

Mémoire de fin d'études

Traitemen~t d'image et Machine Learning pour la detection (dépistage) automatique de la rétinopathie diabétique

PROPOSÉ PAR :
Mme. BENATCHBA Karima
Mme.Faisal Touka
OPTION : SIQ

RÉALISÉ PAR:

BRAHIM Yacine
CHERIEF Yassine

Année universitaire : 2019/2020

Remerciements

Avant toute chose, nous souhaitons exprimer notre entière reconnaissance à toute personne ayant contribué de près ou de loin à la réalisation de ce travail.

D'abord, nous tenons à remercier nos encadrantes Mme. BENATCHBA Karima et Mme. FAISAL Touka pour leur disponibilité tout au long de ce projet et pour leurs précieux conseils.

Ensuite, nous aimerais remercier tous nos amis et nos camarades qui ont été présents et qui nous ont apporté leur soutien.

Yacine

J'aimerais remercier en particulier mon cher binôme et mon ami, CHERIEF Yassine, sans qui ce travail n'aurait pas pu être réalisé. Je le remercie pour tous ses efforts, son soutien, son sérieux et son investissement dans ce présent travail.

Je souhaite aussi remercier mes chers parents, ma sœur, mon frère pour leur soutien inestimable et leurs encouragements.

Yassine

Je tiens à témoigner toute ma reconnaissance envers mon ami et binôme, BRAHIM Yacine, pour son investissement total, son soutien ainsi que tous les efforts qu'il a fournis et sans lesquels ce travail n'aurait pas abouti.

Je remercie également ma mère qui a toujours été là pour moi tout au long de mon parcours à l'ESI.

Resume

La rétinopathie diabétique (RD) est une complication du diabète qui affecte les yeux.C'est l'une des premières causes de la cécité à travers le monde.Le diagnostic de cette maladie à travers les images du fond d'oeil d'un rétinographe requiert des cliniciens expérimentés pour pouvoir identifier la présence de plusieurs petites lésions caractérisantes de la RD.Ceci rend la détection de cette dernière très fastidieuse et très longue.

Ceci nous a motivé à mener ce modeste travail qui consiste à mettre en place un système de détection et de classification automatique de la RD basé sur les réseaux de neurones convolutionnels.Nous avons également mis en place une plateforme qui facilitera aux utilisateurs spécialistes et non spécialistes l'interaction avec le modèle de classification.

Le modèle de classification a été entraîné à l'aide d'une carte graphique sur une base d'apprentissage publié par Kaggle.Nous avons atteint une précision de classification égale à 89.1%,une spécificité de 91.4% et une sensibilité égale à 90.2%.

Mots-clés : Rétinopathie diabétique, réseaux de neurones convolutionnels, classification d'image.

Abstract

Diabetic retinopathy (DR) is a diabetic complication that affects the eyes. It is one of the leading causes of blindness around the world. The Diagnosis of DR through retinal fundus images obtained by a retinograph requires experienced clinicians in order to identify the presence of several small lesions characterizing DR. This makes the detection process of the latter very tedious and very long.

This motivated us to carry out this modest work in which we propose a detection and automatic DR classification system based on convolutional neural networks (CNN). We have also set up an online platform that will facilitate the interaction with the classification model for both specialists and non specialists.

We trained our network using a high-end graphics processor unit (GPU) on the publicly available Kaggle dataset and demonstrated impressive results. We achieved a classification accuracy equal to 89.1%, a specificity of 91.4% and a sensitivity equal to 90.2%.

Keywords : Diabetic retinopathy, Convolutional Neural Networks, Deep Learning, Image Classification.

Table des figures

| | | |
|------|----------------------------------------------------------------------------------------------------------------------|----|
| 1.1 | illustration des symptômes de la RD. | 5 |
| 1.2 | Rétinopathie diabétique minime. | 5 |
| 1.3 | Rétinopathie diabétique non proliférante modérée. | 6 |
| 1.4 | Rétinopathie diabétique non proliférante sévère. | 6 |
| 1.5 | Rétinopathie diabétique proliférante. | 7 |
| 1.6 | Image du fond d'œil prise par un rétinographe | 7 |
| 2.1 | Différence entre IA, AA et apprentissage profond. | 10 |
| 2.2 | La performances des modèles standards vs. DL en de la taille de la base d'apprentissage.source : Andrew Nj's course. | 12 |
| 2.3 | Un simple réseau de neurones avec 3 couches. | 12 |
| 2.4 | Traitemet effectué par un neurone. | 14 |
| 2.5 | Une itération du processus d'apprentissage. | 18 |
| 2.6 | Exemple de produits de convolutions avec des filtres connus. | 21 |
| 2.7 | Exemple d'un sous-échantillonnage en 3D. | 21 |
| 2.8 | Exemple d'un sous-échantillonnage Max pooling en 2D. | 22 |
| 2.9 | Une simple architecture d'un réseau de neurones convolutionnel. | 23 |
| 3.1 | Variations de l'énergie par rapport a l'orientation | 27 |
| 3.2 | Les étapes de traitement de l'image originale | 29 |
| 3.3 | Résultats de la détection des exsudats | 33 |
| 3.4 | Les étapes du processus d'élimination de la papille optique | 35 |
| 3.5 | Les étapes de détection des exsudats. | 36 |
| 4.1 | Structure du réseau. | 42 |
| 4.2 | Illustration du processus Data-augmentation. | 44 |
| 4.3 | Matrice de confusion. | 44 |
| 4.4 | Problème de la taille optimale des portions. | 46 |
| 4.5 | Exemple de lésions détectées. | 47 |
| 4.6 | L'architecture du CNN. | 48 |
| 4.7 | Résultats de classification des lésions sur SiDRP. | 48 |
| 4.8 | Résultats de classification des lésions sur DIARETDB1. | 49 |
| 4.9 | Résultats de la classification des images sur SiDRP. | 49 |
| 4.10 | La différence entre l'apprentissage traditionnel et l'apprentissage par transfert | 51 |
| 4.11 | Le pipeline de transfert des paramètres. | 53 |

| | |
|--------------------------------------------------------------------------------------------------|-----|
| 4.12 Résultats de la classification basée sur full fine-tunning. | 54 |
| 4.13 Description du schéma du fine-tunning partiel. | 55 |
| 4.14 Résultats de la classification basée sur le fine-tunning partiel. | 55 |
| 6.1 Processus de classification de la RD | 65 |
| 6.2 Images extraites de la base d'apprentissage | 66 |
| 6.3 Le résultat après l'amélioration du contraste | 66 |
| 6.4 Le résultat après le floutage | 67 |
| 6.5 Le résultat après la suppression du fond noir. | 68 |
| 6.6 illustration du processus de prétraitement. | 68 |
| 6.7 Un bloc résiduel. | 70 |
| 6.8 bloc identité. | 71 |
| 6.9 bloc de convolution. | 71 |
| 6.10 Partie 1 du ResNet50. | 72 |
| 6.11 Partie 2 du ResNet50. | 72 |
| 6.12 Partie 3 du ResNet50 : réseau entièrement connecté. | 73 |
| 6.13 Les images originales | 77 |
| 6.14 Exemple de data augmentation | 77 |
| 7.1 Logo de python | 81 |
| 7.2 Logo de Google Colab | 81 |
| 7.3 Logo de Django | 82 |
| 7.4 Logo de Google Drive | 82 |
| 7.5 architecture de la plate-forme RetinoCheck | 86 |
| 7.6 Modèles de l'application model_prediction | 86 |
| 7.7 Modèles de l'application authentication | 86 |
| 7.8 Modèles de l'application CORE_APP | 87 |
| 7.9 Choix de l'image du test | 88 |
| 7.10 Résultats du test sur la plateforme | 89 |
| 7.11 Rapport du test sur la plateforme | 89 |
| 7.12 Étiquetage des images du fond d'œil | 90 |
| 8.1 La courbe d'apprentissage. | 94 |
| 8.2 La courbe d'apprentissage (précision). | 95 |
| 9.1 Matrice de confusion. | 98 |
| 10.1 La courbe d'apprentissage dans le cas d'un apprentissage par transfert. | 100 |
| 10.2 La courbe d'apprentissage (précision) dans le cas d'un apprentissage par transfert. | 100 |
| 10.3 Matrice de confusion du modèle dans le cas d'un apprentissage par transfert. | 101 |

| | |
|------------------------------------------------------------------------------------------------------|-----|
| 10.4 Matrice de confusion du modèle dans le cas d'un apprentissage par transfert sélectif. | 102 |
| 10.5 Courbe d'apprentissage dans le cas d'un apprentissage par transfer sélectif. | 103 |
| 10.6 Courbe d'apprentissage(précision) dans le cas d'un apprentissage par transfer sélectif. | 103 |

Liste des tableaux

| | | |
|------|-------------------------------------------------------------------------------------------------------------------------------------|-----|
| 5.1 | Tableau récapitulatif | 58 |
| 6.1 | Le nombre et la fréquence des images dans chaque classe | 63 |
| 7.1 | Bibliothèques utilisées dans l'implementation | 83 |
| 9.1 | Spécificité et sensibilité. | 98 |
| 10.1 | Spécificité et sensibilité du modèle dans le cas d'un apprentissage par transfert et dans le cas d'un apprentissage à zéro. | 101 |
| 10.2 | Spécificité et sensibilité dans le cas d'un apprentissage par transfert sélectif. | 102 |

Introduction générale

Le diabète — ou le diabète sucré — est un trouble métabolique chronique caractérisé par un taux de glucose excessif dans le sang. Celui-ci est dû à un manque absolu d'insuline (diabète de type 1) ou à un manque relatif voir une efficacité insuffisante de l'insuline présente (diabète de type 2). Au fil du temps ce taux élevé de glucose dans le sang endommage les vaisseaux sanguins, ces dommages affectent à la fois le cœur, le système nerveux, les reins ainsi que les yeux.[1]

La croissance rapide du nombre des diabétiques est l'un des principaux défis de l'organisation mondiale de la santé. Le nombre de personnes atteintes de la maladie continue de croître à un rythme alarmant, l'organisation mondiale de la santé prévoit qu'en 2040, le nombre de personnes atteintes de diabète à travers le monde avoisinera les 600 millions [2], [3]. La situation devient de plus en plus critique du fait que seulement la moitié des diabétiques se rendent compte de leurs maladies, ceci constitue une réelle menace à leurs santés. En effet une étude montre qu'un patient diabétique est 25 fois plus susceptible à perdre la vue qu'un patient sain [4].

Le diabète est connu pour être à l'origine de dizaines de maladies graves entre-autres la rétinopathie diabétique (RD) [1].

La rétinopathie diabétique (RD) est une complication microvasculaire du diabète, c'est une maladie très fréquente chez les diabétiques (30 % - 31 %) des diabétiques développent une rétinopathie diabétique après les deux premières années, les statistiques montrent également que plus de 50% des patients diabétiques de type2 sont touchés par la RD [3].

Le caractère discret de la RD fait d'elle une maladie désastreuse (l'une des causes majeures de la cécité dans le monde). En effet une personne atteinte par la RD ne se rend compte qu'à un stade avancé de la maladie où le traitement de cette dernière devient très compliqué voire impossible [5].

Jusqu'à ce jour, le traitement le plus efficace de la RD ne peut être administré qu'aux premiers stades de la maladie. Par conséquent, la détection précoce grâce à un dépistage régulier est d'une importance capitale. Pour réduire le coût de ces dépistages à l'examen du fond d'œil une technologie de capture d'image numérique peut être utilisée pour ce fait, car cette dernière permettra d'automatiser le processus de détection de la RD et réduisant ainsi les coûts de dépistage de la maladie.

Par ailleurs, la RD peut être une maladie fatale dans plusieurs pays d'Afrique subsaharienne où les systèmes de santé sont exsangues. En effet ces pays font face à une pénurie importante des établissements de santé et des praticiens. Par conséquent, ces pays font face à un énorme fardeau économique et de santé publique en raison du diabète et ses

complications notamment la RD. Les statistiques montrent que la prévalence du diabète en Afrique varie entre 6.3% et 15.4% de la population dont 30.2 – 31.6 % souffrent de la rétinopathie diabétique [3].

Malheureusement ces pays détiennent le ratio le plus faible d'ophtalmologues / millions d'habitants soit 3 ophtalmologues pour un million d'habitants contre 80 ophtalmologues / million d'habitants dans les pays développés [6]. Ce manque drastique d'ophtalmologues dans ces pays représente un réel défi pour les systèmes de santé et pour la santé publique.

Ce manque de praticiens , d'infrastructures et de sensibilisation du public est l'un des principaux défis à relever immédiatement.Afin de prévenir la cécité dans ces pays,il est primordial de mettre en place des systèmes de détection automatique de la RD peu coûteux pour assister les ophtalmologues et/ou non ophtalmologues à diagnostiquer la maladie avec moins de pression et éviter que le diagnostic soit erroné en raison du nombre élevé des patients et par faute de temps.

Au cours de ces dernières années, l'intelligence artificielle et plus précisément l'apprentissage profond et l'apprentissage par transfert ont permis d'aboutir à des résultats très satisfaisants dans l'imagerie médicale et la détection des maladies notamment la détection des tumeurs [7], [8].

La précision des résultats de classification et de détection des maladies de tels modèles d'apprentissage profond s'est avérée comparable à une classification faite par des spécialistes en milieu hospitalier. Ces modèles basés sur les techniques d'apprentissage profond offrent des avantages majeurs pour les soins de santé en réduisant le travail manuel et en permettant des économies de coûts et de ressources, ce potentiel devrait être incorporé et déployé dans les systèmes de dépistage qui ne sont actuellement pas dotés d'une telle technologie.

Les facteurs énoncés ci-dessus nous ont motivé à mener le présent travail qui consiste, dans un premier temps, à mettre en place un système de détection automatique de la rétinopathie diabétique et ses différents stades. Ce dernier est entièrement basé sur des techniques de traitement d'images et d'apprentissage profond. Dans un second lieu nous allons proposer une plateforme d'aide à la détection de la RD, la dite plateforme sera dotée de fonctionnalités qui feront d'elle à la fois un outil pédagogique et professionnel qui aide les ophtalmologues en phase d'apprentissage à mieux visualiser les caractéristiques de la RD et d'assister les non spécialistes à dépister la maladie.

