Confusion Matrix MobileNet
[[230  5  7  0]
 [ 7 233  0  2]
 [ 12  2 225  3]
 [ 0  0  1 241]]
Test Classification Report MobileNet
precision recall f1-score support
   Cnv      0.92    0.95    0.94    242
   DME      0.97    0.96    0.97    242
  DRUSEN    0.97    0.93    0.95    242
  NORMAL    0.98    1.00    0.99    242
 accuracy              0.96    968
 macro avg             0.96    0.96    968
 weighted avg          0.96    0.96    968
```

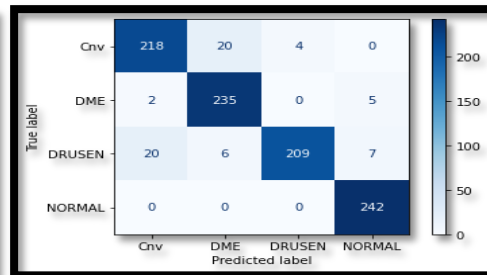


Figure 64 : Matrice de Confusion MobileNet (Module1).

- Observations:
 1. Test Loss: 0, 1334
 2. Test Accuracy: 0, 9597
 3. Precision = 0, 96 || Recall (rappel) = 0, 96 || f1-score = 0,96

❖ **Apprentissage par transfert DenseNet121**

- Accuracy&Loss :

```

31/31 [=====] - 4s 128ms/step - loss: 0.0995 - accuracy: 0.9752
Test Loss: 0.09945472329854965
Test accuracy: 0.9752066135406494
    
```

Figure 23: Test et Accuracy DenseNet121

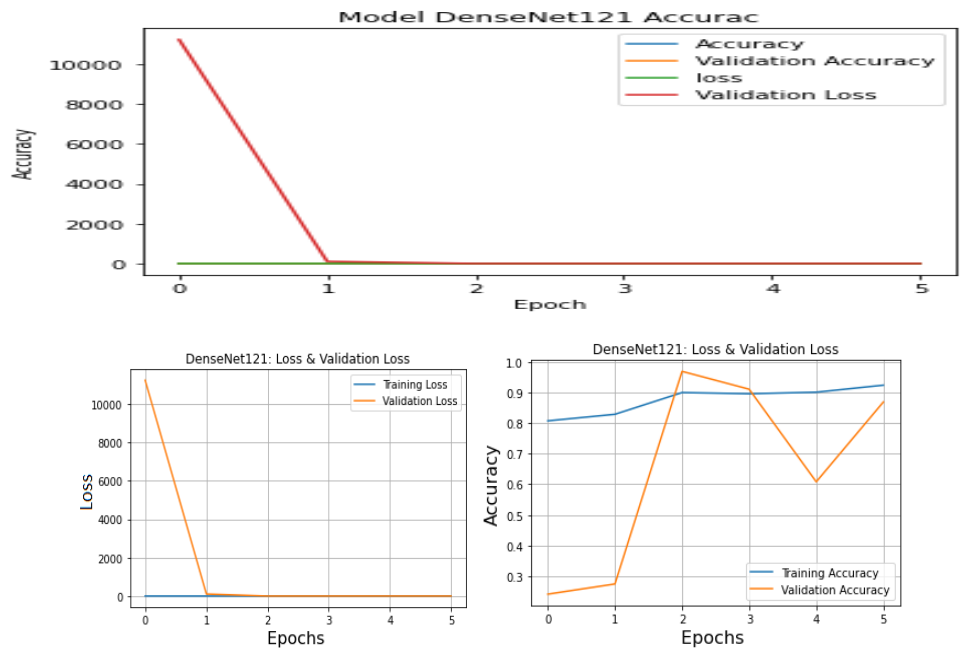


Figure 66 : Représentations graphiques de précision et de perte DenseNet121

- Matrice de confusion

```

Test Confusion Matrix DenseNet121
[[239  3  0  0]
 [  2 240  0  0]
 [ 19  0 223  0]
 [  0  0  0 242]]
Test Classification Report DenseNet121
precision recall f1-score support
Cnv      0.92   0.99   0.95   242
DME      0.99   0.99   0.99   242
DRUSEN   1.00   0.92   0.96   242
NORMAL   1.00   1.00   1.00   242
accuracy macro avg   0.98   0.98   0.98   968
weighted avg  0.98   0.98   0.98   968
    
```

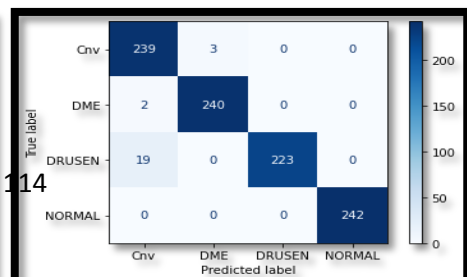


Figure 67 : Matrice de Confusion DenseNet121

- Observations :
 1. Test Loss: 0, 0994
 2. Test Accuracy: 0, 9752
 3. Precision = 0, 98 || Recall (rappel) = 0, 98 || f1-score = 0,98.

II.1.3 Comparaison des performances des modèles

Modèles CNN	Test Accuracy	Test Loss	Test F1-Score
CNN 3 Couches	0.9018	0.2782	0.90
CNN 5 Couches	0.9390	0.1709	0.94
CNN 7 Couches	0.9328	0.1936	0.93
VGG16	0.9297	0.2188	0.93
ResNet50	0.9814	0.0527	0.98
MobileNet	0.9597	0.1334	0.97
DenseNet121	0.9752	0.0994	0.98

Figure68 : Comparaison des performances des modèles

- Le modèle d'apprentissage par transfert ResNet50 et DenseNet121 ont obtenu de meilleurs résultats que les autres modèles.
- ResNet50 est le modèle le plus performant parmi les 7 modèles ci-dessus. ResNet50 Transfer Learning a le score moyen f1-micro le plus élevé, soit 0,981.
- Nous pouvons également améliorer le score f1-micro si nous entraînons le modèle avec des images entières de 84K.

II.1.4 Phase de test

Le vrai test pour savoir si le réseau neuronal fonctionne correctement est de savoir si le modèle est capable de prédire les données de test, c'est-à-dire de nouvelles données qui ne sont pas vues par le modèle. Donc la phase finale de notre travail consiste à tester des images. Les blocs de cellules ci-dessous accompliront cela et les résultats obtenus.

```
retinal_dict = {"[0]": "CNV", "[1]": "DME", "[2]": "DRUSEN", "[3]": "NORMAL"}

retinal_dict_n = {"CNV": "CNV ", "DME": "DME", "DRUSEN": "DRUSEN", "NORMAL": "NORMAL"}
```

```
from google.colab.patches import cv2_imshow
def draw_test(name, pred, im):
    retinal = retinal_dict[str(pred)]
    BLACK = [0,0,0]
    expanded_image = cv2.copyMakeBorder(im, 80, 0, 0, 100 ,cv2.BORDER_CONSTANT,value=BLACK)
    cv2.putText(expanded_image, retinal, (20, 60) , cv2.FONT_HERSHEY_SIMPLEX,1, (0,0,255), 2)
def getRandomImage(path):
    folders = list(filter(lambda x: os.path.isdir(os.path.join(path, x)), os.listdir(path)))
    random_directory = np.random.randint(0,len(folders))
    path_class = folders[random_directory]
    print("Class - " + retinal_dict_n[str(path_class)])
    file_path = path + path_class
    file_names = [f for f in listdir(file_path) if isfile(join(file_path, f))]
    random_file_index = np.random.randint(0,len(file_names))
    image_name = file_names[random_file_index]
    print("image_name:",image_name)

    return cv2.imread(file_path+"/"+image_name)
```

```
for i in range(0,10):
    input_im = getRandomImage("/content/drive/My Drive/Mon Projet/kaggle/DataSet/test/")
    input_original = input_im.copy()
    input_original = cv2.resize(input_original, None, fx=0.5, fy=0.5, interpolation = cv2.INTER_LINEAR)

    input_im = cv2.resize(input_im, (224, 224), interpolation = cv2.INTER_LINEAR)
    input_im = input_im / 255.
    input_im = input_im.reshape(1,224,224,3)

    # Get Prediction
    res = np.argmax(classifier.predict(input_im, 1, verbose = 0), axis=1)
    print("resultat :",res)

    # Show image with predicted class
    draw_test("Prediction", res, input_original)
    cv2.waitKey(0)

cv2.destroyAllWindows()
```

```
Class - NORMAL
image_name: NORMAL-5193994-1.jpeg
resultat : [3]
Class - CNV
image_name: CNV-8184974-1.jpeg
resultat : [0]
Class - DME
image_name: DME-9925591-1.jpeg
resultat : [1]
Class - NORMAL
image_name: NORMAL-5171640-1.jpeg
resultat : [3]
Class - NORMAL
image_name: NORMAL-5156112-1.jpeg
resultat : [3]
Class - DRUSEN
image_name: DRUSEN-9894035-2.jpeg
resultat : [2]
Class - DME
image_name: .DS_Store
```

Figure 69 : représente La phase de prédiction(Module1)
Tous les sept modèles CNN entraînés nous a données une correcte prédiction.

II.2 Module 2

Dans ce module nous allons atteindre notre objectif majeur, Donc Nous allons construire un classificateur qui prédira une classe parmi les quatre classes (Normal, Glaucome, Rétinopathie Diabétique, DMLA). Comme nous n'avons pas un ensemble de jeu de données assez grande et prétraitée, Nous avons proposé l'approche de réutiliser les modèles pré-entraînés du module 1 pour obtenir une correcte prédiction.

II.2.1 Jeux de données

Après une grande recherche, Nous avons fusionné plusieurs ensembles de jeu de données pour créer notre DataSet qui atteindre notre objectif.

- **DataSet 1 :**

La reconnaissance intelligente des maladies oculaires (ODIR) est une base de données ophtalmique structurée de 5 000 patients âgés, des photographies en couleur du fond d'œil gauche et droit et des mots-clés de diagnostic des médecins [10].

Cet ensemble de données est destiné à représenter un ensemble d'informations «réelles» sur les patients collectées par ShangongMedicalTechnology Co., Ltd. Auprès de différents hôpitaux / centres médicaux en Chine. Dans ces institutions, les images du fond de l'œil sont capturées par diverses caméras du marché, telles que Canon, Zeiss et Kowa, ce qui donne des résolutions d'image variées.

Les annotations ont été étiquetées par des lecteurs humains formés avec une gestion du contrôle de la qualité. Ils classent le patient en huit étiquettes, notamment :

- Normal (N), Diabète (D), Glaucome (G), Cataracte ,Dégénérescence maculaire liée à l'âge (A), Hypertension (H), Myopie pathologique (M), Autres maladies / anomalies (O).

- **DataSet 2 :**

La base de données ORIGA-light contient 650 rétines annotées images. Chaque image est étiquetée avec des informations de classement. Elle contient de dossier glaucome positif et glaucome négatif[11].

- **DataSet 3 :**

Le glaucome est un groupe d'affections oculaires qui endommagent le nerf optique, dont la santé est vitale pour une bonne vision. Ces dommages sont souvent causés par une pression anormalement élevée dans votre œil. Le glaucome est l'une des principales causes de cécité chez les personnes de plus de 60 ans.

Cet ensemble de données contient des images / octets de numérisation de l'œil. [12]

II.2.1.1 Procédure de traitement de l'ensemble de jeu de données

II.2.1.2 Extraction des images de glaucome, DMLA, rétinopathie diabétique et normal.

Dans cette phase, Nous avons extraire les images à partir d'un fichier CSV de DataSet1. La figure ci-dessous représente un exemple d'un extrait de code source.

```
▶ counter=0
for(i,row)in df.iterrows():
    if row['Right-Diagnostic Keywords']=='glaucoma':
        filename=row["filename"]
        img_p_1=os.path.join(images_path,filename)
        img_cp_p_1=os.path.join(target_fold,filename)
        shutil.copy2(img_p_1,img_cp_p_1)
        print("img copy",counter)
        counter+=1
    print(counter)
```

Figure 70 : Exemple de glaucome.

1. Fusionnement des images glaucome positif de DataSet2 et 3.
2. Division de l'ensemble de jeu construit en trois ensembles train, validation et test.la distribution de l'ensemble de jeu est comme suit : 80% pour train ,18% pour validation et 2% pour le test. La figure ci-dessous représente le code source de la distribution.

```
database="/content/drive/MyDrive/Mon Projet/DatasetN"
splitfolders.ratio(database,output="/content/drive/MyDrive/Mon Projet/Patho_Ophtalmo_DataSet/DataSetP/",seed=1337,ratio=[0.8,0.18,0.02])
Copying files: 3146 files [54:57, 1.05s/ files]
```

Figure 71: la distribution de jeu de données (train, val, test)

L'ensemble de jeu de données obtenu contient 3146 images de fond d'œil,2515 **images (Train)**, 564 **images (Val)** et 67 **images (Test)**.

II.2.1.3 Prétraitement et Visualisation de données

Nous devons visualiser nos données images fond d'oeil et prétraiter les images avant de les transmettre aux modèles CNN. Nous avons utilisé la technique de l'augmentation de données. Cette technique est bien expliquée dans le Module 1.

❖ Visualisation de données

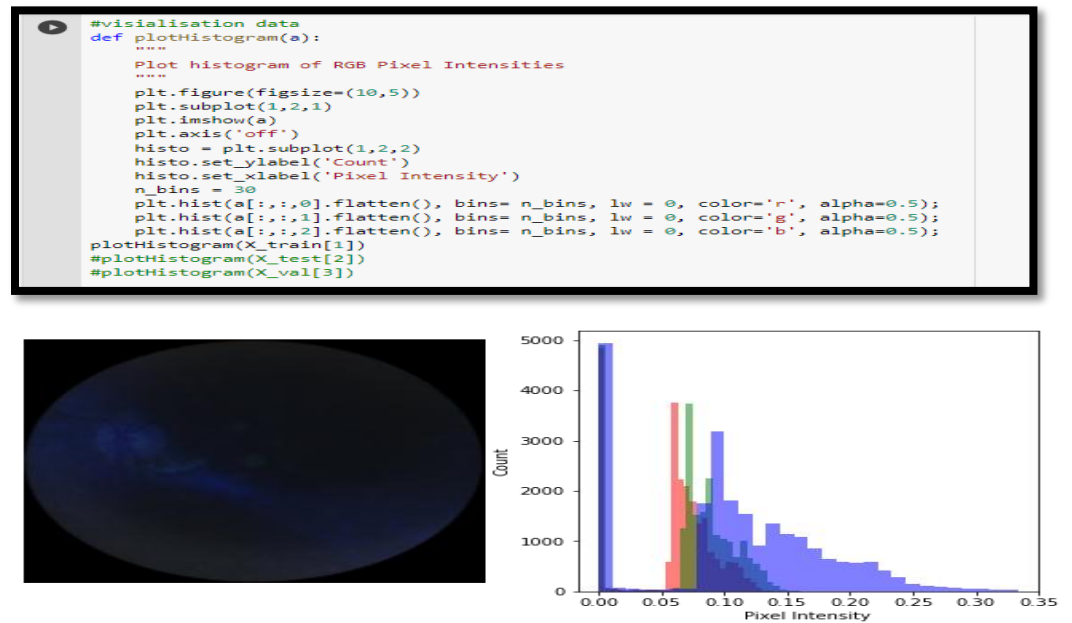


Figure 72: représentation de `X_train`

❖ Visualisation de 20 images de fond d'œil glaucomes de l'ensemble d'entraînement.

```
print("Glaucoma")
imgs = glob('/content/drive/MyDrive/Mon Projet/Patho_Ophtalmo_DataSet/DataSet/train/Glaucoma/**')
i = 0
plt.rcParams['figure.figsize'] = (15.0, 15.0)
plt.subplots_adjust(wspace=0, hspace=0)
for l in imgs[:25]:
    im = cv2.imread(l)
    im = cv2.resize(im, (224, 224))
    plt.subplot(5, 5, i+1)
    plt.imshow(cv2.cvtColor(im, cv2.COLOR_BGR2RGB))
    plt.axis('off');
    i += 1
```

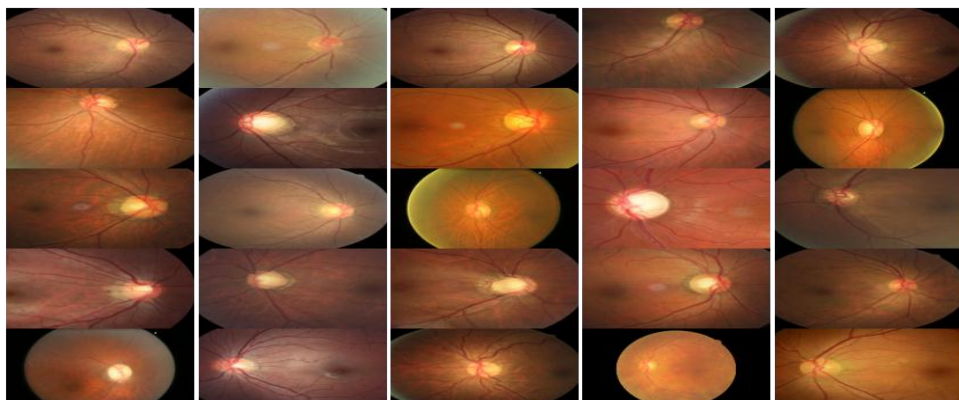


Figure 73: exemples d'image fond d'œil glaucome

II.2.2 Implémentation et Mise en œuvre

Dans ce Module, Nous avons choisis de réutiliser tous les modèles préformés dans le Module 1, Comme nous avons un petit ensemble de jeu de données et Comme nous n'avons pas les ressources et le temps souhaités pour former les modèles. Les réseaux neuronaux nécessitent généralement beaucoup de temps dans le processus de formation. Afin de réduire les coûts de calcul. Pour un concept de transfert, une fois le modèle préformé dans un domaine source contenant de grands ensembles de données, il peut être réutilisé dans d'autres domaines cibles. Après la formation du modèle avec les ensembles de données du domaine cible, le temps de formation peut être réduit grâce à la réutilisation.

Nous avons entraîné tous les classifieurs de Module 1 sur ces features., et nous avons entraînée des modèles CNN à zéro pour faire une comparaison.

Le Modèle MobileNet Nous a donné un meilleur résultat d'après plusieurs tests des tous les modèle entraînés. Nous avons présenté les résultats obtenus dans ce que suit.

II.2.2.1 Evaluation des performances

❖ Retinal_Mobile Net

- Accuracy&Loss :