Table des matières

| | |
|-------------------------------------------------------------------------------------|-----------|
| Résumé | 5 |
| Abstract | 7 |
| Table des figures | 9 |
| Liste des tableaux | 13 |
| Introduction générale | i |
| | |
| I État de l'art | 1 |
| | |
| 1 La rétinopathie diabétique | 3 |
| 1.1 Introduction et définition | 3 |
| 1.2 Symptômes et caractéristiques de la RD | 3 |
| 1.2.1 Les microanévrismes rétiniens | 3 |
| 1.2.2 Les hémorragies | 4 |
| 1.2.3 Les exsudats | 4 |
| 1.2.4 Les nodules cotonneux | 4 |
| 1.2.5 Anomalies microvasculaires intrarétiniennes « AMIR » | 4 |
| 1.2.6 Néovaisseaux prérétiniens et prépapillaires | 4 |
| 1.2.7 L'oedème maculaire | 4 |
| 1.3 Classification et stades de la rétinopathie diabétique | 5 |
| 1.3.1 Rétinopathie diabétique non proliférante minime | 5 |
| 1.3.2 Rétinopathie diabétique non proliférante modérée | 6 |
| 1.3.3 Rétinopathie diabétique non proliférante sévère (préproliférante) | 6 |
| 1.3.4 Rétinopathie diabétique proliférante | 6 |
| 1.4 Diagnostic de la rétinopathie diabétique | 7 |
| 1.5 La photographie du fond d'œil | 7 |
| | |
| 2 Apprentissage profond | 9 |
| 2.1 Introduction | 9 |
| 2.2 Différence entre intelligence artificielle,apprentissage automatique et appren- | |
| tissage profond | 9 |
| 2.3 Origine et histoire des réseaux de neurones artificiels | 10 |

TABLE DES MATIÈRES

| | | |
|----------|-----------------------------------------------------------------------------------------------------------------------|-----------|
| 2.4 | Pourquoi les réseaux de neurones artificiels ? | 11 |
| 2.5 | La composition et principe des réseaux de neurones artificiels | 12 |
| 2.5.1 | Fonction de combinaison | 13 |
| 2.5.2 | Fonction d'activation | 13 |
| 2.5.3 | Les poids synaptiques (weights and bias) | 14 |
| 2.6 | L'apprentissage d'un réseau de neurones | 14 |
| 2.6.1 | Fonction objectif | 15 |
| 2.6.2 | Forward propagation | 17 |
| 2.6.3 | Algorithme du gradient (Gradient Descent) | 17 |
| 2.6.4 | Rétropropagation du gradient (backpropagation) | 17 |
| 2.7 | Réseaux de neurones convolutionnels "CNN" | 19 |
| 2.7.1 | Les couches d'un réseau de neurones convolutionnel | 20 |
| 2.7.2 | Exemple d'une architecture CNN simple | 23 |
| 2.8 | Métriques d'évaluation | 24 |
| 2.9 | Conclusion | 24 |
| 3 | Feature based methods | 25 |
| 3.1 | Introduction | 25 |
| 3.2 | Détection de la structure des vaisseaux sanguins | 25 |
| 3.2.1 | Estimer le profil sanguin avec une fonction gaussienne et sa dérivée | 25 |
| 3.2.2 | Filtre Kalman et les filtres gaussien adaptées | 26 |
| 3.2.3 | Conclusion | 27 |
| 3.3 | Détection des microanévrismes | 28 |
| 3.3.1 | Introduction | 28 |
| 3.3.2 | Ségmentation des MA dans des Angiographies fluoroscénique | 28 |
| 3.3.3 | Détection des MA par la Normalisation local du contraste | 31 |
| 3.4 | Détection des exsudats | 33 |
| 3.4.1 | Introduction | 33 |
| 3.5 | Discussion sur les feature-based methods | 37 |
| 4 | Introduction des réseaux de neurones dans la détection et la classification de la rétinopathie diabétique (RD) | 39 |
| 4.1 | Méthode 1 : Convolutional Neural Networks for Diabetic Retinopathy[26] . . | 41 |
| 4.1.1 | Les réseaux de neurones convolutionnels ou Convolutional neural network «CNN» | 41 |
| 4.1.2 | Problème de surapprentissage | 41 |
| 4.1.3 | Structure du réseau de neurones | 42 |
| 4.1.4 | Prétraitement | 43 |
| 4.1.5 | Training | 43 |
| 4.1.6 | Data-Augmentation | 44 |

| | | |
|----------|--------------------------------------------------------------------------------------------------------------------------------|-----------|
| 4.1.7 | Résultats | 44 |
| 4.2 | Méthode 2 : Transformed Representations for Convolutional Neural Networks in Diabetic Retinopathy Screening [27] | 45 |
| 4.2.1 | Identification des candidats(Multiscale C-MSER Segmentation) . . | 46 |
| 4.2.2 | Representational Transformation | 46 |
| 4.2.3 | Le réseau de neurones convolutionnels «CNN» | 47 |
| 4.2.4 | Résultats | 47 |
| 4.3 | Méthode 3 : CNN Based Transfer Learning for Diabetic Retinopathy Fundus Image Classification. Xiaogang Li et al [28] | 50 |
| 4.3.1 | Apprentissage par transfert | 51 |
| 4.3.2 | Les modèles CNNs pré-entraînés | 52 |
| 4.3.3 | Fine-tunning des modèles pré-entraînés | 52 |
| 4.3.4 | Implémentation et Résultats | 52 |
| 5 | Tableau Récapitulatif | 57 |

II CONCEPTION ET RÉALISATION

| | | |
|----------|-----------------------------------------------------------------------|-----------|
| 6 | Conception du modèle | 61 |
| 6.1 | Introduction | 61 |
| 6.2 | La Base d'apprentissage | 62 |
| 6.2.1 | Introduction | 62 |
| 6.2.2 | Description | 62 |
| 6.2.3 | Étude et statistiques | 62 |
| 6.2.4 | Problème de déséquilibre de la base d'apprentissage | 63 |
| 6.2.5 | Choix de technique d'équilibrage | 64 |
| 6.3 | Processus général de la classification de la RD | 64 |
| 6.3.1 | Pré-traitement | 65 |
| 6.3.2 | L'architecture du réseau de neurones convolutionnel proposé | 69 |
| 6.3.2.1 | Introduction | 69 |
| 6.3.2.2 | ResNet50 | 69 |
| 6.4 | Pourquoi avoir choisi ResNet50 ? | 73 |
| 6.5 | Les techniques utilisées pour éviter le surapprentissage | 74 |
| 6.5.1 | Normalisation du lot (batch normalization) | 74 |
| 6.5.2 | Dropout | 75 |
| 6.5.3 | Régularisation | 75 |
| 6.6 | Data Augmentation | 75 |

TABLE DES MATIÈRES

| | |
|-----------------------------------------------------------------------------------|------------|
| 7 Réalisation et Implémentation de l'outil RetinoCheck | 79 |
| 7.1 Introduction | 79 |
| 7.2 Présentation générale de la plateforme | 79 |
| 7.3 Fonctionnalités de l'outil | 80 |
| 7.4 Présentation des outils et technologies utilisés | 81 |
| 7.4.1 Langage de programmation | 81 |
| 7.4.2 Environnement de développement | 81 |
| 7.4.3 Frameworks | 82 |
| 7.4.4 Stockage de la base d'apprentissage | 82 |
| 7.4.5 Bibliothèques | 82 |
| 7.5 Présentation de l'architecture de Django | 84 |
| 7.6 Présentation de l'architecture de la plateforme | 85 |
| 7.7 Implémentation du modèle de classification | 87 |
| 7.7.1 Pré-traitement des images | 87 |
| 7.7.2 Validation du modèle | 87 |
| 7.8 Scénarios d'utilisation | 87 |
| 7.9 Conclusion | 90 |
| III TESTS ET RÉSULTATS | 91 |
| 8 Entrainement du modèle | 93 |
| 8.1 Introduction | 93 |
| 8.1.1 Valeurs des hyperparamètres | 93 |
| 8.1.2 Courbes d'apprentissage | 94 |
| 9 Tests et Résultats | 97 |
| 9.1 Introduction | 97 |
| 9.2 Résultats | 98 |
| 10 Apprentissage par transfert | 99 |
| 10.1 Introduction | 99 |
| 10.2 Impact d'apprentissage par transfert sur la vitesse d'apprentissage. | 99 |
| 10.3 Apprentissage par transfert sélectif | 102 |
| 10.4 Conclusion | 104 |
| Conclusion | 105 |
| Acronymes | 111 |
| Glossaire | 113 |

PREMIÈRE PARTIE

État de l'art

La rétinopathie diabétique

1.1 Introduction et définition

La rétinopathie diabétique est une micro Angiopathie : une infection capillaire de la rétine (vaisseaux sanguins de la rétine). C'est une grave complication du diabète qui touche 50 % des patients diabétiques de type 2 (à différents stades), Elle est considérée comme l'une des causes majeures de la cécité dans le monde [9]. Au cours de diabète l'excès de glucose dans le sang (en dessus de la norme) fragilise les parois des petits vaisseaux sanguins qui perdent leur étanchéités et leur élasticités avec le temps. Ceci entraîne une apparition de lésions dans ces vaisseaux. La rétine étant riche de ces derniers se retrouve parmi les plus touchée.

Le phénomène se propage pour atteindre des zones nombreuses de la rétine. Ces dernières ne seront plus alimentées en nourriture et oxygène jusqu'à arriver à la Macula qui se trouve au centre de la rétine. Cette dernière est considérée comme le centre de la vision, lorsque le phénomène atteint ce niveau il entraîne systématiquement un gonflement de la Macula ce qui induit une baisse d'acuité visuelle [10].

Les vaisseaux alimentant la Macula se trouvent donc de plus en plus fragilisés et peuvent saigner facilement entraînant de graves hémorragies et par conséquent une perte de la vue.

1.2 Symptômes et caractéristiques de la RD

La rétinopathie diabétique regroupe des lésions de la rétine caractéristiques chez des personnes ayant un diabète depuis plusieurs années. Les anomalies rétiennes qui caractérisent la rétinopathie diabétique suivent une évolution prédictible sans variations majeures dans leur ordre d'apparition. Parmi ces lésions caractéristiques on trouve :

1.2.1 Les microanévrismes rétiniens

Ce sont des ectasies pariétales (gonflement anormal) des veines rétiennes dû à la fragilisation des parois de ces dernières. Ce sont les premiers signes ophtalmoscopiques de la RD. Ils apparaissent sous forme de points rouges de petite taille ou sous forme

de points blancs fluorescents en Angiographie¹. Les microanévrismes peuvent disparaître spontanément. Mais l'augmentation du nombre des microanévrismes est un bon indice de progression de la RD.

1.2.2 Les hémorragies

Ils ont un aspect analogue, et sont parfois difficiles à distinguer des microanévrismes. Ce sont des saignements de la rétine causés par la rupture des microanévrismes et des capillaires.

1.2.3 Les exsudats

Ce sont des accumulations de lipoprotéines dans l'épaisseur de la rétine ; ils apparaissent sous forme de dépôts jaunes, et sont habituellement disposés en couronne.

1.2.4 Les nodules cotonneux

Ce sont des lésions blanches, superficielles et de petite taille. Ils correspondent à une nécrose ischémique (un arrêt de la circulation sanguine causé par un caillot ou bouchon) localisées des fibres optiques rétiennes, due à l'occlusion d'une artériole pré-capillaire.

1.2.5 Anomalies microvasculaires intrarétiniennes « AMIR »

Ce sont des petites proliférations capillaires intra-rétiniennes en bordure des zones de non-perfusion capillaires.

1.2.6 Néovaisseaux prérétiniens et prépapillaires

Ce sont des proliférations anormales de vaisseaux rétiens à partir de la surface de la rétine ou de la papille.

1.2.7 L'œdème maculaire

C'est un épaississement rétinien dû à l'accumulation de liquide dans le tissu interstitiel de la rétine. Lorsqu'il est important, il prend un aspect d'œdème maculaire cystoïde (OMC).

La 1.1 suivante illustre toutes les lésions décrites ci-dessous :

1. une technique d'imagerie médicale portant sur les vaisseaux sanguins qui ne sont pas visibles sur des radiographies standards. Elle impose l'injection d'un produit de contraste lors d'une imagerie par rayons X. L'angiographie est un examen invasif

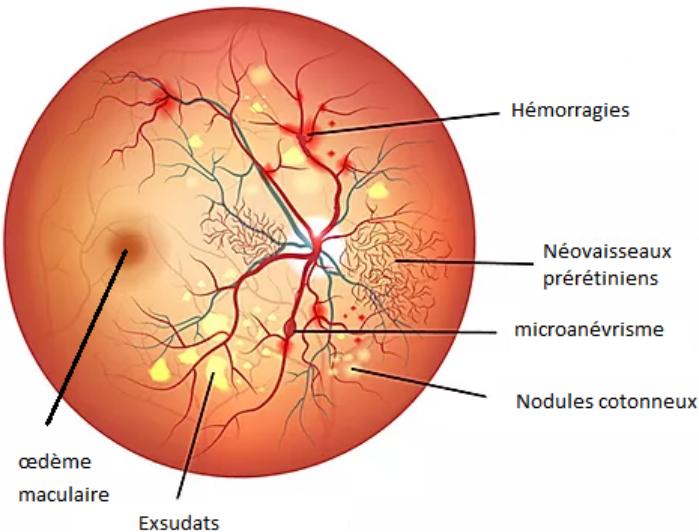


FIGURE 1.1 – illustration des symptômes de la RD.

1.3 Classification et stades de la rétinopathie diabétique

La rétinopathie diabétique débute par un stade de rétinopathie diabétique non proliférante, et évolue vers la rétinopathie diabétique non proliférante sévère (ou préproliférante), puis vers le stade de la rétinopathie diabétique proliférante caractérisée par la prolifération de néovaisseaux à la surface de la rétine et/ou sur la papille[11]. Selon les lésions présentes dans la rétine la RD peut être attribué à l'un des stades suivants :

1.3.1 Rétinopathie diabétique non proliférante minime

Elle est caractérisée par la présence des microanévrismes uniquement et dans certains cas très rares par quelques petits exsudats.

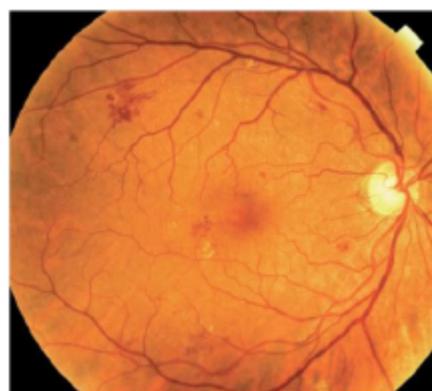


FIGURE 1.2 – Rétinopathie diabétique minime.

1.3.2 Rétinopathie diabétique non proliférante modérée

Elle est caractérisée par un nombre important de microanévrismes, la présence des exsudats, l'apparition des nodules cotonneux et des hémorragies intrarétiniennes.



FIGURE 1.3 – Rétinopathie diabétique non proliférante modérée.

1.3.3 Rétinopathie diabétique non proliférante sévère (préproliférante)

Elle est caractérisée par des microanévrismes et des hémorragies étendues sur les 4 quadrants de la rétine, la présence des anomalies microvasculaires intra-rétiniennes « AMIR » et des modifications veineuses.

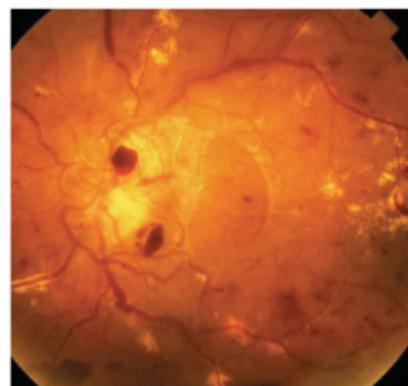


FIGURE 1.4 – Rétinopathie diabétique non proliférante sévère.

1.3.4 Rétinopathie diabétique proliférante

C'est le stade le plus avancé de la maladie. En effet les signaux envoyés par la rétine pour se nourrir déclenchent la croissance de nouveaux vaisseaux sanguins appelés néovaisseaux. Ces nouveaux vaisseaux sanguins se caractérisent par leurs parois minces et fragiles, ce qui entraîne un risque élevé de saignement. Ce saignement contamine le gel vitreux et cela provoque une perte de vision sévère et pouvant aller jusqu'à la cécité.

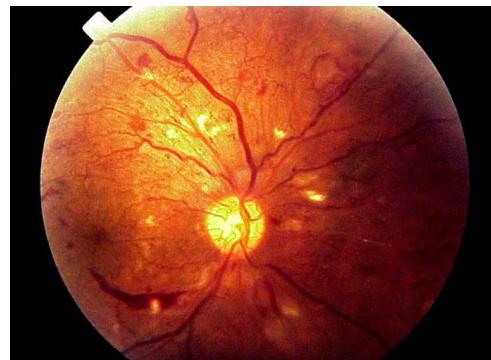


FIGURE 1.5 – Rétinopathie diabétique proliférante.

1.4 Diagnostic de la rétinopathie diabétique

La rétinopathie diabétique se développe discrètement sans que le malade ne perçoive le moindre symptôme. C'est la baisse de l'acuité visuelle qui témoigne de la présence des lésions avancées caractérisant la maladie. De ce fait il est primordial que les diabétiques soient sensibilisés et aient une éducation d'effectuer des contrôles réguliers chez leurs ophtalmologues [12].

Le diagnostic de la rétinopathie diabétique repose sur l'examen biomicroscopique du fond d'œil après la dilatation de la pupille pour avoir un champ de vision plus large complété par des photographies du fond d'œil.

1.5 La photographie du fond d'œil

La photographie du fond d'œil est l'examen de référence pour le diagnostic de la RD ; elle permet d'avoir un document de référence. Des photographies de la surface intérieure de l'œil, y compris la rétine, le système vasculaire rétinien, le disque optique, la Macula et le pôle postérieur du pôle postérieur sont réalisés. Elle permet aussi de visualiser les différents signes de la RD présents dans la surface intérieur de l'œil. En effet, la gravité de celle-ci est estimée par le nombre et la sévérité d'un certain nombre de signes cliniques (hémorragies intra-rétiennes, anomalies veineuses, AMIR).

La photographie du fond d'œil est également utilisée pour le dépistage de la RD qui d'adresse aux patients sans RD connue. Les photographies peuvent être prises en général sans dilatation pupillaire par des techniciens non médecins à l'aide de Rétinographe. L'illustration suivante est une image du fond d'œil prise par un Rétinographe.

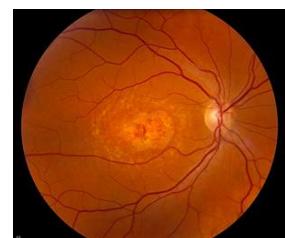


FIGURE 1.6 – Image du fond d'œil prise par un rétinographe

Apprentissage profond

2.1 Introduction

Au cours de ces dernières années l'intelligence artificielle a fait un progrès énorme pratiquement dans tous les domaines ce qui a fait que cette discipline est devenue un domaine de recherche très actif et a attiré un grand nombre de chercheurs qui contribuent aujourd'hui à son évolution.

L'intelligence artificielle a révolutionné la vie quotidienne des gens. Son impact sur le quotidien de chacun de nous est remarquable. En effet elle est omniprésente ; elle apparaît même dans nos actions les plus élémentaires telles que la recommandation des articles susceptible de plaire au client, la recommandation des musiques et la proposition des meilleurs itinéraires selon le trafic routier . . . etc. L'IA est présente également dans un tas de domaine ; par exemple, en imagerie médicale, des techniques d'IA sont utilisées pour la détection et la segmentation des lésions de plusieurs maladies, elle est aussi présente dans le domaine d'automobile et la fabrication des véhicules autonomes.

Le succès qu'a connu l'intelligence artificielle ces dernières années est dû principalement à l'efficacité des techniques d'apprentissage automatique (Machine Learning) et plus précisément l'apprentissage profond qui ne cessent d'évoluer et de prouver leur fiabilité et leur robustesse mais surtout leur utilisabilité.

2.2 Différence entre intelligence artificielle, apprentissage automatique et apprentissage profond

L'intelligence artificielle c'est l'ensemble des méthodes et techniques employées pour simuler le processus de l'intelligence humaine avec la machine. C'est l'art de créer des machines exécutant des tâches qui nécessitent de l'intelligence lorsqu'elles sont exécutées par l'humain "Kurzweil, 1990".

L'apprentissage automatique est une sous branche de l'intelligence artificielle. C'est un ensemble de méthodes et techniques dont l'objectif est de permettre aux machines

d'apprendre par eux-mêmes. Un algorithme d'apprentissage automatique vise à identifier l'utilité et la contribution de chaque information en entrée au résultat final pour pouvoir prédire des résultats sans avoir de règles et de modèles explicites préprogrammés. Autrement dit l'apprentissage automatique signifie la capacité d'une machine à apprendre à accomplir une certaine tâche sans être explicitement programmée pour le faire.

L'apprentissage profond est une sous branche de l'apprentissage automatique, il consiste à décomposer la tâche et la répartir sur des algorithmes d'apprentissage automatique organisés en couches successives (ex :régression linéaire).Chaque couche s'appuie sur la sortie de la couche précédente. Ces couches constituent ensemble un réseau de neurone artificiel qui imite le fonctionnement du réseau de neurone humain pour résoudre un problème donné.

La figure ci-dessous montre la différence entre l'intelligence artificielle, l'apprentissage automatique et l'apprentissage profond.

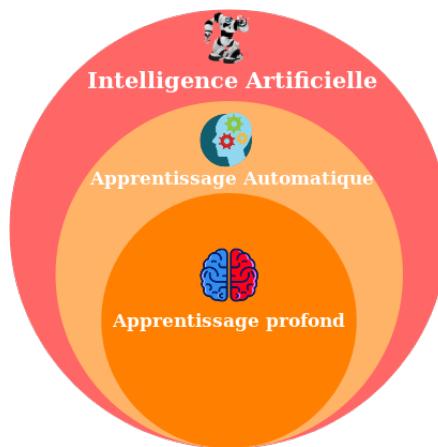


FIGURE 2.1 – Différence entre IA, AA et apprentissage profond.

2.3 Origine et histoire des réseaux de neurones artificiels

Ce fut les neurologues Warren McCulloch and Walter Pitts qui ont l'idée d'étudier le réseau de neurones biologique pour comprendre son fonctionnement vers de début des années 40, Ils ont introduit la notion du neurone formel qui n'est rien d'autre qu'une représentation mathématique d'un neurone biologique (voir la figure 2.4) et ont abouti au fait que les réseaux de neurones formels simples peuvent théoriquement réaliser des fonctions logiques, arithmétiques et symboliques complexes[13].Cependant les deux neurologues n'ont pas proposé une méthode de mise à jour des paramètres du réseau de neurones, c'est donc

Donald Hebb qui est venu avec une nouvelle méthode d'adaptation des poids synaptiques dans son ouvrage *The Organization of Behavior*[14]. Par la suite Rosenblatt a introduit le concept du perceptron ; il s'agit d'un simple réseau constitué d'une couche d'entrée et une couche de sortie utilisé dans les problèmes de classification binaire [28]. Cette dernière découverte fut contestée car le perceptron est en fait n'était pas adapté aux problèmes non linéaires ce qui constitue une limitation importante. Ce domaine de recherche est resté inactif durant les années qui se sont suivies jusqu'à l'an 1975 où Werbos a proposé l'algorithme de rétropropagation du gradient[15], ce dernier est utilisé pour mettre à jour les paramètres du réseau de neurones et a donc relancé à nouveau le domaine de recherche.

Durant la période 1975-2010 les réseaux de neurones artificiels étaient très peu utilisé pour deux principales raisons :

- L'absence des bases d'apprentissage sur lesquelles les réseaux de neurones peuvent apprendre.
- L'absence des unités de calcul pour pouvoir entraîner un réseau de neurones dans un temps raisonnable.

Avec le développement des capacités des machines plusieurs modèles de réseaux de neurones artificiels ont connu un grand succès et ont fait leurs preuves dans plusieurs domaines entre autres : les réseaux de neurones convolutionnels, les réseaux de neurones récurrents.

2.4 Pourquoi les réseaux de neurones artificiels ?

L'une des raisons qui fait que les réseaux de neurones sont les plus utilisés parmi les autres méthodes est que les modèles d'apprentissage profond ont tendance à bien fonctionner avec un nombre de données important grâce à l'interconnexion entre les couches du réseau contrairement aux anciens modèles d'apprentissage automatique qui cessent de s'améliorer après un certain point appelé point de saturation. Comme on peut voir dans la figure 2.2.

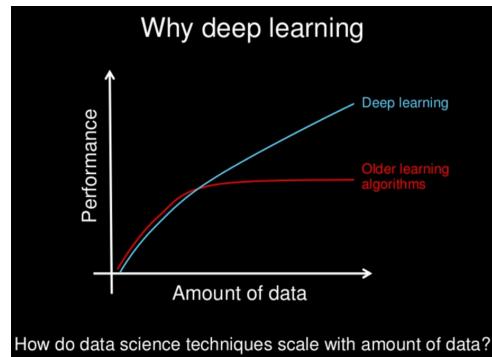


FIGURE 2.2 – La performances des modèles standards vs. DL en fonction de la taille de la base d'apprentissage. source : Andrew Ng's course.

2.5 La composition et principe des réseaux de neurones artificiels

Un réseau de neurones est une succession de couches ; une couche i contient N_i neurones qui sont reliés aux N_{i+1} neurones de la couche suivante. Le fonctionnement d'un neurone artificiel j ressemble au celui d'un neurone réel à l'instant t , le neurone j reçoit des données provenant des neurones de la couche précédente, ces données subissent un traitement, l'information en sortie est transmise par la suite aux neurones de la couche suivante. La dernière couche du réseau contient un seul neurone dans le cas d'un problème de régression ou $k > 1$ neurones correspondant au nombre de classes dans le cas d'une classification. La figure suivante montre l'architecture d'un réseau de neurone simple contenant une couche d'entrée, une couche cachée et une couche de sortie.

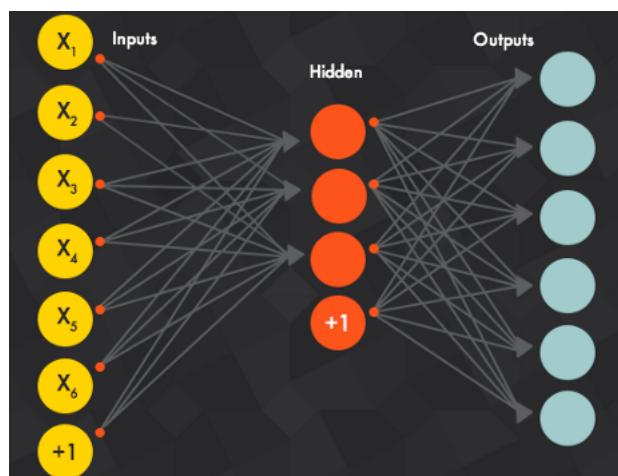


FIGURE 2.3 – Un simple réseau de neurones avec 3 couches.

2.5.1 Fonction de combinaison

Une fonction de combinaison est une application linéaire appliquée aux données provenant des neurones de la couche $i - 1$ précédente avant qu'elles soient traitées par un neurone j de la couche i . C'est une combinaison linéaire des sorties des neurones de la couche précédente où les coefficients correspondent aux poids synaptiques associés ; ces coefficients sont propres aux connexions entre les neurones, autrement dit chaque connexion entre deux nœuds correspond en réalité à un poids synaptique. L'expression ci-dessous représente la formule de la fonction de combinaison d'un nœud j .

$$h = \sum_i W_i \times X_i + b_j \quad (2.1)$$

W_i : le poids associé à l'information X_i provenant du neurone i de la couche précédente.
 b_j : une constante propre au neurone j appelée bias.

2.5.2 Fonction d'activation

Une fonction d'activation est une fonction non-linéaire qui vient donc pour casser la linéarité du réseau de neurone. Elle correspond donc au traitement de l'information effectué dans un neurone. En effet la fonction d'activation décide si le neurone doit être activé ou non d'où l'appellation.

Il existe plusieurs fonctions d'activation dans la littérature parmi les plus connues on cite :

1. **Step function** : C'est la fonction d'activation la plus simple.

$$a = \begin{cases} 0, & \text{si } h \leq c, c \in R \\ 1, & \text{sinon} \end{cases}$$

2. **Sigmoid** : C'est l'une des fonctions d'activation les plus utilisées, les valeurs sont comprises entre 0 et 1.

$$a = \frac{1}{1 + e^{-h}} \quad (2.2)$$

3. **Tanh** : Elle ressemble à la fonction précédente, sauf pour les valeurs de la fonction qui sont comprises en -1 et 1 .

$$a = \frac{2}{1 + e^{-2h}} - 1 \quad (2.3)$$

4. **ReLU (Rectified Linear Unit)** : C'est la fonction d'activation la plus utilisée dans les réseaux de neurones convolutionnels car elle est moins couteuse que les deux fonctions précédentes, de plus elle donne des meilleurs résultats.

$$a = \max(0, h) \quad (2.4)$$

2.5.3 Les poids synaptiques (weights and bias)

Ils représentent l'essence des réseaux de neurones. Ce sont des coefficients associés aux informations qui circulent dans le réseau. En effet un poids est associé à chaque connexion entre deux neurones, le modèle se sert de ces poids et des bias pour calculer les fonctions de combinaison pour aboutir au résultat final.

Les poids synaptiques tout comme les bias sont mis à jour au cours de l'apprentissage pour minimiser la fonction cout, autrement dit pour améliorer le modèle. La figure ci-dessous résume le traitement d'information dans un neurone.

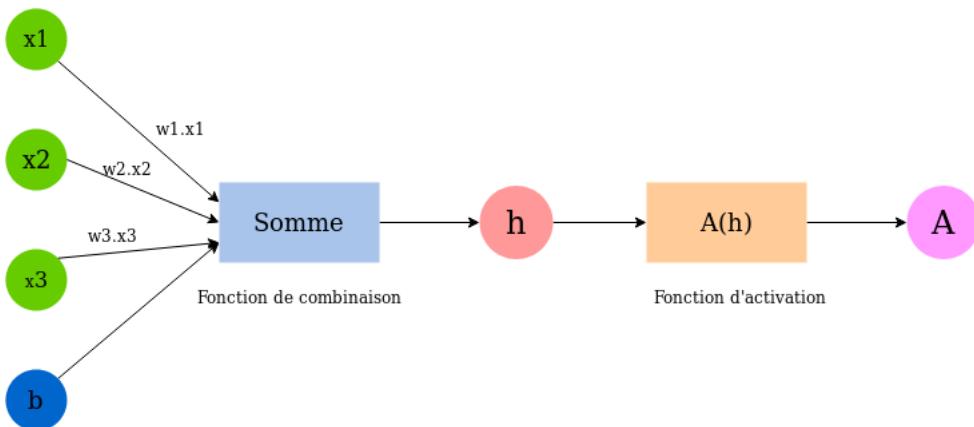


FIGURE 2.4 – Traitement effectué par un neurone.

2.6 L'apprentissage d'un réseau de neurones

Les paramètres les plus importants d'un réseau de neurones sont les poids synaptiques et le seuil de chaque neurone (bias), et la façon de les ajuster. Ce sont eux qui déterminent l'évolution du réseau en fonction de ses informations d'entrée. Il faut choisir un mécanisme permettant de les calculer et de les faire converger si possible vers une valeur assurant une classification aussi proche que possible de l'optimale. C'est ce qu'on nomme la phase d'apprentissage du réseau. Dans un modèle de réseaux de neurones formels, apprendre revient donc à déterminer les coefficients synaptiques les plus adaptés à classifier les exemples présentés.

L'apprentissage d'un réseau de neurones passe donc par plusieurs étapes, pour comprendre ces étapes et l'objectif que l'apprentissage tente d'atteindre, nous allons présenter quelques définitions importantes à la compréhension du processus d'apprentissage.

2.6.1 Fonction objectif

La donnée en sortie d'un réseau de neurone est complètement dépourvue de sens si elle n'est pas évaluée ou comparée au résultat attendu. La fonction objectif exprime la qualité de prédiction d'un réseau de neurone en termes de capacité à prédire le résultat attendu.

Si les prédictions du réseau s'écartent trop des résultats attendus , la fonction objectif retourne une valeur très grande qui signifie que les prédictions ne correspondent pas aux résultats attendus. Progressivement, la fonction objectif est minimisée à l'aide d'un algorithme d'optimisation pour aboutir aux résultats escomptés.

Le choix d'une fonction objectif est primordial car deux fonctions objectif différentes donnent deux valeurs d'erreur (résultats) différents, ceci aura un effet direct sur le processus d'apprentissage et la qualité du modèle. Par conséquent plusieurs facteurs peuvent motiver le choix de la fonction objectif parmi ces facteurs : la nature du problème initial (régression ou classification), la facilité de calcul du gradient de la fonction objectif et la nature de la base d'apprentissage.

Il existe un bon nombre de fonctions objectif qui peuvent être classées en deux catégories selon la nature du problème initial.Dans ce qui suit on pose :

n : le nombre de données dans la base d'apprentissage.

i : la i -ème donnée dans la base d'apprentissage.

y_i : l'étiquette associé à la i -ème donnée dans la base d'apprentissage (résultat attendu).

\hat{y}_i : la prédiction de la i -ème donnée dans la base d'apprentissage.