```
18/18 [=====] - 15s 809ms/step - loss: 0.5072 - accuracy: 0.7713
Test Loss: 0.5071824789047241
Test accuracy: 0.771276593208313
```

Figure 74: Test et Accuracy Retinal_MobileNet.

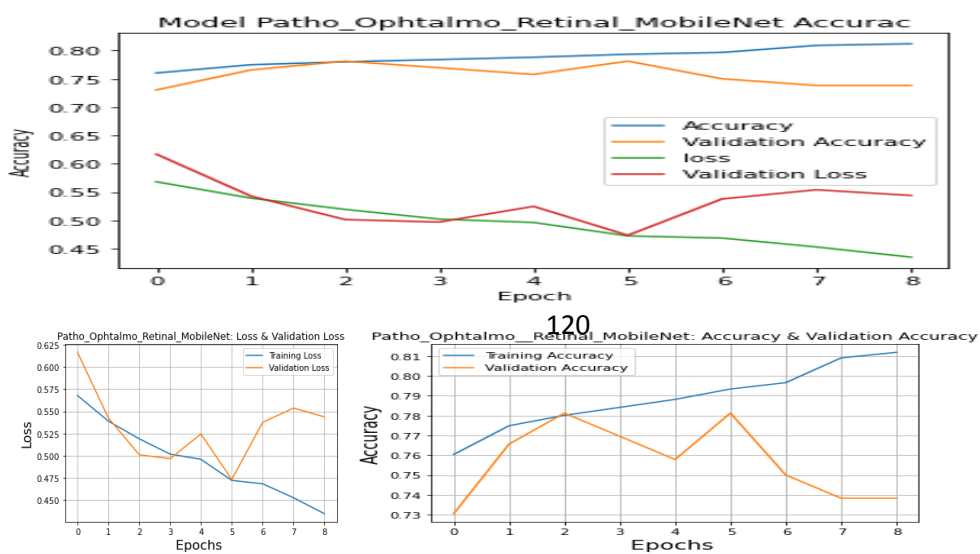


Figure 75: Représentations graphiques de précision et de perte Retinal_MobileNet

▪ Matrice de Confusion

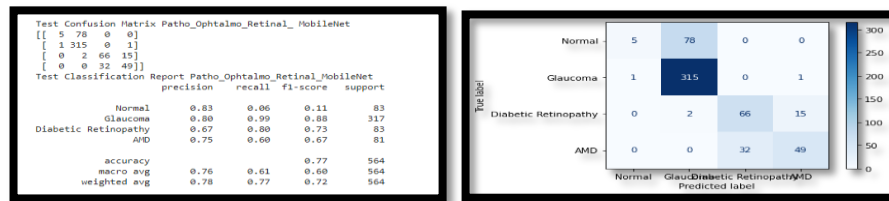


Figure 76 : Matrice de Confusion Retinal_MobileNet

▪ Observations:

1. Test Loss: 0, 5071
2. Test Accuracy: 0, 7712
3. Precision = 0, 78 || Recall (rappel) = 0, 77 || f1-score = 0,72

❖ Apprentissage par transfert MobileNet

▪ Accuracy&Loss :

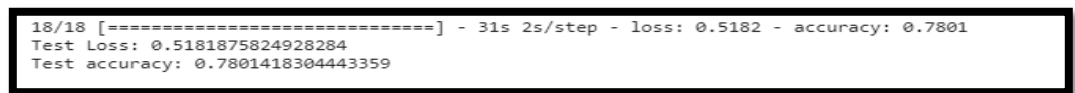


Figure 77: Test et Accuracy MobileNet (Module2)

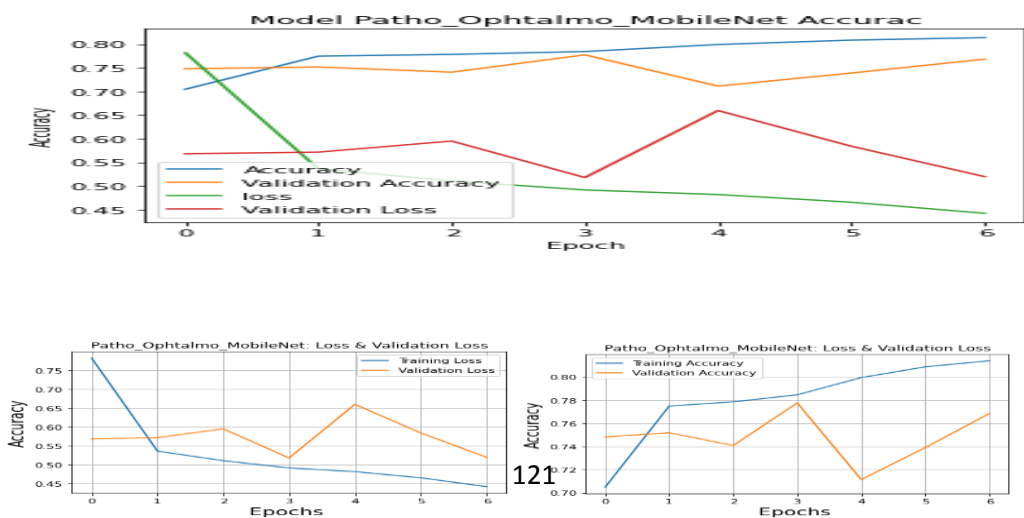


Figure 78: Représentations graphiques de précision et de perte MobileNet(Module2).

▪ Matrice de Confusion

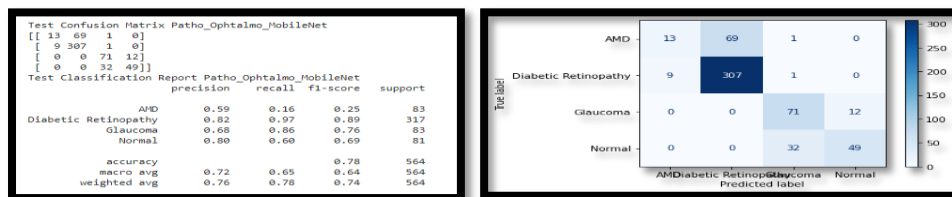


Figure 79: Matrice de Confusion MobileNet(Module2)

▪ Observations :

1. Test Loss: 0, 5181
2. Test Accuracy: 0, 7801
3. Precision = 0, 76 || Recall (rappel) = 0, 78 || f1-score = 0, 74

II.2.2.2 Comparaison des performances des modèles

Modèles CNN	Test Accuracy	Test Loss	Test F1-Score
Retinal_MobileNet	0.7712	0.5071	0.72
MobileNet	0.7801	0.5118	0.74

Tableau 7: Comparaison des performances des modèles

- Le modèle d'apprentissage par transfert MobileNet a obtenu de meilleur résultat que les autres modèles, pour cela nous avons monté leur résultat obtenu.
- Dans ce tableau, nous avons présenté une comparaison entre le modèle préformé dans le module1 que nous avons le nommé Retinal_MobileNet et le Modèle d'apprentissage par transfert MobileNet (ImageNet). Nous avons remarqué que les deux modèles ont obtenu presque le même résultat.
 - Nous pouvons également améliorer le score f1-micro si nous entraînons le modèle avec un grands dataset bien prétraité.

II.2.2.3 Phase de test

Dans ce module, nous avons obtenu des prédictions correcte et des prédictions non correctes, On peut dire 70 % des prédictions étaient correctes. La figure ci-dessous représente quelque prédiction.

```
[123] patho_ophtalmo_dict = {"[0]": "AMD",
                             "[1]": "Diabetic Retinopathy",
                             "[2]": "Glaucoma ",
                             "[3]": "Normal"}

patho_ophtalmo_dict_n = {"AMD": "AMD",
                          "Diabetic Retinopathy": "Diabetic Retinopathy",
                          "Glaucoma": "Glaucoma",
                          "Normal": "Normal"}
```

```
for i in range(0,2):
    input_im = getRandomImage("/content/drive/MyDrive/Mon Projet/Patho_Ophtalmo_Dataset/Dataset/test/")
    input_original = input_im.copy()
    input_original = cv2.resize(input_original, None, fx=0.5, fy=0.5, interpolation = cv2.INTER_LINEAR)

    input_im = cv2.resize(input_im, (224, 224), interpolation = cv2.INTER_LINEAR)
    input_im = input_im / 255.
    input_im = input_im.reshape(1,224,224,3)

    # Get Prediction
    res = np.argmax(Classifier.predict(input_im, 1, verbose = 0), axis=1)
    print("resultat :",res)
    # Show image with predicted class
    draw_test("Prediction", res, input_original)
    cv2.waitKey(0)

cv2.destroyAllWindows()