1. Fonction objectif pour un problème de régression

Dans un problème de régression on tente de prédire une valeur continue telle que le prix d'une maison , précipitation de pluie dans une région ... etc.Parmi les fonctions objectif les plus utilisées dans les problèmes de régression on trouve :

(a) Erreur quadratique moyenne (Mean squared error)

Comme son nom l'indique, elle exprime la moyenne quadratique de la différence entre la prédiction du modèle et la valeur réel.Elle est très utilisée à cause

de sa simplicité mais surtout à cause de son efficacité. En effet l'introduction du carré dans la fonction permet de mieux pénaliser les mauvaises prédictions ; plus la prédiction s'éloigne de la valeur réelle plus la valeur retournée par la fonction est grande et inversement. De plus la simplicité de cette fonction facilite le calcul du gradient lors de l'apprentissage. La formule mathématique de la fonction :

$$erreur = \frac{\sum_i (y_i - \hat{y}_i)^2}{n} \quad (2.5)$$

(b) Moyenne des valeurs absolues des écarts (Mean Absolute Error)

Elle exprime la moyenne des valeurs absolues des écarts. Contrairement à l'erreur quadratique moyenne cette fonction nécessite quelques opérations pour le calcul du gradient. Cependant elle est très utile pour les données qui s'écartent fortement des autres données ou observations de la base d'apprentissage (données aberrantes) en raison d'absence du carré. La formule mathématique :

$$erreur = \frac{\sum_i |y_i - \hat{y}_i|}{n} \quad (2.6)$$

2. Fonction objectif pour un problème de classification

Dans un problème de classification on essaie de prédire la classe à laquelle appartient la donnée en entrée i.e. la valeur en sortie est une valeur discrète. Ex : la reconnaissance de l'écriture manuscrite des chiffres où on essaie de reconnaître le chiffre illustré dans l'image ; la valeur prédit par le modèle est donc discrète et comprise entre 0 et 9. La fonction objectif la plus courante dans les problèmes de classification est :

— Cross Entropy Loss

Cette fonction exprime à quel point les prédictions du modèles et les résultats attendus se ressemblent. L'avantage principal de la fonction est qu'elle pénalise sévèrement les prédictions qui sont loins de la valeur réelle. Par exemple si la valeur de prédiction est de 0.02 alors que la valeur réelle est de 1, la valeur retournée par cette fonction est donc très grande. Ceci permet en effet un meilleur apprentissage. La formule mathématique :

$$erreur = - \sum_i^C y_i \log \hat{y}_i \quad (2.7)$$

2.6.2 Forward propagation

Ce terme désigne la propagation d'une ou plusieurs données de la base d'apprentissage à travers le réseau de neurones, C'est le cœur du processus d'apprentissage. La donnée est injectée dans le réseau de neurones, ceci constitue la couche d'entrée du modèle. Chaque neurone j d'une couche cachée i reçoit les données provenant des n_{i-1} neurones de la couche précédent avec les poids associés, le neurone j est d'abord pré-activé avec le calcul de la fonction de combinaison , ensuite la fonction d'activation est appliquée au résultat retourné par la pré-activation, enfin la sortie de la fonction d'activation est transmise aux neurones de la couche suivante avec les poids associés. ... jusqu'à arriver à la dernière couche, ce processus est appelé forward propagation pass.

2.6.3 Algorithme du gradient (Gradient Descent)

C'est un algorithme classique d'optimisation de la fonction objectif [16]. Il est utilisé pour trouver les valeurs optimales des poids (coefficients) du réseau qui minimise la fonction objectif.

L'algorithme du gradient est la première alternative quand les poids ne peuvent pas être calculés analytiquement (par exemple en utilisant l'algèbre linéaire) et doivent être recherchés un algorithme d'optimisation. Il permet donc de trouver la meilleure combinaison des poids qui minimise la fonction objectif. La mise à jour des poids du modèle se fait suivant la formule suivante : $W_i^t = W_i^{t-1} - \alpha \times \frac{\partial \text{erreur}}{\partial W_i^{t-1}}$ Chaque mise à jour d'un poids représente un pas vers l'optimum global, la vitesse avec laquelle l'algorithme avance est contrôlée par un certain paramètre α appelé *Learning rate*.

2.6.4 Rétropropagation du gradient (backpropagation)

L'algorithme de rétropropagation du gradient est l'essence du processus d'apprentissage, c'est une extension d'algorithme du gradient où les dérivées partielles de la fonction objectif par rapport aux poids des couches cachées sont calculées en utilisant la loi de dérivation d'une fonction composée , cela nous évitera de refaire les mêmes calculs du moment que les dérivées partielles de la fonction objectif par rapport aux poids d'une couche i dépendent de celles de la couches suivantes et qui sont calculées préalablement. Ceci permet de réduire considérablement le temps d'apprentissage.

A présent que nous avons présenté tous les outils nécessaires pour permettre à un réseau de neurones de s'entrainer, Nous allons expliquer brièvement comment est-ce qu'un réseau de neurones s'entraîne sur une base d'apprentissage.

Le processus d'apprentissage commence par l'initialisation aléatoire de tous les poids du modèle ensuite une forward propagation est effectuée pour pouvoir calculer la fonction objectif. Une fois l'erreur calculée, l'algorithme de rétropropagation du gradient est exécuté pour ajuster les poids du modèle de façon à minimiser la fonction objectif. Ces opérations sont répétées jusqu'à ce que le modèle ne puisse plus progresser ou jusqu'à un critère d'arrêt donné. La figure suivante illustre le processus d'apprentissage dans sa version la plus simple.

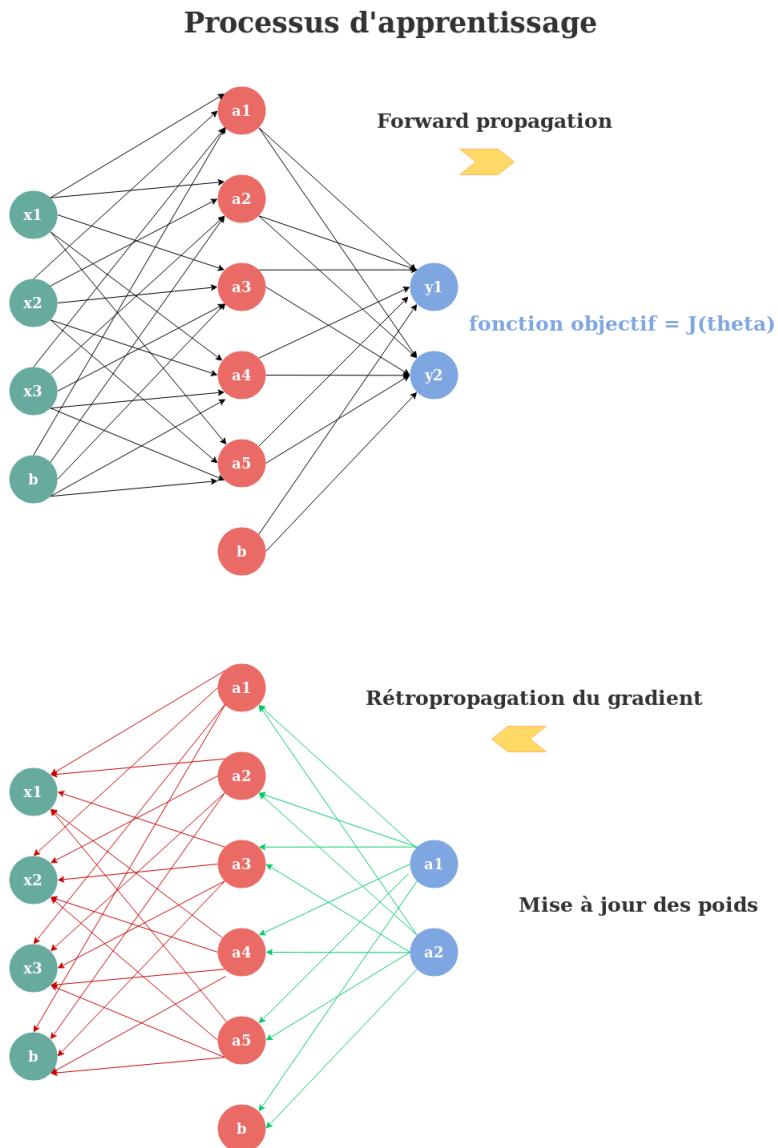


FIGURE 2.5 – Une itération du processus d'apprentissage.

2.7 Réseaux de neurones convolutionnels "CNN"

Dans un réseau de neurones classique , chaque neurone de la couche i est connecté avec tous les neurones de la couche suivante. Ceci était possible car le nombre de caractéristiques (features) des données qu'un réseau de neurones simple était censé traité était relativement petit.

Mais cela devient vite inenvisageable si le nombre de caractéristiques (**features**) des données est grand comme c'est le cas pour les images. En effet pour traiter une image de taille $224 \times 224 \times 3$ il faut un nombre de poids égal à $224 \times 224 \times 3 = 150528$ pour la première couche uniquement, ceci étant complètement absurde en raison du temps d'apprentissage énorme nécessaire pour pouvoir entraîné un modèle simple sur des images de petites tailles ($224 \times 224 \times 3$).

Une autre raison pour laquelle le réseau de neurones dans sa version la plus simple n'est pas efficace pour ce type de données est qu'il ne tient pas compte des relations spatiales qui existent entre les éléments présents dans l'image car cette dernière sera transformée en un vecteur. Il est donc judicieux de trouver une autre façon pour traiter ce type de données et qui tient en considération les relations spatiales présentes dans l'image et pouvoir extraire les caractéristiques déterminantes dans la classification de l'image en un temps raisonnable, d'où l'apparition des réseaux de neurones convolutionnels.

Un réseau de neurones convolutionnels (images) est un algorithme d'apprentissage profond qui prend en entrée une image , attribue une importance à chaque élément/objet présent dans l'image (grâce aux poids que le réseau apprend au fur et à mesure) et pouvoir les différencier. En effet les caractéristiques de l'image sont extraites grâce aux paramètres que le réseau apprend au fur et à mesure. Le but des réseaux de neurones convolutionnels est donc de transformer l'image en entrée en une telle façon à faciliter le traitement sans perdre les caractéristiques déterminantes dans la prédiction.

Un réseau de neurones convolutionnel n'est rien d'autre qu'une succession de couches de convolution et de sous-échantillonnage suivies par une succession de couches de neurones simples.

2.7.1 Les couches d'un réseau de neurones convolutionnel

Dans ce qui suit nous allons détailler le fonctionnement de chaque couche du CNN [17].

1. Couche d'entrée :

Cette couche correspond à l'image en entrée. Cette dernière est représentée par une matrice de dimension $W \times H \times C$; W pixels en largeur, H pixels en longueur et C plans de couleurs (3 si c'est une image en couleur , 1 si elle est en noir et blanc). Lors de l'apprentissage les images sont injectées en groupe dans le réseau i.e. la matrice d'entrée sera de dimension $B \times W \times H \times C$ où B est le nombre d'images injectées.

2. Couche de convolution :

Communément appelée “feature extractor” (extracteur de caractéristiques), elle constitue le cœur d'un réseau de neurones convolutionnel. La couche de convolution consiste en k filtres de convolution qui sont mis à jour au fur et à mesure lors de l'apprentissage. Les filtres sont des matrices 3D généralement de petites tailles ($3 \times 3 \times C, 5 \times 5 \times C, 7 \times 7 \times C$) où C est la profondeur de la matrice résultante de la couche précédente ex : C=3 pour les filtres de convolution de la première couche de convolution dans le cas des images en couleur.

On fait balayer chaque filtre de convolution F_i sur la matrice m provenant de la couche précédente, à chaque étape de balayage le produit scalaire entre la région de la matrice couverte par le filtre de convolution et le filtre lui-même est calculé , le produit de convolution donne lieu à une matrice de même longueur et largeur que la matrice m . Le processus est répété pour chaque filtre de la couche de convolution, les matrices qui résultent de ces produits de convolution sont superposées les unes sur les autres en formant une matrice avec une profondeur égale au nombre de filtre de la couche de convolution k et de même longueur et largeur que la matrice de la couche précédente.

Pour finir la matrice résultante est passée à une fonction d'activation le plus souvent ReLU. La sortie de la fonction est transmise à la couche suivante.

La 2.6 suivante montre le résultat des produits de convolution avec des filtres connus, dans le cas du CNN les filtres de convolution correspondent aux poids que le modèle doit apprendre.

| Operation | Filter | Convolved Image |
|----------------------------------|----------------------------------------------------------------------------------|-----------------|
| Identity | $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ | |
| Edge detection | $\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$ | |
| | $\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ | |
| | $\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$ | |
| Sharpen | $\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ | |
| Box blur (normalized) | $\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ | |
| Gaussian blur (approximation) | $\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$ | |

FIGURE 2.6 – Exemple de produits de convolutions avec des filtres connus.

3. Couche de sous-échantillonnage (Pooling)

C'est une couche qui se place généralement entre deux couches de convolution, son rôle est de réduire la taille spatiale (largeur, longueur) de la matrice m provenant de la couche précédente tout en gardant les caractéristiques les plus importantes pour la prédiction. Elle consiste à balayer chaque plan de la matrice m et en prenant la moyenne ou la valeur maximale dans une région de taille généralement égale à 2×2 pixels avec un pas de 2. Ceci résulte en une matrice de taille $(W/2, H/2, C)$. W, H, C étant respectivement la largeur, la longueur et la profondeur de la matrice M.

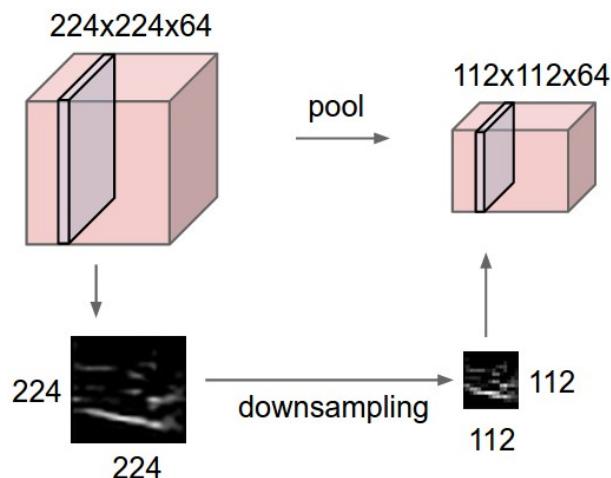


FIGURE 2.7 – Exemple d'un sous-échantillonnage en 3D.

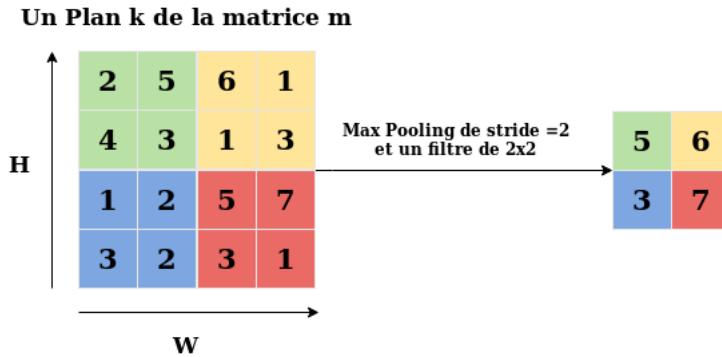


FIGURE 2.8 – Exemple d'un sous-échantillonnage Max pooling en 2D.

Toutefois il est possible de remplacer ces couches de sous-échantillonnage avec des couches de convolution de grande taille à condition que lors du balayage des filtres il ne faut pas dépasser les limites de l'image.

4. Une couche de transformation (Flatten layer)

Cette couche a pour but de transformer la dernière matrice en un vecteur de dimension 1. Ce vecteur représente en fait la couche d'entrée du réseau de neurones entièrement connecté.

5. Réseau de neurones entièrement connecté

Pour pouvoir prédire le résultat final, les caractéristiques extraites à partir de l'image en entrée doivent être injectées dans un réseau de neurones entièrement connecté qui est constitué généralement d'une couche d'entrée, une couche cachée et une couche de sortie disposée telles qu'elles sont présentées dans la partie précédente.

2.7.2 Exemple d'une architecture CNN simple

Un simple réseau de neurones convolutionnels se compose de :

- **Couche d'entrée** :représente l'image en entrée par ex : une image de taille $32 \times 32 \times 3$; une image de 32 pixels en longueur et largeur et trois plans de couleur (R,G,B).
- **Couche de convolution** :Elle contient 12 filtres de convolutions de taille 7×7 , on fait balayer chacun de ces filtres sur l'image en entrée et on calcule le produit scalaire entre la région de l'image couverte par le filtre et le filtre lui-même. Ceci a pour effet de produire 12 matrices de taille 32×32 .
- **Fonction d'activation ReLU** :une fonction d'activation est appliquée aux résultats précédents.
- **Couche de sous-échantillonnage** : Cette couche a pour but de réduire la taille de la matrice précédente tout en essayant de garder les caractéristiques importantes de l'image. Un exemple résultat qu'on peut avoir à l'issu de cette étape est 12 matrices de taille 16×16 chacune.
- **Couche de transformation** : à l'issue de cette étape les matrices précédentes sont transformées en vecteur.
- **Un réseau entièrement connecté** : Le vecteur obtenu est injecté dans un réseau de neurones entièrement connecté qui ressemble à ceux présentés plus haut.La figure ci-dessous illustre l'architecture présentée ci-dessus.