Class :Normal
image_name: NL_300.png
resultat : [3]
Class :Diabetic Retinopathy
image_name: 4557_right.jpg
resultat : [1]
```

Figure 80: représente phase de prédiction (Module2).

Conclusion

Dans ce chapitre, nous avons vu notre part de contribution au problème de classification des maladie d’ophtalmologie , représentant les outils et les jeux de données utilisé, ainsi que les étapes que nous avons suivies pour obtenir les résultats que nous montrons également pour les déférents modèles dont le but de faire la comparaison.

Conclusion générale et Perspectives

Conclusions

Nous avons présenté dans ce mémoire de fin d'étude une description d'un système de diagnostic des maladies oculaires. On a travaillé sur les maladies qui entraînent la cécité et qui nécessitent un diagnostic précoce et précis telles que la Rétinopathie Diabétique (RD), Glaucome et Dégénérescences Maculaires liées à l'âge (DMLA) utilisant l'approche Deep Learning (DL).

Dans un premier temps, nous avons mené d'une étude bibliographique sur l'anatomie de l'œil et les maladies ophtalmiques citées précédemment, leurs symptômes et le diagnostic de ces dernières. Ensuite, nous avons défini les concepts (L'apprentissage automatique ML et l'apprentissage profond DL) qui ont une relation avec notre problématique et la solution envisagée. Nous avons également présenté et classifié les travaux qui ont été effectués dans ce contexte et qui ont obtenu des résultats prometteurs.

Nous avons proposé comme contribution une approche basée sur les réseaux de neurones convolutifs (CNN). Elle consiste à construire un classifieur qui prédira une classe parmi quatre classes (Normal, Glaucome, Rétinopathie diabétique, DMLA), afin de poser un diagnostic. Nous avons d'abord préparé les données d'entraînement, notre étude s'est articulée autour de deux grands axes :

- ✓ Le premier consiste à entraîner le modèle CNN proposés (Phase d'entraînement)
- ✓ Le deuxième, consiste à faire la prédiction en utilisant le modèle proposé (Phase de prédiction)

Le Système proposé comporte deux modules :

- ❖ Module1 basé sur le transfert Learning (exemple VGG-16) sur une base de type OCT des maladies de la rétine. Nous avons utilisé le transfert learning afin de bénéficier de ses avantages dans l'optimisation du temps d'entraînement, augmenter la précision du modèle aussi à cause de l'insuffisance des données .Ce module nous a donné des résultats satisfaisants.

- ❖ Module2 basé sur l'entraînement d'un modèle CNN avec une petite base des images de type Fond d'œil des maladies (RD, Glaucome, DMLA), en utilisant le résultat du module précédent, ce module nous a donné des résultats satisfaisants et il a augmenté la précision

de prédiction du modèle.

Ce travail a été réalisé malgré les difficultés qui ont été rencontrées surtout dans la phase de collecte et préparation de données ainsi que la phase du test, concernant la base de données, l'objectif et l'idéal c'était d'avoir une base de données d'imageries médicales géante de plusieurs centres médicaux ophtalmologiques consultées et jugés par plusieurs médecins ophtalmologistes.

En effet, au départ, c'était prévu de collecter les données au niveau du service d'ophtalmologie de CHU Franz Fanon de Blida. La pandémie de Corona Virus (Covid19) nous a mis dans l'impossibilité de faire ça. Ce qui il nous a poussé à chercher des datasets, de les prétraiter et de les utiliser. D'autre part, le manque du matériel puissant nécessaire pour l'entraînement des modèles nous a fait perdre beaucoup du temps.

Perspectives

Plusieurs perspectives peuvent être envisagées afin d'enrichir ce travail :

- ✓ Le système présenté peut également être étendu pour détecter d'autres maladies liées à la RD tels que les néo vaisseaux et les nodules cotonneux et d'autres pathologies qui affectent la rétine à savoir : le glaucome, la cataracte.
- ✓ Un système de classification de la RD et la DMLA et glaucome doit être mis en œuvre afin de faciliter aux ophtalmologistes l'identification des stades de ses maladies avec précision.
- ✓ Comme les patients sont censés avoir un examen régulier de la rétine, il est important de développer une procédure de détection de changement progressif dans la rétine au fil du temps qui pourrait être intéressant pour le dépistage de la RD et la DMLA dans le long terme.

Bibliographies

- [1] F. Arcadu, F. Benmansour, A. Maunz, J. Willis, Z. Haskova, and M. Prunotto, “Deep learning algorithm predicts diabetic retinopathy progression in individual patients,” *NPJ Digit. Med.*, vol. 2, no. 1, pp. 1–9, 2019.
- [2] U. Raghavendra, H. Fujita, S. V Bhandary, A. Gudigar, J. H. Tan, and U. R. Acharya, “Deep convolution neural network for accurate diagnosis of glaucoma using digital fundus images,” *Inf. Sci. (Ny)*, vol. 441, pp. 41–49, 2018.
- [3] U. Schmidt-Erfurth *et al.*, “Machine learning to analyze the prognostic value of current imaging biomarkers in neovascular age-related macular degeneration,” *Ophthalmol. Retin.*, vol. 2, no. 1, pp. 24–30, 2018.
- [4] L. Leininger, “Anatomie de l’œil,” 2013. <https://dr-leininger.fr/oeil-et-la-vision/anatomie-de-oeil#liris-la-couleur-des-yeux> (accessed Sep. 08, 2020).
- [5] Blog, “Le Cerveau à tous les niveaux,” 2010. http://lecerveau.mcgill.ca/flash/a/a_02/a_02_cr/a_02_cr_vis/a_02_cr_vis.html (accessed Sep. 08, 2020).
- [6] B. Hafida, “Système d’analyse des images couleur du fond d’œil pour la détection des éléments principaux de la rétine Soutenu,” 2018.
- [7] B. El Mathari, “Rôle de la protéine dystrophine Dp71 dans l’inflammation vasculaire rétinienne.” 2014.
- [8] guyard, “Balades Naturalistes,” *Des cellules souches dans la rétine*, 2014. <http://baladesnaturalistes.hautetfort.com/archive/2012/07/28/des-cellules-souches-dans-la-retine.html> (accessed Sep. 08, 2020).
- [9] A. Alharthi and V. Emamian, “An Automated Mechanism for Early Screening and Diagnosis of Diabetic Retinopathy in Human Retinal Images,” *Br. J. Appl. Sci. Technol.*, vol. 12, no. 1, pp. 1–15, 2016, doi: 10.9734/bjast/2016/20702.
- [10] M. S. Maheswari and A. Punnolil, “A novel approach for retinal lesion detection in diabetic

- retinopathy images,” *People*, vol. 4, no. 6, 2014.
- [11] “Collège des Ophtalmologistes Universitaires de France (COUF) Enseignement d ’ Ophtalmologie POLYCOPIE Version 2013 Sommaire,” pp. 156–166, 2013.
- [12] P. J. P. Adenis and D. P. Y. Robert, “Compus d’auphtalmologie,” 2016. <http://umvf.omsk-osma.ru/campus-ophtalmologie/poly/01304Dico.html>.
- [13] A. Fanny, “Comparaison entre le champ visuel FDT et l’OCT du complexe ganglionnaire maculaire dans le diagnostic du glaucome pré-périmétrique.,” 2016.
- [14] “Le fond d’oeil.” <https://sites.google.com/site/amdmla/le-fond-d-oeil> (accessed Oct. 08, 2020).
- [15] F. Amoroso, “L ’ angiographie-OCT L ’,” vol. 10, 2016.
- [16] BELHOUARI Imane et BENABDELKRIM Fatima, “UN SYSTEME D’AIDE A LA DECISION MEDICALE POUR LA DETECTION PRECOCE DU GLAUCOME,” 2017.
- [17] D. Poole, A. Mackworth, and R. Goebel, “Computational Intelligence,” 1998.
- [18] T. M. Mitchell, “Machine learning and data mining,” *Commun. ACM*, vol. 42, no. 11, pp. 30–36, 1999.
- [19] A. Amraoui, W. Baghli, and B. Benmammar, “Amélioration de la fiabilité du lien sans fil pour un terminal radio cognitive mobile,” 2011.
- [20] A. Metref, “Contribution à l’étude du problème de synchronisation de porteuse dans le contexte de la Radio Intelligente.” 2010.
- [21] M. LAUR, “Anticipation des changements de notes des obligations du portefeuille d’un assureur par méthode de machine learning,” pp. 1–59, 2013.