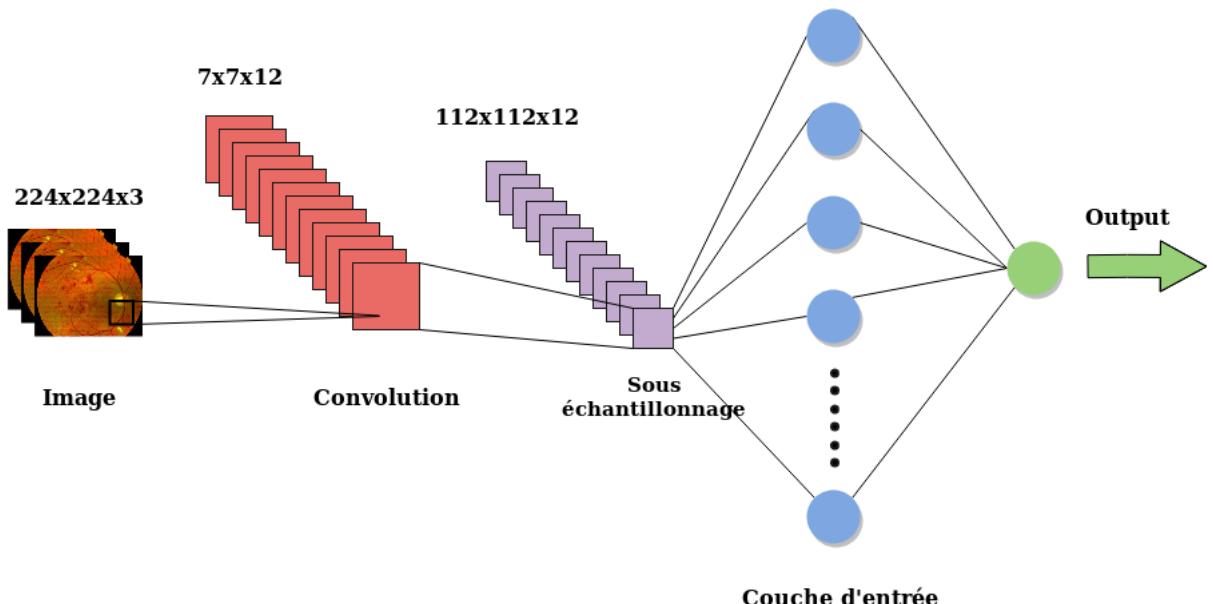


FIGURE 2.9 – Une simple architecture d'un réseau de neurones convolutionnel.

2.8 Métriques d'évaluation

Une fois le réseau de neurones convolutionnel entraîné, il doit être évalué et testé sur une base de test contenant des images de différentes classes. En effet plusieurs métriques d'évaluation peuvent être utilisées pour juger l'efficacité du modèle dans ce qui suit nous allons énumérer les différentes métriques utilisées.

1. **Précision (Accuracy)** : Elle représente la portion des images correctement classifiées sur le nombre total des images.

$$precision = \frac{VP + VN}{VP + VN + FP + FN} \quad (2.8)$$

2. **Sensibilité** : La sensibilité du modèle est sa capacité à détecter un maximum de malades (c'est-à-dire à avoir le moins de faux négatifs).

$$sensibilite = \frac{VP}{VP + FN} \quad (2.9)$$

3. **Spécificité** : La spécificité du modèle est sa capacité à ne détecter que les malades (avoir le moins de faux positifs).

$$specifique = \frac{VN}{VN + FP} \quad (2.10)$$

2.9 Conclusion

Nous avons vu dans ce chapitre le principe et le fonctionnement de base des réseaux de neurones en général et les réseaux de neurones convolutionnels en particulier. L'objectif ultime de ce chapitre était de définir tous les termes relatifs aux réseaux de neurones pour une meilleure compréhension de la suite de ce travail.

Feature based methods

3.1 Introduction

Comme son nom l'indique cette approche est basée sur l'extraction d'une ou plusieurs caractéristiques de la RD entre-autres : micro anévrismes, hémorragies, exsudats, structure des vaisseaux sanguins de la rétine, œdème maculaire . . . etc. Dans le paragraphe suivant nous allons énumérer quelques méthodes d'extraction des caractéristiques de la maladie.

3.2 Détection de la structure des vaisseaux sanguins

La rétinopathie diabétique affecte la largeur, la couleur et la forme des vaisseaux sanguins rétiniens. Lorsque cette dernière s'aggrave de plus en plus, plusieurs composants de la rétine seront privés de nourriture véhiculée par les vaisseaux sanguins, provoquant ainsi l'envoi de signaux par la rétine pour s'alimenter, ces derniers déclenchent la croissance de nouveaux vaisseaux sanguins anormaux et très fragiles. Ils se développent le long de la rétine et à la surface du gel vitreux qui remplit l'intérieur de l'œil et sont caractérisés par leurs parois minces et fragiles, Il peut en résulter une perte de vision grave, voir la cécité

La caractérisation des anomalies vasculaires est donc un moyen de distinguer les images rétiniennes normales de celles atteintes de RD. Dans ce qui suit nous allons présenter trois travaux qui emploient différentes techniques pour extraire des anomalies dans les vaisseaux sanguins rétiniens.

3.2.1 Estimer le profil sanguin avec une fonction gaussienne et sa dérivée

Dans une première méthode L.Gang et al [2] les profils des vaisseaux (diamètre, longueur) ont été estimés en utilisant une fonction gaussienne et sa dérivée seconde. Afin d'obtenir un modèle permettant de déterminer la largeur d'un vaisseau, deux méthodes ont été combinées : la détection de contours (edge detection) et les filtres adaptés (matched filters). La méthode de Canny et « Gradient Operator » sont utilisés pour détecter les contours du vaisseau. Dans le filtrage adapté, un produit de convolution est appliqué entre le profil du vaisseau et le filtre adapté. Une fois que le profil du vaisseau correspondant

au filtre adapté est trouvé, la mesure du diamètre du vaisseau est effectuée à l'aide des paramètres du modèle. Les filtres adaptés (Matched filters) sont préférables car ils aident à éliminer les petites branches aux points de bifurcation. Ensuite suivant le diamètre du vaisseau calculé on peut déterminer si l'image est atteinte par la RD ou saine

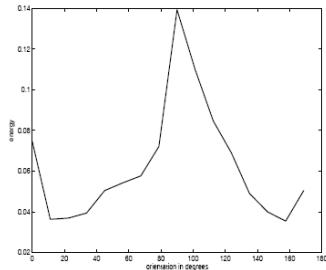
3.2.2 Filtre Kalman et les filtres gaussien adaptées

O.Chutatape et al[18] ont proposé une autre méthode où la détection et le suivi des vaisseaux sanguins dans les images rétiniennes sont effectués à l'aide de filtres gaussiens adaptés (Gaussian matched filters) et à l'aide de filtre de Kalman. Un filtre de Gauss adapté a été utilisé pour localiser le centre et la largeur du vaisseau. Tandis que le filtre de Kalman est utilisé pour estimer l'emplacement du prochain segment du vaisseau sur la base du modèle observé. Les bifurcations ont également été prises en compte par cette méthode.

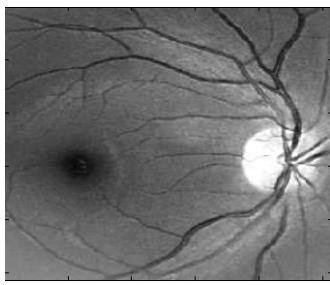
Une autre méthode est proposée par Vallabha et al[19] qui se focalise sur la détection des changements au niveau des vaisseaux observés pendant les stades avancés de la RD en utilisant l'hypothèse que les vaisseaux sanguins dans une image rétinienne normale ont une orientation dominante contrairement aux vaisseaux sanguins d'une rétine atteinte par une RD qui sont caractérisés par de multiples orientations dominantes. Ces anomalies sont détectées en traitant les images rétiniennes en utilisant des filtres de Gabor sélectifs en fonction de l'échelle et de l'orientation. La distinction entre la RD minime et la RD sévère est clairement observée dans les échelles plus fines des filtres de Gabor. Afin d'estimer la présence ou l'absence des anomalies, les résultats obtenus après l'application des filtres de Gabor de fines échelles sont représentées dans un graphe. Ce dernier montre les variations d'énergie par rapport à l'orientation. On remarque que si une image est sans anomalies vasculaires aura une orientation dominante (présence d'un seul pic) « 3.1b », alors que si l'image présente des anomalies vasculaires elle montre de multiples orientations dominantes reflétées par la présence de plusieurs pics.



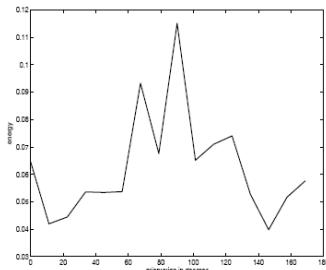
(a) Image rétinienne d'une NPDR minime



(b) Variation de l'énergie en fonction de l'orientation pour la figure 1



(c) Image rétinienne d'une NPDR sévère



(d) : Variation de l'énergie en fonction de l'orientation de la figure 3

FIGURE 3.1 – Variations de l'énergie par rapport a l'orientation

3.2.3 Conclusion

La détection des anomalies vasculaires n'est pas toujours une approche fiable et efficace pour détecter la présence de la rétinopathie diabétique car d'une part les anomalies au niveau des vaisseaux sanguins rétiniens peuvent être des signes d'une autre maladie et non pas la rétinopathie diabétique seulement, d'autre part ces anomalies vasculaires n'apparaissent qu'à un stade avancé de la maladie donc elles ne contribuent pas au dépistage précoce de la RD. De plus ces méthodes nécessitent des images du fond d'œil de très bonne qualité pour pouvoir distinguer les vaisseaux sanguins de tout type de bruit pouvant être présent dans l'image.

Ces méthodes peuvent être plus fructueuses en les combinant avec des méthodes de détection des autres lésions qui caractérisent la RD citées ci-dessous.

3.3 Détection des microanévrismes

3.3.1 Introduction

La RD se caractérise par la présence de plusieurs lésions dans la rétine tout au long de la phase de développement de cette maladie, parmi ces lésions on trouve les microanévrismes (MAs) qui apparaissent au tout début de la maladie, ils sont donc considérés comme un signe avant-coureur de la maladie ce qui fait que la détection de ces derniers est d'une importance fondamentale pour le dépistage de la RD.

Les MAs sont des gonflements anormaux des capillaires rétiniens dus à l'affaiblissement de la paroi des vaisseaux sanguins rétiniens. Bien que les capillaires ne soient pas visibles dans les images du fond d'œil, les MAs apparaissent sous forme de points d'un rouge foncé et isolés. Comme pour les vaisseaux sanguins, les MAs apparaissent avec le contraste le plus élevé dans le plan vert de l'image.

Plusieurs facteurs rendent l'identification de ces lésions très difficile pour un système automatisé entre autre : la variation de l'éclairage de l'image, la variabilité de la clarté de l'image et la texture de fond de l'image très variable. Les points sombres se rencontrent généralement sur les vaisseaux sanguins dont le contraste par rapport au fond atteint un maximum, ainsi que sur les bifurcations où les vaisseaux eux-mêmes sont indistincts. Les réflexions lumineuses sont courantes dans les images rétiniennes : des petites brèches peuvent apparaître entre les zones lumineuses et celles-ci, lorsqu'elles sont interprétées localement, nous ne pourrons pas les distinguer des MAs. Une autre difficulté est que les MAs peuvent être regroupés en cluster, de sorte que les MAs individuels ne présentent plus les caractéristiques d'un point isolé

3.3.2 Ségmentation des MA dans des Angiographies fluoroscénique

Une nouvelle stratégie de traitement d'image est proposée par Spencer et al [4] qui permet de segmenter et quantifier les microanévrismes dans des angiographies fluoroscénique.

La méthode décrit une nouvelle technique de croissance de région (technique de segmentation par approche régions) qui délimite avec précision tous les objets pouvant être interprétés comme des microanévrismes. L'analyse ultérieure des caractéristiques de morphologie et d'intensité de ces objets candidats permet de distinguer les microanévrismes des autres lésions et de la texture du fond d'œil.

En effet l'image en entrée subit plusieurs modifications et traitements au cours du processus de détection.

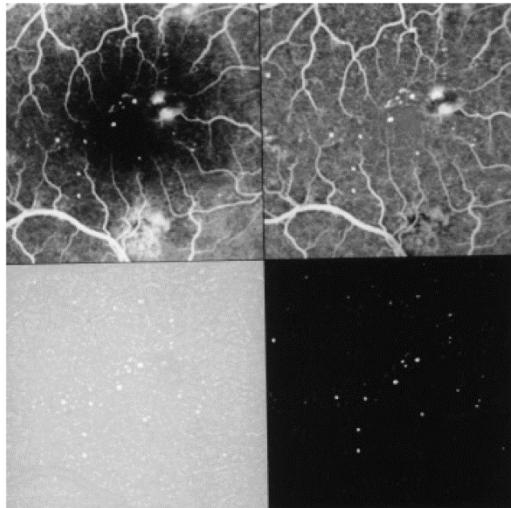


FIGURE 3.2 – Les étapes de traitement de l'image originale

1. **Correction de l'uniformité de la luminance** : Une étape de correction de l'uniformité de la luminance est effectuée et ce pour éviter d'avoir une image trop éclairé ou trop sombre.
2. **Shade-correction (correction de l'ombre)** : Les microanévrismes font partie du réseau vasculaire rétinien, chacun étant attaché au capillaire à partir duquel il s'est développé. Bien qu'il soit possible de voir le réseau capillaire complet dans l'Angiographie fluoroscénique, les vaisseaux les plus fins ne sont souvent pas visibles, pour cette raison, les microanévrismes peuvent apparaître comme des points hyper fluorescents isolés car le capillaire auquel ils sont attachés n'est pas visible (Voir figure 3.2). Par conséquent, les images de l'angiographie fluoroscénique peuvent être considérées comme ayant deux composants principaux : les principales caractéristiques des vaisseaux rétiniens visibles (microanévrismes et vaisseaux principaux) et le fond de la rétine « **background retina** » (dont l'intensité change progressivement dans l'image). Une correction de l'ombre a été utilisée pour ces deux derniers composants principaux, le résultat obtenu est illustré dans la figure 3.2.
3. **Bilinear top-hat transformation** : Les microanévrismes sont séparés des vaisseaux sanguins rétiniens en appliquant une transformation morphologique appelée « top-hat transformation » sur l'image obtenu après la correction des ombres. Cette transformation morphologique sur l'image grise était bien adaptée à la segmentation des microanévrismes des vaisseaux sanguins comme le montre la figure 3.2
4. **Filtre Adapté (Matched filtering)** : Pour affiner encore plus la segmentation des microanévrismes des autres candidats un filtre de microanévrisme adapté est

appliqué après la transformation morphologique dans l'étape précédente. Le résultat obtenu est illustré dans la figure 3.2.

5. **Seuillage d'image (Threshold)** : Un seuillage d'image est ensuite appliqué pour convertir l'image courante (en niveau de gris) en une image binaire (noir et blanc). Les objets binaires de l'image binaire obtenue sont des candidats susceptibles d'être des microanévrismes.
6. **Region-growing (croissance de régions)** : L'algorithme de croissance de région prend chaque objet binaire dans l'image binaire contenant les candidats susceptibles d'être classés comme microanévrisme et en utilisant l'image originale pour fournir des informations en niveaux de gris, il délimite la zone marquée par l'objet binaire. L'image originale a été choisie pour fournir les informations en niveaux de gris, de sorte que chaque objet binaire raffiné par l'algorithme de croissance de région soit directement lié aux données non traitées.
7. **Analyse des objets candidats** : Les objets binaires ont été analysés pour segmenter les microanévrismes des autres composants de la rétine. Un certain nombre de critères ont été utilisés pour déterminer si un objet était effectivement un microanévrisme, en se basant sur des informations dérivées à la fois de sa taille et de sa forme, en utilisant l'objet binaire défini et l'intensité du gris. En effet un intervalle de valeurs de périmètre a été défini pour éliminer toutes les lésions hors cet intervalle, un autre critère de filtrage est le « Aspect-ratio » (défini comme le rapport entre longueur et la largeur) a été utilisé pour déterminer la forme des objets binaires. Classiquement, les microanévrismes sont des lésions sphériques et apparaissent donc comme des cercles hyper fluorescents dans les angiographies fluroscénique avec un Aspect-ratio = 1

Conclusion

Cette méthode n'est applicable qu'aux images d'angiographie à la fluorescéine. Ces dernières ne sont pas souvent disponibles à cause de leur cout élevé et leur faible contribution à faire apparaître d'autres lésions qui caractérisent la RD ; en effet ce type d'examen est souvent considéré comme examen complémentaire à l'examen du fond d'œil car il fournit une meilleure vision sur les vaisseaux sanguins et donc les microanévrismes et les hémorragies simples tandis que les autres lésions restent confuses et n'apparaissent pas clairement.

D'autre part ces images ne permettent pas une distinction claire entre les microanévrismes et les hémorragies qui sont deux signes de deux stades différents de la maladie. De plus la présente méthode propose une classification binaire de la maladie et ne fournit aucun indice sur la gravité du stade actuel.

3.3.3 Détection des MA par la Normalisation local du contraste

Fleming, Alan D, et al [20] proposent une méthode qui met l'accent sur le rôle important de la normalisation local du contraste « **Local Contrast Normalization** » dans la détection des microanévrismes. Elle présente une nouvelle méthode de normalisation du contraste, Elle décrit aussi une nouvelle méthode d'exclusion des vaisseaux sanguins rétiniens dans la détection des microanévrismes.

1. **Prétraitement (Preprocessing)** : Une phase de prétraitement doit être inclus dans le processus de détection des microanévrismes les étapes suivantes sont effectuées dans l'ordre :
 - (a) Enlever le bruit en appliquant un filtre médian suivi d'un filtre gaussien.
 - (b) Une normalisation du contraste global est effectuée en divisant le résultat de l'étape précédente par l'écart type calculé.
 - (c) La papille optique est détectée en appliquant un algorithme décrit dans l'article [21] la région détectée par l'algorithme ne sera pas tenu en compte dans la suite du processus de détection des microanévrismes du fait que la présence de la papille optique peut porter à confusion.
2. **Détection des candidats potentiels (Candidate Detection)** : la détection des candidats potentiels passe par plusieurs étapes :
 - (a) Une Transformation morphologique est appliquée sur l'image originale pour extraire des points qui sont plus sombres que les points qui les entourent.
 - (b) Une fois les points les plus sombres extraits on applique un deuxième filtre gaussien.
 - (c) Ensuite, un seuillage d'image est appliqué pour éliminer les bruits qui ont été considérés comme de potentiel candidat. A l'issu de cette étape on distingue plusieurs régions où chaque région D sera réduite à un seul point q telle que $S(q) = \min_{p \in D}(S(p))$ où S est l'image originale.
3. **Croissance de région (Candidate Region Growing)** : Un algorithme de croissance de région simple est utilisé pour peaufiner le résultat de l'étape précédente ; construire des régions à partir des points q en rajoutant à rajoutant tous les points avoisinants ressemblants à q . Ainsi tous les candidats potentiels seront mis en évidences.
4. **Evaluation des candidats (Candidate Evaluation)** : Pour mieux représenter les candidats on essaie d'épouser la forme de chaque candidat par une ellipse caractérisée par un certain nombre de paramètres. Ces paramètres servent pour calculer d'autres paramètres utiles pour déterminer les caractéristiques de chaque candidat. Plusieurs caractéristiques (la largeur de l'ellipse, l'énergie du candidat) sont définies pour être utilisées dans la classification.

5. **Watershed Retinal Region Growing** : L'objectif de cette étape est de délimiter les régions de la rétine qui se trouve autour de chaque candidat pour être utilisé dans la normalisation locale du contraste. Pour des raisons d'efficacité, le traitement n'est appliqué qu'à une sous-image I' centrée sur q extraite de l'image originale. Pour réduire le nombre de régions générées par l'algorithme de région de croissance basé sur les Ligne de partage des eaux « Watershed » la fonction h-transform est appliquée à I' . L'algorithme de région de croissance basé sur les Ligne de partage des eaux « Watershed » est ensuite exécuté sur $HMIN(I')$ divisant l'image I' en plusieurs régions comme le montre la figure suivante.
6. **Évaluation des caractéristiques de contraste normalisées** : Une normalisation locale du contraste est ensuite appliquée aux régions générées par l'algorithme précédent.
7. **Classification** : Un ensemble de 71 images a été utilisé dont 7 images ne contenaient pas des MAs. La méthode proposée a identifié 1417 candidats potentiels, pour chaque candidat potentiel un certain nombre de caractéristique sont calculées, les valeurs obtenus sont transformées de sorte que la distribution de ces dernières soit approximativement uniforme dans l'intervalle [0,1]. Ces valeurs servent à entraîner un modèle basé sur les k plus proches voisins (k-NN classifier) avec k = 15.

Avantages et critiques

La caractérisation de cette méthode est l'utilisation d'une technique de normalisation locale du contraste après l'élimination des bruits présents sur l'image en utilisant un filtre médian et l'élimination des vaisseaux sanguins. En effet la normalisation locale du contraste autour des candidats potentiels permet de mieux faire apparaître ces derniers et de les mieux distinguer de la surface voisine. Toutefois si l'élimination des bruits et des vaisseaux sanguins ne procure pas des résultats pertinents ; autrement dit la phase d'élimination a échoué d'éliminer les bruits qui peuvent porter à confusion, ceci dit la normalisation locale du contraste pourrait avoir un effet paradoxal en amplifiant les bruits qui seront considérés comme des candidats potentiels, ce qui peut fausser les résultats

De même que la méthode précédente, elle ne permet pas une distinction claire entre les hémorragies et les microanévrismes, de plus la classification binaire ne procure aucun indice sur la gravité du stade actuel de la maladie

3.4 Détection des exsudats

3.4.1 Introduction

Les exsudats sont des accumulations de lipides et de protéines dans la rétine, il s'agit généralement de lésions brillantes, réfléchissantes, blanches ou crème qui sont visibles sur la rétine. La présence de telles lésions témoigne de l'augmentation de la perméabilité des vaisseaux et un risque d'œdème de la rétine[6]. Bien qu'ils ne représentent pas un danger pour la vue, ils constituent un signe de l'accumulation de liquide organique dans la rétine. Cependant, s'ils apparaissent près du centre de la macula, ils sont considérés comme des lésions menaçant la vue. La plupart du temps, ils sont associés à des microanévrismes.

Les exsudats apparaissent généralement dans une couleur jaunâtre ou blanche avec différentes tailles, formes et emplacements. On les voit souvent sous forme de taches isolées ou regroupées dans des clusters ou sous forme d'anneau circulaire d'exsudats avec des microanévrismes au centre de l'anneau. La figure ci-dessous montre un exemple de détection d'exsudats Dans un œil sain et un œil atteint par une RD proliférante. Les points blancs indiquent les zones où des exsudats ont été détectés. Une étape importante du processus d'extraction consiste à retirer les principales structures de la rétine, telles que l'arbre des vaisseaux sanguins et le disque optique. Après la suppression de ces structures, les exsudats seront détectés à l'aide d'une séquence d'algorithmes de traitement d'images

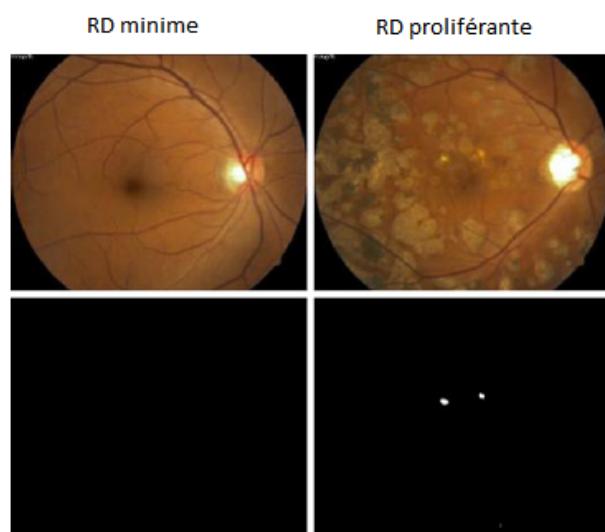


FIGURE 3.3 – Résultats de la détection des exsudats

Plusieurs techniques ont été appliquées pour la détection de ces lésions, on distingue des méthodes qui se basent sur les réseaux de neurones, le seuillage d'image, des algorithmes de croissance de régions, la normalisation des couleurs et ajustement du contraste suivi

d'un fuzzy c-mean clustering. Dans la suite nous allons présenter une méthode de détection d'excédent introduite par Sopharak, Akara, et al[22] basée sur des techniques de la morphologie mathématique sur des images rétiniennes de pupilles non dilatées qui sont des images de faible qualité. De plus la méthode décrite ci-dessous présente l'avantage de localiser les excédents sur la base de la position du macula.

Le processus de détection passe par plusieurs étapes décrites ci-dessous :

1. Prétraitement (Preprocessing)

- Passage de l'espace de couleur de RVB (RGB en anglais)¹ au HSL (Teint,saturation,luminosité)² car ce dernier permet de séparer le canal luminosité des deux autres canaux.
- Appliquer un **filtre médian** sur le canal luminosité pour réduire le bruit.
- Une amélioration du contraste en effectuant une égalisation d'histogramme communément appelée Égalisation adaptative de l'histogramme (Adaptive histogram equalization) sur des petites zones,cette étape sert à mettre en évidence les régions lumineuses (les excédents et la papille optique) en leurs attribuant des valeurs de luminosité plus élevées que les autres régions.

2. Élimination de la papille optique

La papille optique elle aussi apparaît comme une région très lumineuse, elle peut donc être considérée comme un grand excédent par l'algorithme de détection de ce fait son élimination est très importante dans le processus de détection des excédents. dans ce qui suit nous décrivons les étapes de l'élimination de la papille optique :

- Un opérateur morphologique (ouverture ϕ) avec un élément structurant B_1 sous forme d'un cercle d'un diamètre de 8 pixels est appliqué au canal de luminosité f_I (intensity channel) de l'image originale , ceci permettra d'éliminer les vaisseaux présents dans la papille optique. $OP_1 = \phi_{B_1}(f_I)$ le résultat de cette opération est illustré dans la 3.4(a).
- Un seuillage (un seuil α_1) est effectué sur l'image obtenu à l'issu de l'étape précédente, le résultat de cette opération(3.4(b)) sera utilisé comme masque dans l'étape suivante.
- Les pixels du masque précédent sont tous intervertis avant d'être superposés sur l'image originale pour supprimer les zones lumineuses candidates.Le résultat de cette opération OP_2 est illustré dans la 3.4(c).
- Une reconstruction morphologique par dilatation R est appliquée sur l'image précédente $OP_3(x) = R_{f_i}(OP_2)$, le résultat de la reconstruction par dilatation

1. Espace colorimétrique où chaque point représentant une couleur a trois valeurs des couleurs primaires : du rouge, du vert et du bleu.

2. représentation alternative du modèle de couleur RVB, conçu dans les années 1970 par des chercheurs en infographie pour s'aligner plus étroitement avec la façon dont la vision humaine perçoit les attributs de création de couleurs

est illustré dans la figure 5(d).

- 3.4 (d) est soustraite de l'image originale f_I , Un seuillage d'image est appliqué à l'image obtenu avec un seuil α_2 déterminé par l'algorithme de Otsu[23] $OP_4 = T_{\alpha_2}(f_I - OP_3)$. Par conséquent les intensités(lumières) élevées sont reconstituées tandis que le reste est éliminé, comme indiqué dans la 3.4 (e).
- Dans la majorité des cas la papille optique peut être identifié facilement comme le plus grand candidat restant. cependant dans certains cas les exsudats sont volumineux donc il peut y avoir certaines zones de l'image plus grandes que la papille optique. Comme cette dernière apparaît sous une forme ronde, Elle est donc identifié comme la plus grande région dont la forme est circulaire. Une fois que la région est déterminée elle sera supprimée de l'image originale. le résultat final est représenté par la 3.4 (f).

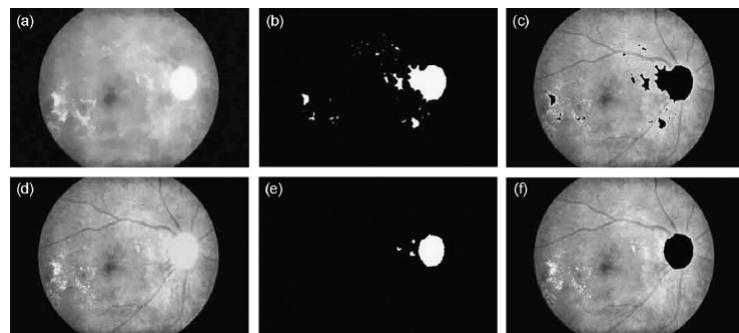


Fig. 3. (a) Intensity image after closing. (b) thresholded image. (c) marker image. (d) reconstructed image. (e) thresholded result of difference image and (f) optic disc area eliminated from the contrast enhanced image.

FIGURE 3.4 – Les étapes du processus d'élimination de la papille optique

3. Détection des exsudats

- Comme dans la phase précédente les vaisseaux avec un contraste élevé sont éliminés en appliquant une fermeture morphologique. L'image obtenu est divisée en plusieurs régions de taille 7×7 pixels, pour chaque région on calcule son écart type et les pixels de la région en question prendront la valeur de l'écart type correspondant, comme le montre la 3.5 (a). La nouvelle image obtenue subit un seuillage avec un seuil Alpha3 déterminé lui aussi par l'algorithme d'Otsu. Une autre dilatation est appliquée pour s'assurer que tous les pixels avoisinants chaque région marquée dans le résultat précédent soit inclus dans la région candidate correspondante 3.5 (b).
- Les régions candidates représentants des potentiels exsudats doivent être marquées entièrement et non pas par leurs contours seulement, pour cela on applique l'algorithme de remplissage par diffusion, le résultat est illustré dans la 3.5 (c).
- La 3.5 (c) montre que la présence d'une région candidate à la place précédemment occupée par la papille optique, cela est dû à la dilatation appliquée lors

du processus d'élimination de la papille optique, pour remédier à cela, un autre opérateur morphologique est appliqué pour éliminer cette région figure10(d).

- L'image obtenu à l'issu de l'étape précédente sera utilisée comme masque montrant tous les candidats potentiels 3.5 (e).
- Comme dans le processus de la détection de la papille optique, Une reconstruction morphologique par dilatation R est appliquée sur l'image précédente 3.5 (e), le résultat de la reconstruction par dilatation est illustré dans la 3.5 (f).
- La 3.5 (f) est soustraite de l'image originale f_i , un seuillage d'image est ensuite appliqué à l'image obtenu avec un seuil α_4 . 3.5 (g).
- L'image originale et 3.5 (g) contenant les candidats potentiels sont superposées mettant ainsi en lumière tous les exsudats de la rétine. Comme le montre la 3.5 (h).

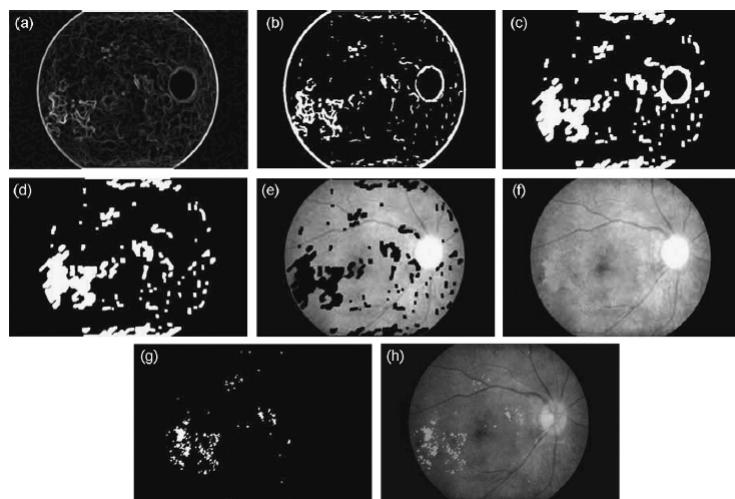


Fig. 4. Exudates detection of “image 1”. (a) Local variation image, (b) thresholded image, (c) enclosed areas were flood-filled, (d) optic disc was removed from image, (e) marker image, (f) reconstructed image, (g) difference image AND (h) result superimposed on the original image.