- [22] M. Taffar, “Initiation à l’apprentissage Automatique.”
- [23] I. Goodfellow *et al.*, “Generative adversarial nets,” in *Advances in neural information processing systems*, 2014, pp. 2672–2680.
- [24] Microsoft, “Machine Learning France.” <https://docs.microsoft.com/fr-fr/archive/blogs/mlfrance/une->

premiere-introduction-au-deep-learning (accessed Sep. 10, 2020).

- [25] V. M. Pradhan, J. Vala, and P. Balani, “A survey on sentiment analysis algorithms for opinion mining,” *Int. J. Comput. Appl.*, vol. 133, no. 9, pp. 7–11, 2016.
- [26] N. P. Rougier, “Modeles de mémoires pour la navigation autonome.” Université Henri Poincaré-Nancy 1, 2000.
- [27] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*, vol. 1. MIT press Cambridge, 2016.
- [28] X. Zhang, J. Zou, K. He, and J. Sun, “Accelerating very deep convolutional networks for classification and detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 1943–1955, 2015.
- [29] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv Prepr. arXiv1412.6980*, 2014.
- [30] C. Szegedy *et al.*, “Going deeper with convolutions,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.
- [31] “Activation Functions in Neural Networks,” 2016. <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6> (accessed Jul. 04, 2020).
- [32] N. Buduma and N. Locascio, *Fundamentals of deep learning: Designing next-generation machine intelligence algorithms*. “O’Reilly Media, Inc.,” 2017.
- [33] V. Zocca, G. Spacagna, D. Slater, and P. Roelants, *Python deep learning*. Packt Publishing Ltd, 2017.
- [34] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.
- [35] U. E. T. De, “Analyse d’images médicales, application à la catégorisation des types de tuberculose pulmonaire,” *Master en Inform.*, 2018.
- [36] “RNNoise: Learning Noise Suppression,” 2017. <https://jmvalin.ca/demo/rnnoise/>.
- [37] I. Vasilev, D. Slater, G. Spacagna, P. Roelants, and V. Zocca, *Python Deep Learning: Exploring*

deep learning techniques and neural network architectures with Pytorch, Keras, and TensorFlow. Packt Publishing Ltd, 2019.

- [38] A. Y. Ng and M. I. Jordan, “On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. Advances in NeuralInformation Processing Systems 14.” MIT Press Cambridge, 2002.
- [39] D. S. W. Ting *et al.*, “Artificial intelligence and deep learning in ophthalmology,” *Br. J. Ophthalmol.*, vol. 103, no. 2, pp. 167–175, 2019.
- [40] S. J. Kim, K. J. Cho, and S. Oh, “Development of machine learning models for diagnosis of glaucoma,” *PLoS One*, vol. 12, no. 5, p. e0177726, 2017.
- [41] H.-Y. Tsao, P.-Y. Chan, and E. C.-Y. Su, “Predicting diabetic retinopathy and identifying interpretable biomedical features using machine learning algorithms,” *BMC Bioinformatics*, vol. 19, no. 9, p. 283, 2018.
- [42] H. Bogunović *et al.*, “Prediction of anti-VEGF treatment requirements in neovascular AMD using a machine learning approach,” *Invest. Ophthalmol. Vis. Sci.*, vol. 58, no. 7, pp. 3240–3248, 2017.
- [43] E. V Carrera, A. González, and R. Carrera, “Automated detection of diabetic retinopathy using SVM,” in *2017 IEEE XXIV International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, 2017, pp. 1–4.
- [44] A. A. Salam, T. Khalil, M. U. Akram, A. Jameel, and I. Basit, “Automated detection of glaucoma using structural and non structural features,” *Springerplus*, vol. 5, no. 1, p. 1519, 2016.
- [45] D. Darmawan, ~~濟無~~*No Title No Title*, vol. 53, no. 9. 2019.
- [46] Y. Tong, W. Lu, Y. Yu, and Y. Shen, “Application of machine learning in ophthalmic imaging modalities,” *Eye Vis.*, vol. 7, pp. 1–15, 2020.
- [47] S. Phan *et al.*, “Evaluation of deep convolutional neural networks for glaucoma detection,” *Jpn. J. Ophthalmol.*, vol. 63, no. 3, pp. 276–283, 2019, doi: 10.1007/s10384-019-00659-6.
- [48] J. J. Gómez-Valverde *et al.*, “Automatic glaucoma classification using color fundus images based on convolutional neural networks and transfer learning,” *Biomed. Opt. Express*, vol. 10, no. 2, pp. 892–913, 2019.
- [49] T. Nagasawa *et al.*, “Accuracy of ultrawide-field fundus ophthalmoscopy-assisted deep learning

- for detecting treatment-naïve proliferative diabetic retinopathy,” *Int. Ophthalmol.*, vol. 39, no. 10, pp. 2153–2159, 2019, doi: 10.1007/s10792-019-01074-z.
- [50] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, “A survey on deep transfer learning,” in *International conference on artificial neural networks*, 2018, pp. 270–279.
- [51] P. M. Burlina, N. Joshi, K. D. Pacheco, T. Y. A. Liu, and N. M. Bressler, “Assessment of Deep Generative Models for High-Resolution Synthetic Retinal Image Generation of Age-Related Macular Degeneration,” *JAMA Ophthalmol.*, vol. 137, no. 3, pp. 258–264, Mar. 2019, doi: 10.1001/jamaophthalmol.2018.6156.
- [52] Y. Peng *et al.*, “DeepSeeNet: a deep learning model for automated classification of patient-based age-related macular degeneration severity from color fundus photographs,” *Ophthalmology*, vol. 126, no. 4, pp. 565–575, 2019.
- [53] H. Muhammad *et al.*, “Hybrid deep learning on single wide-field optical coherence tomography scans accurately classifies glaucoma suspects,” *J. Glaucoma*, vol. 26, no. 12, p. 1086, 2017.
- [54] R. Gargeya and T. Leng, “Automated identification of diabetic retinopathy using deep learning,” *Ophthalmology*, vol. 124, no. 7, pp. 962–969, 2017.
- [55] P. M. Burlina, N. Joshi, M. Pekala, K. D. Pacheco, D. E. Freund, and N. M. Bressler, “Automated grading of age-related macular degeneration from color fundus images using deep convolutional neural networks,” *JAMA Ophthalmol.*, vol. 135, no. 11, pp. 1170–1176, 2017.
- [56] M. D. Abràmoff *et al.*, “Improved Automated Detection of Diabetic Retinopathy on a Publicly Available Dataset Through Integration of Deep Learning,” *Invest. Ophthalmol. Vis. Sci.*, vol. 57, no. 13, pp. 5200–5206, Oct. 2016, doi: 10.1167/iovs.16-19964.
- [57] Q. Abbas, “Glaucoma-Deep : Detection of Glaucoma Eye Disease on Retinal Fundus Images using Deep Learning,” vol. 8, no. 6, pp. 41–45, 2020.
- [58] T. R. Gadekallu *et al.*, “Early detection of diabetic retinopathy using PCA-firefly based deep learning model,” *Electronics*, vol. 9, no. 2, p. 274, 2020.
- [59] J. P. Campbell *et al.*, “Automated diagnosis of plus disease in retinopathy of prematurity using deep learning,” *J. Am. Assoc. Pediatr. Ophthalmol. Strabismus {JAAPOS}*, vol. 22, no. 4, p. e12, 2018.
- [60] “PLOS ONE,” *Application of convolutional neural networks for classification of adult mosquitoes in the field.* <https://doi.org/10.1371/journal.pone.0210829.g001> (accessed Nov. 02, 2020).

- [61] “ResearchGate,” *Classification phase by CNN*, 2008.
https://www.researchgate.net/figure/Classification-phase-by-CNN_fig3_333183634 (accessed Nov. 02, 2020).
- [62] “wildml.” <http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/> (accessed Nov. 02, 2020).
- [63] Prabhu, “Understanding Hyperparameters and its Optimisation techniques.”
<https://towardsdatascience.com> (accessed Nov. 03, 2020).
- [64] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *arXiv Prepr. arXiv1502.03167*, 2015.
- [65] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, “Efficient backprop,” in *Neural networks: Tricks of the trade*, Springer, 2012, pp. 9–48.
- [66] M. Lin, Q. Chen, and S. Yan, “Network in network,” *arXiv Prepr. arXiv1312.4400*, 2013.
- [67] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, “Deep learning. vol. 1.” MIT press Cambridge, 2016.
- [68] K. Gurney, *An introduction to neural networks*. CRC press, 1997.