FIGURE 3.5 – Les étapes de détection des exsudats.

Les résultats de cette méthode montrent qu'un nombre considérable de candidats qui ont été faussement classés comme des exsudats aussi bien dans des images atteintes de la RD que des images saines, cela est dû principalement à la présence de bruit dans l'image.

L'une des principales faiblesses de cette méthode vient du fait que l'algorithme dépend des résultats d'autres tâches, à savoir la détection du disque optique et l'élimination des vaisseaux sanguins. Les résultats de la détection des exsudats dépendent donc de l'efficacité des résultats de ces tâches.

L'évaluation des résultats de détection des exsudats ne repose pas sur des critères d'évaluation robuste. Ceci dit l'efficacité d'une telle méthode n'est pas toujours garantie lorsqu'elle est appliquée à d'autres images. Il est plus judicieux donc de combiner la

présente méthode avec les autres méthodes de détection pour aboutir à de meilleurs résultats de classification.

3.5 Discussion sur les feature-based methods

- Toutes les méthodes présentées précédemment utilisent des techniques de traitement d'image telles que le prétraitement (normalisation du contraste par ex), les techniques de segmentation, les filtres 2D adaptés (Matched Filters), le seuillage d'image ... etc. Ces techniques ont été largement utilisées pour la détection des lésions caractérisant la RD. Bien que les systèmes discutés précédemment soient utiles dans l'analyse des caractéristiques spécifiques de la rétine, ils n'intègrent pas toutes les caractéristiques de la rétinopathie diabétique en une seule entité et donc ne fournissent pas un système dans son ensemble pour la classification des différents stades de la RD.
- Toutes les méthodes présentées proposent une classification binaire de la RD et ne procurent donc aucun indice sur la gravité du stade actuel de la maladie.
- Toutes les évaluations des méthodes précédentes ont été faites sur des petits ensembles d'images. Jagadish Nayak et al²⁵ énoncent qu'un grand nombre de méthodes basées sur les caractéristiques (feature-based methods) ont échoué lorsqu'elles sont appliquées sur un grand ensemble d'images anormales du fond d'œil.
- Comme la RD se caractérise par la présence de plusieurs lésions au même temps, l'identification de chaque lésion séparément ne permet pas donc d'avoir des résultats pertinents. La combinaison de ces méthodes semble donc être une solution à ce problème, cependant si ces méthodes sont combinées ensemble le temps nécessaire pour la classification d'une image donnée devient phénoménal car il faut détecter tous les lésions présentes dans l'image séparément et puis combiner les caractéristiques de ces lésions pour pouvoir classer l'image en question.
- L'efficacité de ces méthodes dépend principalement de la qualité des images du fond d'œil. L'image du fond d'œil du patient doit être suffisamment claire pour montrer les détails de la rétine. Les images de faible qualité (éclairage non uniforme, faible contraste, flou ou image pâle) donnent des résultats très médiocres même lorsque des techniques d'amélioration de qualité sont précédemment appliquées[22].

Introduction des réseaux de neurones dans la détection et la classification de la rétinopathie diabétique (RD)

Les méthodes présentées dans la section précédente reposent toutes sur la détection des lésions caractérisant la RD en utilisant un ensemble de techniques de traitement d'images. Ces lésions sont mises en lumière par ces algorithmes pour en extraire quelques propriétés importantes dans la classification qui est dans la grande majorité des cas une classification à 2 classes uniquement : **Rétinopathie Diabétique non-proliférante et Rétinopathie Diabétique Proliférante.**

L'apprentissage profond ou Deep Learning désigne l'ensemble des techniques de Machine Learning, une sous-branche de l'intelligence artificielle qui vise à construire automatiquement des connaissances à partir d'un très grand ensemble de connaissances appelé base d'apprentissage. Le succès de ces techniques leur a conféré un rôle important dans le monde contemporain ; Ces techniques sont utilisées quasiment dans toutes les applications pratiques (reconnaissance des visages et de la parole, traduction automatique des textes, voiture autonome, ... etc.). Le domaine de l'imagerie médicale n'échappe pas à la règle car on trouve de plus en plus des architectures de réseaux de neurones spécialement adaptées pour la détection des lésions cérébrales, le dépistage précoce du cancer du sein, détection et classification de la Rétinopathie Diabétique, ... etc.

La classification binaire de la RD a fait l'objet de plusieurs recherches, plusieurs travaux ont été menés à cet égard et ont abouti à des résultats satisfaisants.

Gardner et al [24] ont utilisé des réseaux de neurones simples. Ils ont obtenu des valeurs de sensibilité et de spécificité relativement élevées (88,4 % et 83,5 % respectivement) pour une classification binaire de la maladie (pas de RD, RD). Ils ont utilisé une petite base d'apprentissage d'environ 200 images et ont divisé chaque image en plusieurs régions, puis un clinicien qualifié va classer chacune de ces régions avant la mise en œuvre du SVM (support vector machine).

Les réseaux de neurones ont également été utilisés dans la classification en trois classes de la rétinopathie diabétique. Nayak et al[25] ont utilisé des caractéristiques telle la superficie des exsudats et des vaisseaux sanguins ainsi que des paramètres de texture. Ces données sont injectées dans un réseau de neurones pour classer les images en **œil sain**, **rétinopathie non proliférante** et **rétinopathie proliférante**. Les résultats de la détection automatique ont été validés en les comparant avec la classification effectuée par des ophtalmologues experts. Ils ont abouti à une précision de classification de 93 %, une sensibilité de 90 % et une spécificité de 100 %. Une base d'apprentissage de 140 images a été utilisée et une extraction des caractéristiques (exsudats, microanévrismes, hémorragies, ... etc.) a été requise pour toutes les images, ce qui s'est avéré un peu couteux en temps d'exécution.

La SVM (machine à vecteurs de support) est la première technique utilisée pour la classification multi-classe (5 classes) de la rétinopathie diabétique. Acharya et al [25] ont proposé une nouvelle méthode de classification de la RD en cinq classes et ce en cherchant les zones de la rétine qui comporte plusieurs caractéristiques de la RD telles que les hémorragies, les micro-anévrismes, les exsudats et les vaisseaux sanguins présentant des anomalies à l'aide de techniques de traitement d'images. Celles-ci ont ensuite été transmises au SVM pour la classification. Le système proposé leur a permis d'obtenir une sensibilité de 82 %, une spécificité de 86 % et une précision de 85,9 %. Le modèle proposé a été entraîné sur une base d'apprentissage relativement petite. La baisse de sensibilité et de spécificité est probablement due à la nature complexe de la classification multi-classe.

Toutes les méthodes précédentes intègrent obligatoirement une phase de détection des principales caractéristiques dont le résultat sera injecté dans les entrées du SVM. Malgré les résultats satisfaisants qu'elles ont pu apporter elles sont difficiles à appliquer en temps réel car d'une part leurs temps d'exécution est relativement élevé (Traitement d'image) et d'autre part, la validation de ces méthodes repose sur des ensembles de test ne dépassant pas les 100 images.

Dans la suite de cette section nous allons passer en revue les travaux les plus pertinents dans la détection (et classification) automatique de la RD en utilisant les réseaux de neurones.

4.1 Méthode 1 : Convolutional Neural Networks for Diabetic Retinopathy[26]

Cette méthode proposée par Pratt, Harry, et al repose sur l'utilisation d'un réseau de neurones convolutionnel «CNN» pour diagnostiquer la RD à partir d'images numériques du fond d'œil et les classer avec précision selon la gravité de la RD décrite par l'image. La méthode a eu recours à une augmentation des données «Data Augmentation» qui est une technique souvent utilisée pour enrichir la base d'apprentissage et prévenir le surapprentissage, permettant ainsi au modèle de mieux généraliser.

4.1.1 Les réseaux de neurones convolutionnels ou Convolutional neural network «CNN»

Les réseaux de neurones convolutionnels (CNN), une branche de l'apprentissage profond, un CNN est un type de réseau de neurones artificiels acyclique dans lequel le motif de connexion entre les neurones est inspiré par le cortex visuel des animaux, dont la particularité est la présence de quelques couches (convolution, max pooling) supplémentaires par rapport à un réseau de neurones simple. Les CNNs sont connus pour avoir réalisé des résultats impressionnantes dans l'analyse et l'interprétation d'images, y compris l'imagerie médicale. Les architectures des CNNs sont conçues pour fonctionner avec des données de type image, de telles architectures ont été construites pour la première fois dans les années 1970 avec des applications utiles et ont pu surpasser les autres approches de tâches complexes telles que la reconnaissance de caractères manuscrite. Cependant, ce n'est qu'après avoir apporté plusieurs modifications dans les réseaux de neurones, telles que l'implémentation du dropout¹, l'apparition de la fonction d'activation ReLu (Rectified Linear unit) mais surtout l'accroissement de la puissance de calcul via les unités de traitement graphique (GPU) que les CNNs sont devenus appropriés pour des problèmes de reconnaissance d'image plus complexes. À l'heure actuelle, les CNNs sont utilisés pour mener à bien des tâches de reconnaissance d'image très complexes avec de nombreuses classes d'objets avec une précision impressionnante.

4.1.2 Problème de surapprentissage

Lors de l'utilisation des CNNs pour un problème donné et plus particulièrement dans la classification multi-classes, deux problèmes se posent : Obtenir des valeurs de sensibilité (patients correctement identifiés comme ayant une RD) et de spécificité (patients correctement identifiés comme n'ayant pas de RD) relativement élevées avec un écart raisonnable entre les valeurs des deux métriques. L'autre problème qui se pose est le surapprentissage.

1. fait référence à l'abandon d'unités (à la fois cachées et visibles) dans un réseau de neurones.

Quand les bases d'apprentissage sont déséquilibrées (l'écart entre les nombres d'images de chaque classe est important) qui est généralement le cas dans les grandes bases d'apprentissage, l'algorithme d'apprentissage tend à dépendre des images de certaines classes ce qui cause un surapprentissage dans les classes les plus volumineuses. Dans la base d'apprentissage utilisée seulement 3 % des images proviennent des classes 4 et 5 (RD sévère et RD proliférante). Des changements doivent apporter pour s'assurer que l'algorithme apprenne toujours les caractéristiques de ces deux dernières classes.

4.1.3 Structure du réseau de neurones

La structure du réseau de neurones convolutionnels utilisé est présentée dans la figure suivante :



FIGURE 4.1 – Structure du réseau.

Le réseau de neurones convolutionnels peut être divisé en deux sous réseaux de neurones ; un premier réseau de neurones (Bleu) comportant que des couches de convolutions et MaxPooling relié à un autre réseau de neurones(rouge) qui se compose uniquement des couches simples et qui sont entièrement connectées.

On constate que le CNN ci-contre comporte un grand nombre de couches de convolutions ceci a été justifié par : «Plus le réseau de neurones est profond plus il apprend de nouvelles caractéristiques». Les premiers blocs du réseau comportent chacun des couches de convolution où chaque couche de convolution est suivie d'une couche de correction de type ReLu et une Batch-Normalization ainsi qu'un max pooling pour réduire la taille des

images générées à l'issu de chaque étape de convolution. À mesure que les couches de convolution deviennent profondes la Batch-Normalization est appliquée une fois dans le bloc. Comme le montre la figure ci-contre.

Pour éviter le surapprentissage et pour remédier au problème de la base d'apprentissage déséquilibrée,ils ont utilisé la technique de *class_weights* qui consiste à affecter un poids à chaque classe ces poids seront tenus en considération lors de la phase d'apprentissage. La technique de Dropout est également appliquée aux couches complètement connectées. Ce qui suit une fiche des caractéristiques du réseau de neurones ci-contre.

Fonction d'activation :ReLU pour les couches intermédiaires et softmax pour la dernière couche.

Regularization Method :L2 Regularization.

Méthode d'initialisation des paramètres :Gaussian.

Fonction objectif :Categorical cross-entropy.

4.1.4 Prétraitement

Comme la base d'apprentissage contient des images de fond d'œil des patients de différentes ethnies, classe d'âge et des niveaux de luminance extrêmement variés, il est donc judicieux d'effectuer une «color normalization» avant d'injecter ces images dans le réseau de neurones. Les images sont aussi redimensionnées pour qu'elles aient une taille réduite pour accélérer l'apprentissage.

4.1.5 Training

Le CNN est initialement «pré-entraîné» sur uniquement 10,920 images initiales, et ce pour obtenir un résultat de classification relativement rapide sans perdre beaucoup de temps d'entraînement «training time». Après 120 épisodes(epochs) d'entraînement sur les images initiales le réseau de neurones est entraîné par la suite sur les 78000 images restantes pour 20 epochs. La technique de *class_weights* présentée précédemment a été utilisée pour éviter le surapprentissage. L'algorithme de Stochastic Gradient descent a été utilisé pour optimiser la fonction objectif avec un Learning rate variable.

4.1.6 Data-Augmentation

Les images originales «prétraitées» ne sont utilisées qu'une seule fois lors de l'entraînement du modèle. Ensuite, une augmentation de données en temps réel a été utilisée tout au long de l'entraînement du modèle pour améliorer la capacité du modèle à localiser les caractéristiques de la RD. À chaque epoch, des modifications aléatoires sont appliquées à chaque image : rotation aléatoire de 0 à 90 degrés, inversion aléatoire horizontale ou verticale et décalages aléatoires horizontaux et verticaux. Comme le montre la figure suivante.

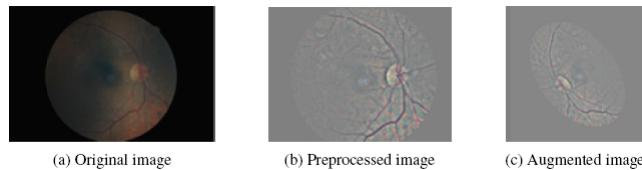


FIGURE 4.2 – Illustration du processus Data-augmentation.

4.1.7 Résultats

La validation a été effectuée sur la base de 5000 images qui ont été gardées spécialement pour cet effet. Le modèle a permis d'avoir une spécificité de 95 %, une précision de 75 % et une sensibilité de 30 %. La matrice de confusion ci-dessous montre les résultats obtenus où les classes sont numérotées de 0 à 4 : 0 – pas de RD, 1 – RD minime, 2 – RD modérée, 3 – RD sévère, 4 – RD proliférante.

| | | 0 | 1 | 2 | 3 | 4 |
|------------|-----|------|-----|-----|----|----|
| True Label | 0 | 3456 | 0 | 145 | 1 | 34 |
| | 1 | 344 | 0 | 27 | 0 | 1 |
| 2 | 543 | 0 | 179 | 5 | 40 | |
| 3 | 40 | 0 | 63 | 10 | 15 | |
| 4 | 28 | 0 | 23 | 3 | 43 | |
| | 0 | 1 | 2 | 3 | 4 | |

FIGURE 4.3 – Matrice de confusion.

Dans la présente méthode les images du fond d'œil subissent un redimensionnement et plusieurs prétraitements avant d'être injecté dans le réseau de neurones sans être rognées. Une telle approche entraîne inutilement un nombre élevé d'itérations sur des portions non informatives ce qui rend l'apprentissage du modèle inutilement long.

4.2 Méthode 2 : Transformed Representations for Convolutional Neural Networks in Diabetic Retinopathy Screening [27]

Les réseaux de neurones convolutionnels «CNN» sont connus pour être des modèles de prédiction très puissants du fait de la précision des résultats qu'ils produisent. Cependant de tels réseaux nécessitent d'immense bases d'apprentissage pour aboutir à des résultats pertinents ce qui entraîne systématiquement un temps d'apprentissage important. Ceci a motivé plusieurs travaux de recherches qui ont été menées pour réduire ce dernier.