- [69] A. Karpathy, “Neural Networks Part 1: Setting Up the Architecture. Notes for CS231n Convolutional Neural Networks for Visual Recognition.” Stanford University: Stanford, CA, USA, 2016.
- [70] F. Visin, K. Kastner, K. Cho, M. Matteucci, A. Courville, and Y. Bengio, “Renet: A recurrent neural network based alternative to convolutional networks,” *arXiv Prepr. arXiv1505.00393*, 2015.
- [71] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, “CNN features off-the-shelf: an astounding baseline for recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2014, pp. 806–813.
- [72] S. Shankar, V. K. Garg, and R. Cipolla, “Deep-carving: Discovering visual attributes by carving deep neural nets,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3403–3412.
- [73] B. Huval, A. Coates, and A. Ng, “Deep learning for class-generic object detection,” *arXiv Prepr. arXiv1312.6885*, 2013.

- [74] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv Prepr. arXiv1409.1556*, 2014.
- [75] “popular-network.” <https://neurohive.io/en/popular-networks/vgg16/> (accessed Nov. 06, 2020).
- [76] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [77] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.
- [78] “colab.” .
- [79] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, “Learning hierarchical features for scene labeling,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, 2012.
- [80] J. Ma, R. P. Sheridan, A. Liaw, G. E. Dahl, and V. Svetnik, “Deep neural nets as a method for quantitative structure–activity relationships,” *J. Chem. Inf. Model.*, vol. 55, no. 2, pp. 263–274, 2015.
- [81] “Interpretation of Performance Measures,” 2016. <http://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures> (accessed Nov. 06, 2020).
- [82] “pandas.pydata.org.” <http://pandas.pydata.org/pandas-docs/version/0.23/> .
- [83] “kaggle.” <https://www.kaggle.com/paultimothymooney/kermany2018> (accessed Jul. 21, 2020).

Annexe

III. Fonction d'activations

III.1 Fonction d'activation linéaire

La fonction est une ligne ou linéaire Par conséquent, la sortie des fonctions ne sera pas confinée entre n'importe quelle range.

- **Équation:** $f(x) = x$
- **Range:** $(-\infty, +\infty)$

Cela n'aide pas avec la complexité ou les divers paramètres des données habituelles qui sont introduites dans les réseaux de neurones .[31]

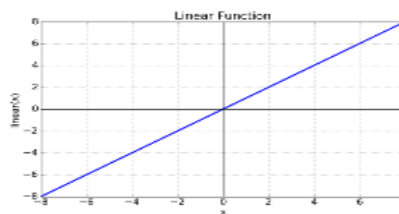


Figure 81 : Fonction d'activation linéaire.[31]

III.2 Fonction d'activation non linéaire

Les fonctions d'activation non linéaire sont les fonctions d'activation les plus utilisées. Il permet au modèle de généraliser ou de s'adapter à une variété de données et de différencier les résultats.[31]

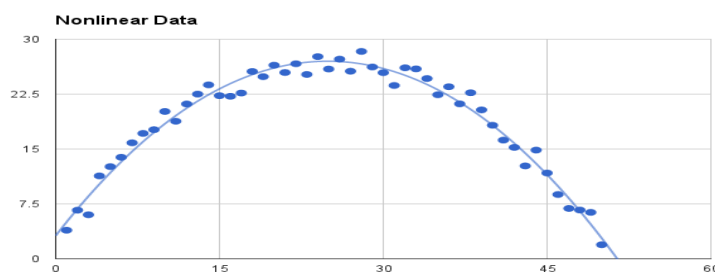


Figure 82 : Fonction d'activation non linéaire.[31]

- **La fonction Sigmoidale :** Cette fonction est l'une des plus couramment utilisées. Il est borné entre 0 et 1, et il peut être interprété stochastiquement comme la probabilité que le neurone s'active, et il est généralement appelé la fonction logistique ou le sigmoïde logistique.[32]

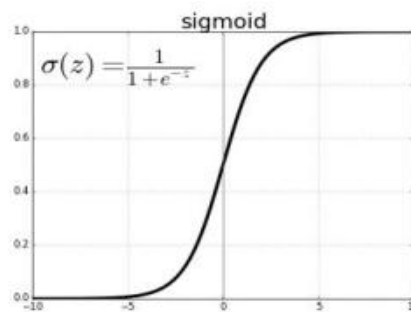


Figure 83 : Représentation graphique de la fonction Sigmoïde.[32]

- **La fonction ReLu** : La fonction RELU est probablement la plus proche de sa correspondante biologique [33]. Cette fonction est récemment devenue le choix de nombreuses tâches (notamment en computer vision) Comme dans la formule ci-dessus, cette fonction renvoie 0 si l'entrée z est inférieure à 0 et retourne z lui-même si il est plus grande que 0[33]
- **La fonction Softmax** : Régression Softmax (synonymes : Logistique multinomiale, Classificateur d'entropie maximale, ou simplement Régression logistique multi-classe) est une généralisation de la régression logistique que nous pouvons utiliser pour la classification multi-classes . Contrairement à d'autres types de fonctions, la sortie d'un neurone d'une couche utilisant la fonction softmax dépend des sorties de tous les autres neurones de sa couche. Cela s'explique par le fait qu'il nécessite que la somme de toutes les sorties soit égale à 1. Dans[32], nous avons un exemple de reconnaissance de chiffres manuscrits. Dans cette tâche, chaque étiquette est comprise entre 0 et 9. Nous devons être confiants dans notre prédiction et cela peut être fait en utilisant une distribution de probabilité. Pour la tâche mentionnée, le vecteur de sortie souhaité est de la forme ci-dessous :

$$[P_0 P_1 P_2 P_3 \dots P_9]$$

Où :

$$\sum_{i=0}^g P_i = 1 \text{-----} 17$$

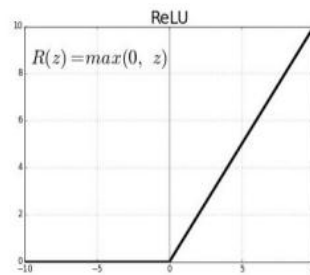
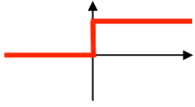
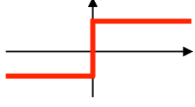
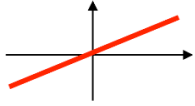


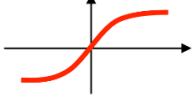
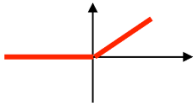
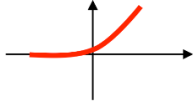


Figure 84: Représentation graphique de la fonction Relu.[32]

Tableau 8: Fonctions d'activations.[31]

Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	Multi-layer Neural Networks	
Rectifier, ReLU (Rectified Linear Unit)	$\phi(z) = \max(0, z)$	Multi-layer Neural Networks	
Rectifier, softplus	$\phi(z) = \ln(1 + e^z)$	Multi-layer Neural Networks	

Copyright © Sebastian Raschka 2016
(<http://sebastianraschka.com>)

IV. Réseaux de Neurones Convolutifs(CNN)

Jusqu'à présent, les réseaux de neurones Convolutifs, aussi appelés CNN ou ConvNet pour Convolutional Neural Network [132][133], sont toujours les modèles les plus performants pour la classification d'images. On distingue deux parties, une première partie que l'on appelle la partie convolutive du modèle et la seconde partie, que l'on appelle la partie classification du modèle qui correspond à un modèle MLP (Multi Layers Perceptron) (voir Figure 40).

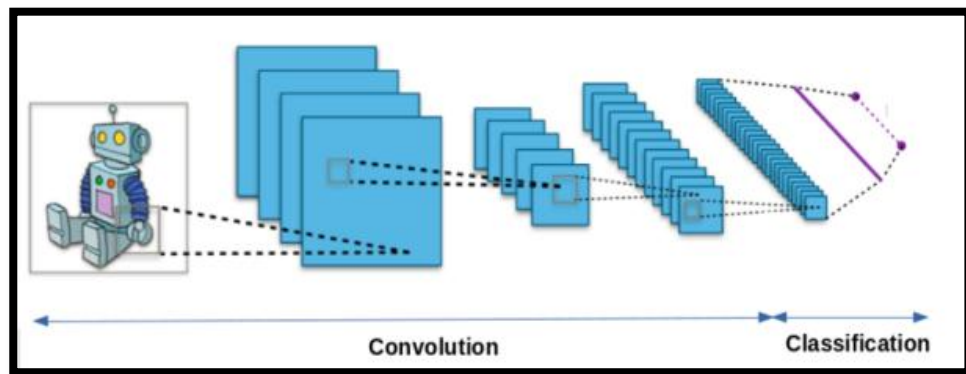


Figure 85 : Les deux(02) parties de Convolutional Neural Network.[62]

CNN est un réseau de neurones multicouches et plus précisément c'est un réseau profond composé de multiples couches qui en générale sont organisées en blocs (CONV→ RELU → POOL). Il existe quatre types de couches pour un réseau de neurone Convolutif (Figure 41):

- La couche de convolution.
- La couche de pooling.
- La couche de correction ReLU .
- La couche fully-connected.

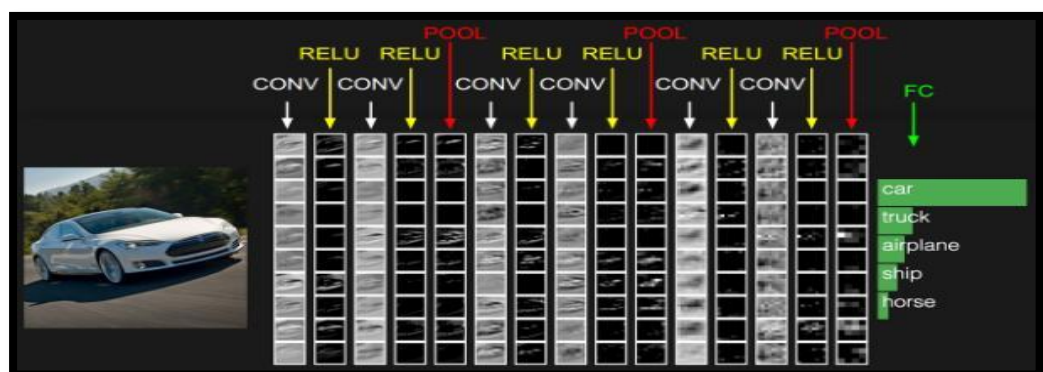


Figure 86 : Les quatre types de couches pour un réseau de neurones Convolutif.[62]

- **La couche de convolution** : La couche de convolution est la composante clé des réseaux de neurones convolutifs, et constitue toujours au moins leur première couche.

Son but est de repérer la présence d'un ensemble de composants dans les images reçues en entrée (Figure 42). Pour cela, on réalise un filtrage par convolution, le principe est de faire "glisser" une fenêtre représentant le composant sur l'image, et de calculer le produit de convolution entre le composant et chaque portion de l'image balayée.

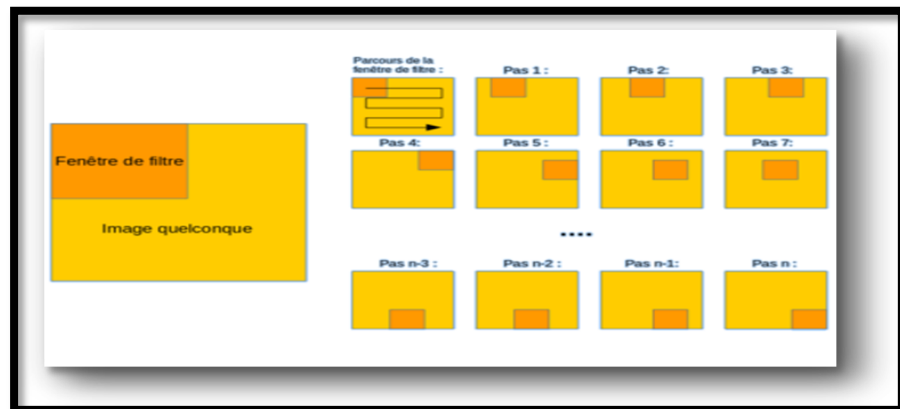


Figure 87 : Schéma du parcours de la fenêtre de filtre sur l'image .[62]

- **Couche de mise en commun**(pooling) : La couche de pooling permet de réduire le nombre de paramètres et de calculs dans le réseau. On améliore ainsi l'efficacité du réseau et on évite le sur-apprentissage. Ce type de couche est souvent placé entre deux couches de convolution : elle reçoit en entrée plusieurs composants maps, et applique à chacune d'entre elles l'opération de pooling.
- **La couche ReLU** :ReLU (Rectified Linear Units) désigne la fonction réelle non-linéaire définie par $\text{ReLU}(x) = \max(0, x)$. Elle remplace toutes les valeurs négatives reçues en entrées par des zéros. Elle joue le rôle de fonction d'activation [134]. Souvent, la correction Relu est préférable, mais il existe d'autres formes :
 - La correction par tangente hyperbolique $f(x) = \tanh(x)$,
 - La correction par la tangente hyperbolique saturante: $f(x) = |\tanh(x)|$,
 - La correction par la fonction sigmoïde : $f(x) = (1 + e^{-x})^{-1}$
- **Couche fully-connected**(FC) : La couche fully-connected constitue toujours la dernière couche d'un réseau de neurones, convolutif. La couche fully-

connected permet de classifier le texte en entrée du réseau : Elle reçoit un vecteur en entrée et produit un nouveau vecteur en sortie. Pour cela, elle applique une combinaison linéaire puis éventuellement une fonction d'activation aux valeurs reçues en entrée.

IV.1 Hyper-paramètres

Un réseau neuronal artificiel est constitué de paramètres modèles et d'hyper paramètres. Les paramètres du modèle sont des attributs tels que les poids et les biais que le modèle utilise pour s'adapter aux données. Les hyper-paramètres sont des attributs ou des propriétés qui dictent l'ensemble du processus d'entraînement et doivent être prédéfinis. [63] Les hyper paramètres doivent être prédéfinis car ils ne peuvent pas être directement appris du processus de formation.

Les hyper-paramètres sont les variables qui déterminent la structure du réseau (ex : nombre d'unités cachées) et les variables qui déterminent comment le réseau est formé (ex : taux d'apprentissage).

IV.1.1 Hyper-paramètres liés à la structure du réseau

IV.1.1.1 Nombre de couches cachées (hiddenlayers)

Dans les réseaux neuronaux à convolution, le nombre de couches cachées peut se rapporter soit au nombre de couches de convolutions, soit au nombre de couches denses. Dans cette recherche, le nombre de couches cachées se réfère au nombre de couches denses. Le but d'une couche est simple : elle transforme les données reçues par une fonction d'activation et envoie cette valeur à la couche de sortie ou à une autre couche cachée. Actuellement, il n'existe pas d'étude approfondie permettant d'établir une ligne directrice générale sur le nombre de couches convolutionnelles à utiliser, cependant, les problèmes nécessitant plus de deux couches cachées sont rares .[64]

IV.1.1.2 Le Dropout pour La régularisation

La régularisation est toute modification que nous apportons à un algorithme d'apprentissage dans le but de réduire son erreur de généralisation mais pas son erreur d'apprentissage .[27] Il s'agit de techniques pour éviter l'over-fitting, l'une de ses techniques est le Dropout.

Dropout est une technique, qui peut être appliquée à la sortie de certaines des couches du réseau. Il enlèvera aléatoirement et périodiquement certains des neurones (ainsi que leurs connexions d'entrée et de sortie) du réseau .[37]

La figure suivante illustre l'application du dropout après une couche complètement connectée.

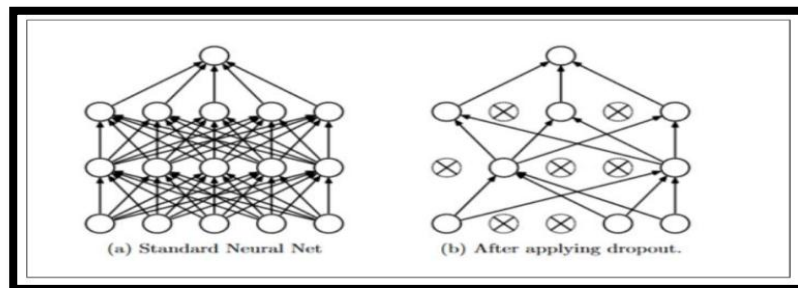


Figure 88 : A Gauche : Un réseau standard sans drop-out. A droite : Un réseau avec un drop-out.[64]

IV.1.1.3 Fonction d'activation

Les fonctions d'activation sont une composante extrêmement importante des réseaux de neurones profonds. Ils décident si un neurone doit être activé ou non. Si l'information que le neurone reçoit est pertinente pour l'information donnée ou si elle doit être ignorée [77]. Généralement, la fonction d'activation du redresseur est la plus populaire. **Sigmoïde** est utilisé dans la couche de sortie lors de la réalisation de prédictions **binaires**. **Softmax** est utilisé dans la couche de sortie lors de la réalisation de prédictions multi-classes.

Pour notre travail lors de la création des modèles CNN, nous avons opté pour la fonction d'activation **ReLU** dans tout le modèle sauf pour la couche de sortie, où on a utilisé la fonction d'activation **softmax**.

IV.1.1.4 Batch normalization

Il est connu depuis longtemps que l'apprentissage du réseau converge plus rapidement si ses entrées sont centrées réduites (moyenne = 0, variance = 1).[65] Comme chaque couche observe des entrées produites par des couches qui la précèdent, il serait avantageux d'obtenir des entrées centrées et réduites pour chaque couche.

Batch normalisation [64] est une composante qui se trouve entre les couches du réseau de neurones, et qui prend continuellement la sortie d'une couche particulière et la normalise avant de l'envoyer à la couche suivante comme entrée.

IV.1.1.5 Global averagepooling

Un réseau de neurones convolutifs traditionnel opère une convolution durant ses premières couches. Pour la classification, la dernière couche de convolution est aplatie puis injectée dans la première couche du perceptron. Cette structure relie la partie convolution à la partie classifieur qui est le perceptron. Les couches de convolution agissent comme des extracteurs de caractéristiques et le résultat est classé de manière traditionnelle.

Cependant le perceptron multi couche est sujet au sur-apprentissage, entravant ainsi la capacité de généralisation du réseau. Dropout fut proposé comme régularisateur.

Global Average Pooling (GAP) [66] fut proposé pour remplacer la partie perceptron multi couches.

L'idée c'est de générer une featuremap pour chaque catégorie correspondante. Au lieu d'ajouter un perceptron après les featuremap, on prend la moyenne de chacune d'elles et le résultat est injecté dans la fonction softmax. Un avantage du GAP par rapport aux couches de perceptron est qu'il n'y a aucuns paramètres à entraîner, par conséquent, le sur-apprentissage est évité.

IV.1.2 Hyper-paramètres liés à l'algorithme d'entraînement

IV.1.2.1 Taux d'apprentissage (Learning rate)

Le taux d'apprentissage est l'un des nombreux hyperparamètres qu'un utilisateur peut adapter dans un réseau de neurones pour améliorer la précision du modèle. Le taux d'apprentissage est une constante entre 0 et 1 qui permet d'ajuster les pondérations du réseau vers un minimum local ou mondial pour une fonction objective d'erreur.[67]

Le taux d'apprentissage peut être considéré comme la vitesse à laquelle le réseau apprend, c'est-à-dire à quelle vitesse les poids du réseau convergent.[67] Lorsque le taux d'apprentissage est relativement faible, les ajustements de poids sont faibles, ce qui fait que Le réseau prend beaucoup de temps à converger[67]. Cependant, si le taux d'apprentissage est augmenté Trop, l'apprentissage devient instable et peut dépasser la solution optimale. Le réseau oscille alors dans un sens ou dans l'autre par rapport au minimum d'erreurs.[68]

IV.1.2.2 Nombre d'époques

Le nombre d'époques est le nombre de fois que l'ensemble des données relatives à la formation est affiché sur le réseau pendant la formation.

Augmentez le nombre d'époques jusqu'à ce que la précision de validation commence à diminuer, même si la précision de l'entraînement augmente (surcharge). Pour notre travail.

IV.1.2.3 Batch Size

La taille du lot est le nombre de des échantillons de formation qui seront utilisés pour effectuer une mise à jour des paramètres du modèle jusqu'à tous les échantillons de formation ont été alimentés par le réseau. Idéalement, pour former un réseau, chaque échantillon de formation serait utilisé pour mettre à jour les paramètres du modèle, toutefois, il s'agit extrêmement inefficace et épuisant. En utilisant un lot d'échantillons par itération, la moyenne des gradients de chaque échantillon du lot est calculée et ensuite utilisée pour mettre à jour les paramètres du modèle. Intuitivement, la taille du lot affectera le temps qu'il faut pour former le réseau ainsi que la précision.

Dans la pratique, la taille du lot est généralement choisie en fonction de la quantité de données pouvant tenir en mémoire et être traitées en parallèle. Ainsi, l'hyperparamètre de la taille du lot n'est généralement pas réglé avec la validation croisée, mais plutôt choisi pour correspondre aux limites du matériel utilisé

Une bonne valeur par défaut pour la taille du lot peut être 32. Essayez également 32, 64, 128, 256, etc.

IV.1.2.4 Optimisateur

Les algorithmes d'optimisation nous aident à minimiser (ou maximiser) une "Fonction Objective" (autre nom pour la fonction Error) $E(x)$ qui est simplement une fonction mathématique qui dépend des paramètres d'apprentissage internes du modèle. Il existe plusieurs algorithmes d'optimisation à savoir[68] :

- **Gradient Descent**

C'est l'algorithme d'optimisation le plus populaire utilisé dans l'optimisation d'un réseau de neurones[68].

- **Variantes de Gradient Descent**

Gradient Descent calculera le gradient de l'ensemble des données mais n'effectuera qu'une seule mise à jour ; donc il peut être très lent et difficile à contrôler pour les datasets qui sont très volumineux.

- **Stochastic gradient descent**

Effectue une mise à jour des paramètres pour chaque exemple d'entraînement, il s'agit généralement d'une technique beaucoup plus rapide[68].

- **Optimisation de gradient descent**

Il existe plusieurs algorithmes qui sont utilisés pour optimiser encore plus le Gradient descent.

- **Momentum**

La grande variance des oscillations de la SGD rend difficile l'atteinte de la convergence. C'est pourquoi on a inventé une technique appelée Momentum qui accélère la SGD en naviguant dans la direction pertinente et atténue les oscillations dans des Directions non pertinentes [68].

- **Nesterovaccelerated gradient**

Un chercheur nommé YuriiNesterov a relevé un problème avec Momentum.Ce qui se passe en réalité, c'est que lorsque nous atteignons les minima, c'est-à-dire le point le plus bas de la courbe, l'élan (Momentum) est assez élevé et il ne sait pas ralentir ce qui lui fait rater complètement les minima[68].

- **Adagrad**

Il permet simplement au taux d'apprentissage de s'adapter en fonction des paramètres. Il fait donc de grosses mises à jour pour les paramètres peu fréquents et de petites mises à jour pour les paramètres fréquents [68].

- **AdaDelta**

Il s'agit d'une extension d'AdaGrad qui tend à supprimer le problème du taux d'apprentissage en décomposition [68].

- **Adam**

Adam signifie Adaptive Moment Estimation. L'estimation du moment adaptatif (Adam) est une autre méthode qui calcule les taux d'apprentissage adaptatif pour chaque paramètre[68].

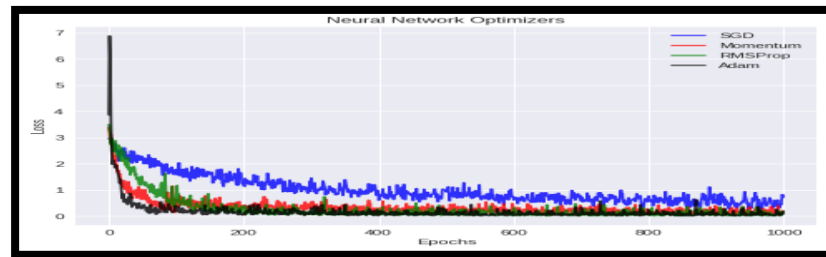


Figure 89: Illustration des différentes valeurs de fonction de coût obtenues par les différents algorithmes d'optimisation[68].

Nous avons travaillé avec l'optimisateur Adam parce qu'il donne la valeur minimale de la fonction de coût et est le premier à converger -voir figure ci-dessus-.

IV.1.2.5 Earlystoppingpour La régularisation

- **Earlystopping** est la méthode la plus simple et la plus efficace pour régulariser la solution et favoriser la généralisation des performances. En d'autres termes, la précision de la validation sur le jeu de validation tenu est contrôlée en permanence et la formation est arrêtée une fois que la précision cesse de s'améliorer. En pratique, on peut sauvegarder les paramètres du modèle les plus performants en plus des paramètres actuels et revenir sur celui qui a été sauvegardé lorsque de nouvelles améliorations semblent peu probables.

En résumé, les hyper-paramètres sont des engrenages qui règlent les paramètres du modèle, ce qui permet d'obtenir un modèle d'apprentissage plus précis.

IV.2 Apprentissage par transfert

L'apprentissage par transfert [69] est une stratégie qui vise à optimiser les performances d'une machine à apprendre à partir de connaissances et d'autres tâches effectuées par une autre machine d'apprentissage. En pratique, d'après[70]., Il n'est pas recommandée de former un modèle ConvNet dès le début (avec initialisation) car la formation d'un modèle

ConvNet nécessite beaucoup de données et prend beaucoup de temps. D'autre part, il est plus habituel d'utiliser les modèles ConvNet déjà en formation et de les réhabiliter pour le problème, c'est appelé "Transfer Learning", qui consiste à transférer l'apprentissage d'un modèle traitant d'un problème à un autre type de problème. Il existe deux types d'apprentissage par transfert :

- L'extraction de variables du ConvNet: ici, le ConvNet est utilisé comme extracteur[71], c'est-à-dire qu'un vecteur est extrait d'une certaine couche du modèle sans modifier sa structure ou son poids et le vecteur précédemment extrait est utilisé pour un nouveau spot.
- Le réglage fin (**fine tuning**) du modèle ConvNet[72][73][73]: ici, le nouveau ConvNet est initialisé avec les poids et la structure du modèle pré-entraîné est utilisée. La structure du modèle pré-entraîné est légèrement modifiée pour la nouvelle tâche et finalement le nouveau modèle est formé pour la nouvelle tâche.

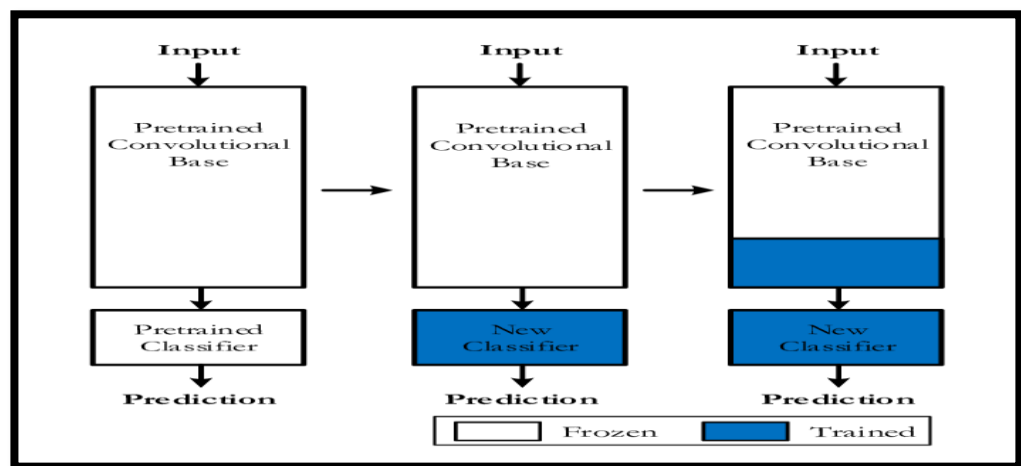


Figure 90 : Diagramme de haut niveau de l'apprentissage par transfert à partir d'un modèle de CNN préformé.[13]

La méthode consiste à prendre un réseau qui a été entraîné sur un jeu de données différent de celui du problème que l'on souhaite résoudre. Les poids sont alors réutilisés comme initialisation pour entraîner un deuxième réseau sur le jeu de données cible. Idéalement, le premier réseau va avoir appris sur un jeu de données beaucoup plus important que celui de la tâche cible, ce, qui lui permet de développer des extracteurs de caractéristiques qui généralisent bien. Dans un réseau de neurones à convolution, les

premières couches extraient des attributs de bas niveau, comme des arêtes ou des couleurs. Plus les couches sont profondes, plus elles extraient des attributs abstraits. Les dernières extrayant généralement des attributs spécifiques au problème. Comme les premières couches extraient essentiellement des caractéristiques très générales, celles-ci peuvent être réutilisées sur des nouveaux problèmes. Cela permet d'accélérer énormément l'entraînement et d'améliorer les résultats. C'est pourquoi, il est souvent beaucoup plus simple d'utiliser cette technique que d'essayer les autres méthodes d'initialisation étant donné que des modèles pré-entraînés sont facilement accessibles sur Internet.

Il est aussi possible de geler certaines couches du réseau. C'est-à-dire de ne pas modifier les valeurs des paramètres pour certaines couches du réseau. En général, ce sont les premières couches qui sont gelées, car elles ont déjà appris à extraire de bonnes caractéristiques. Cela permet de sauver du temps de calcul, car il n'est plus nécessaire de calculer le gradient pour ces couches et peut aider à réduire le sur-apprentissage. En règle générale, plus le problème qu'on cherche à résoudre est similaire à celui sur lequel le réseau a été pré-entraîné, plus on peut geler un nombre élevé de couches.

Voici peut-être trois des modèles les plus populaires :

- ✓ VGG (par exemple VGG16 ou VGG19).
- ✓ GoogLeNet (par exemple InceptionV3).
- ✓ Residual Network (par exemple ResNet50).

Ces modèles sont largement utilisés pour l'apprentissage par transfert, à la fois en raison de leurs performances, mais aussi parce qu'ils sont des exemples qui ont introduit des innovations architecturales spécifiques, à savoir des structures cohérentes et répétitives (VGG), des modules d'initiation (GoogLeNet) et des modules résiduels (ResNet).