La présente méthode propose une approche pour remédier à ce problème en s'attaquant à la base d'apprentissage au lieu d'optimiser le modèle lui-même. L'idée générale est de diviser chaque image du fond d'œil en des petites portions représentatives de tailles moindres au lieu d'utiliser l'intégralité de l'image, dans le cas idéal chaque portion représente une lésion, ces portions de petites tailles sont ensuite injectées dans le CNN où la probabilité p_i qu'une portion i soit extraite d'un œil atteint par la RD est calculé. Une fois que les probabilités p_i d'une image j soient calculées on calcule la probabilité que l'œil représenté par l'image j soit atteint par la RD qui n'est rien d'autre que la moyenne des probabilités p_i .

Cependant pour diviser une image en portions représentatives n'est pas une tâche facile dans le cas de la RD, en effet le premier problème qui se présente est les lésions caractérisant la RD différent dans leurs tailles et dans leurs formes ce qui rend le choix de la taille des portions très difficile.

La figure 4.4 illustre deux lésions de tailles et formes différentes. Supposons que nous plaçons sur chaque lésion un petit rectangle B et un autre plus grand G centrés sur la lésion et qui représentent des portions de l'image, comme le montre la 4.4 (b). La 4.4 (c) montre les images en niveau de gris correspondant aux rectangles verts. Nous notons que les caractéristiques de la grande lésion sont conservées dans G_1 . Cependant, dans G_2 , la petite lésion est considérée comme un bruit à cause de sa petite taille. En revanche si nous essayons de représenter les lésions avec des petits rectangles nous pourrions ne pas être en mesure de caractériser la grande lésion dans son contexte approprié, car son apparence à cette échelle est approximativement uniforme (B_1 sur la 4.4 (d)).

La méthode passe par plusieurs étapes avant de délivrer le résultat final. La première étape consiste à identifier d'abord des candidats potentiels pour être considérés comme des lésions en utilisant Multiscale C-MSER Segmentation, avant de les transformer en des

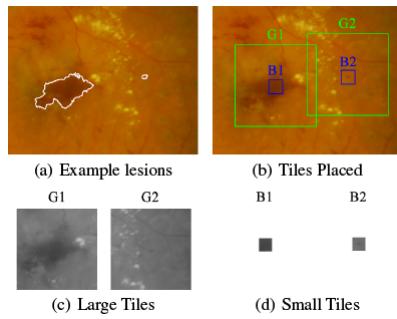


FIGURE 4.4 – Problème de la taille optimale des portions.

portions de taille fixe (Representational Transformation). Cela réduit considérablement le temps de calcul nécessaire, car un grand nombre de pixels dans l'image ne portent pas des informations utiles pour la classification et sont donc éliminés.

4.2.1 Identification des candidats(Multiscale C-MSER Segmentation)

Les MSER (maximally stable extremal regions) sont considérés comme de robustes détecteurs de région, et se base sur le principe que les régions visuellement significatives se distinguent par des frontières qui sont entièrement plus sombre ou plus lumineuse que leurs voisines. Bien que les MSER soient de fins détecteurs de régions, ils peuvent produire des régions à des échelles très similaires, en particulier lorsque les gradients (dérivées) sont subtils. Une variante de MSER, C-MSER (Lim, Lee et Hsu 2012), a été développée sur des images rétinianes. Dans cette méthode, une extension multi-échelles qui recherche le C-MSER à différentes échelles dans une pyramide scale-space est utilisée, où chaque niveau successif est créé redimensionnant l'image par un facteur de deux. À chaque niveau de la pyramide d'échelle, on définit la taille de la plus petite région acceptable afin de supprimer le bruit parasite.

4.2.2 Representational Transformation

Le C-MSER à plusieurs échelles (multiscale C-MSER) fournit une liste de régions candidates susceptibles de contenir certaines lésions. Pour chaque région candidate, un certain nombre de caractéristiques sont calculées pour obtenir une approximation statistique de la forme du candidat. Ces caractéristiques servent aussi pour éliminer les lésions qui sont détectées à des échelles différentes. Une fois les doublons éliminés les candidats restants sont transformés en des carrés de largeur w . Les portions obtenues à l'issue de l'étape précédente subissent une modification d'intensité suivant la formule suivante : $I' = 127 + \frac{10(I-\mu)}{\sigma}$. Où μ représente l'espérance et σ l'écart-type. La figure suivante montre les résultats obtenus à l'issue de cette étape.

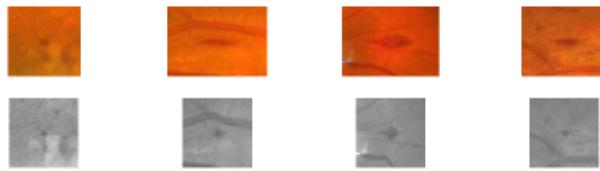


FIGURE 4.5 – Exemple de lésions détectées.

4.2.3 Le réseau de neurones convolutionnels «CNN»

Une simple architecture de réseau de neurones comportant des couches de convolutions et max-pooling suivies par des couches entièrement connectées. Le modèle est entraîné avec la technique de rétro propagation du gradient en utilisant l'algorithme de gradient descent pour mettre à jour les poids.

Le modèle a été entraîné et testé sur chacune des bases d'apprentissage suivantes :

DIARETDB1 : Elle contient 89 images, les images de rang pair sont utilisées pour entraîner le modèle et les autres pour le tester.

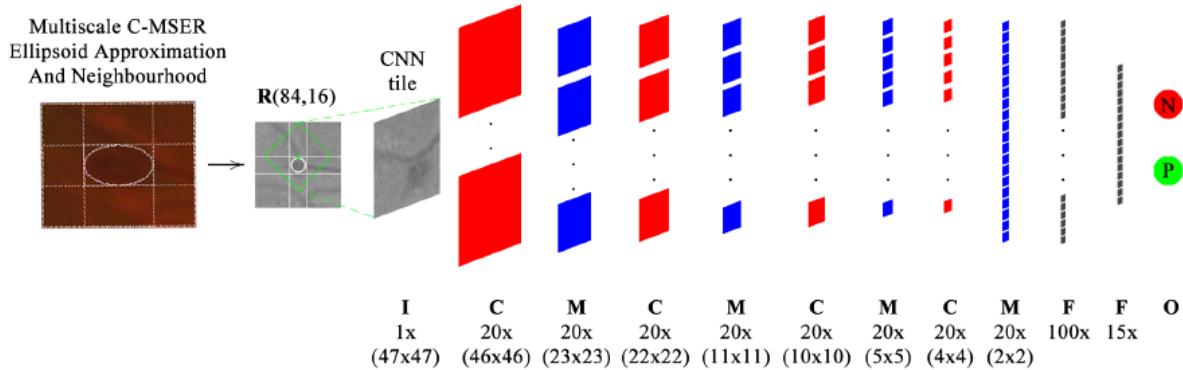
SiDRP : Elle est composée de 2029 images de grande taille dont 1079 images ont été utilisées pour l'apprentissage et 950 images pour le test.

4.2.4 Résultats

Le modèle a été évalué par sa capacité de classifier les lésions qui caractérisent la RD et par son efficacité de détecter la maladie elle-même.

1. Classification des lésions

En premier lieu l'efficacité du modèle présenté ci-dessus (ayant comme entrée les portions des images du fond d'œil) à classer les lésions sur la base des représentations transformées (CNN-TR) des régions candidates produites par C-MSER multi-échelle est comparée avec un réseau de neurones convolutionnel "CNN" formé directement les images brutes (CNN-NT), des machines à vecteurs de support (SVM) et forêts d'arbres décisionnels (RF). Les entrées du CNN-TR sont des portions d'image de taille 47×47 pixels (voir 4.6).



I = Input layer, C = Convolutional layer, M = Max-pooling layer, F = Fully-connected layer, O = Output layer

FIGURE 4.6 – L’architecture du CNN.

Durant l’apprentissage, les portions sont pivotées aléatoirement de manière à refléter l’invariance rotationnelle des lésions. En outre plus de vingt caractéristiques des régions candidates C-MSER (largeur, longueur, périmètre, stabilité … etc.) sont utilisés comme entrées de l’algorithme SVM et de l’algorithme des forêts d’arbres décisionnels.

Les résultats de classification des lésions des deux bases d’apprentissage sont illustrés dans les figures 4.7 et 4.8.

Les tests effectués sur la base SiDRP montrent que : le modèle proposé «CNN-TR» réalise les mêmes résultats qu’un CNN classique «CNN-NT», Cependant sa capacité à détecter les hémorragies et les exsudats est nettement meilleure par rapport aux autres.

Les tests effectués sur la base DIARETDB1 montrent que : les résultats réalisés par l’ensemble des classificateurs sont similaires. Cela peut être dû à la base elle-même ; des situations où des lésions non marquées peuvent se révéler être de vraies lésions et des lésions marquées peuvent, en fait, être de fausses lésions.

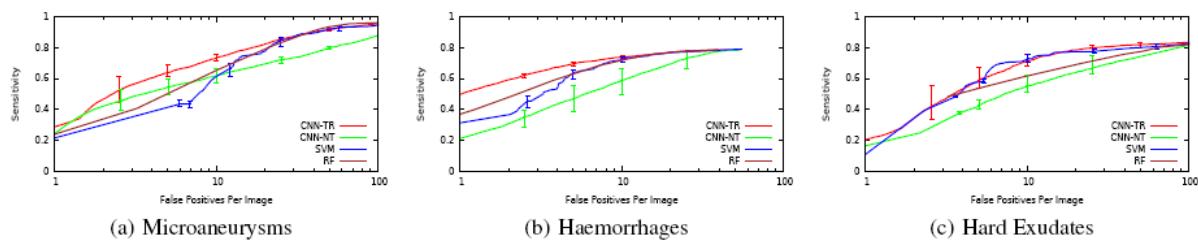


FIGURE 4.7 – Résultats de classification des lésions sur SiDRP.

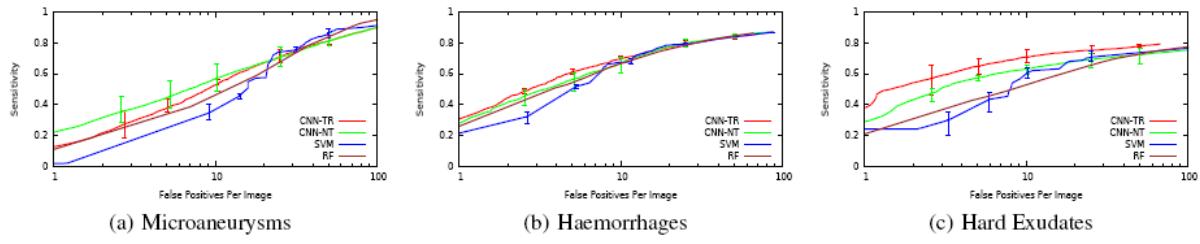


FIGURE 4.8 – Résultats de classification des lésions sur DIARETDB1.

2. Classification des images (RD/No RD)

Le modèle proposé a été testé sur la base d'apprentissage SiDRP. En faisant varier le seuil «**Threshold**» (qu'au-dessus lequel une image est considérée comme atteinte par RD et saine sinon) le CNN-TR a réussi à obtenir une sensibilité de 100 % avec une spécificité de 30 % et pour une sensibilité de 90 % on obtient une spécificité de 68 % plus que tous les classificateurs. Comme le montre la 4.9.

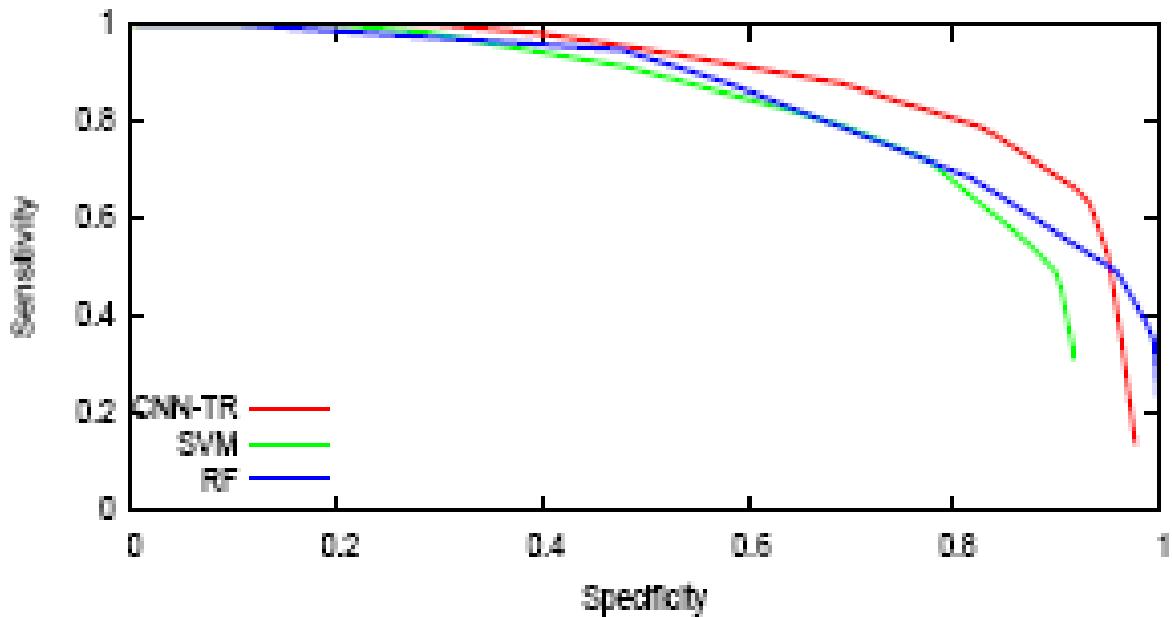


FIGURE 4.9 – Résultats de la classification des images sur SiDRP.

La méthode présentée ci-dessus consiste à extraire des portions ou régions représentatives à partir des images du fond d'œil en utilisant une technique de détection des régions. Ces portions sont par la suite injectées dans le modèle CNN décrit ci-dessus. Une telle approche permet d'éliminer les régions non informatives qui rendent le temps d'apprentissage inutilement long. Cependant l'efficacité des résultats de cette méthode dépend de l'efficacité de la technique de détection des régions représentatives qui elle-même dépend de la qualité d'image en entrée.

4.3 Méthode 3 : CNN Based Transfer Learning for Diabetic Retinopathy Fundus Image Classification. Xiaogang Li et al [28]

L'un des plus grands défis du développement des systèmes de détection automatique dans le domaine d'imagerie médicale est la difficulté de trouver une base d'apprentissage avec un grand nombre d'image annotée rendant ainsi l'application des réseaux de neurones convolutionnels «CNNs» dans ce domaine un réel défi.

L'objectif de cette méthode est de proposer une nouvelle approche pour remédier à ce problème. Il a été constaté que le recours à l'apprentissage par transfert «Transfer Learning» semble apporter des résultats prometteurs.

Xiaogang Li, et al proposent deux façons différentes pour aborder le problème tous basés sur des modèles de réseau de neurones convolutionnels «CNNs». Une première solution consiste à «fine-tune» toutes les couches du réseau de neurones de chaque modèle entraîné préalablement sur une autre base d'apprentissage «AlexNet, GoogLeNet, Vgg, ... etc.» ; l'idée c'est d'initialiser les poids (Weights) du modèle avec celles d'un modèle pré-entraîné sur un autre ensemble de données. Une deuxième solution consiste à «fine-tune» un certain nombre de couches du réseau ; les poids des couches restantes sont fixés lors de l'initialisation.

L'objectif de ce travail est de montrer l'utilité d'apprentissage par transfert appliquée aux réseaux de neurones convolutionnels pour la détection de la RD. En effet les résultats obtenus ont révélé que l'apprentissage par transfert peut améliorer ces derniers d'une manière significative même si le modèle de classification a été entraîné sur des petits ensembles de données (domaine cible), en tirant profit des connaissances apprises sur un autre ensemble de données (domaine source). L'apprentissage par transfert est une technique prometteuse qui favorise l'utilisation des CNNs profonds dans le domaine médical qui est souvent caractérisé par un nombre limité de données.

4.3.1 Apprentissage par transfert

L'apprentissage par transfert est une technique visant à améliorer un apprentissage A en transférant les connaissances apprises lors de l'apprentissage B. La figure suivante montre clairement la différence entre un processus d'apprentissage standard a et un processus d'apprentissage par transfert b. Contrairement au processus d'apprentissage standard qui part de rien l'apprentissage par transfert transmet des connaissances d'un domaine source vers un domaine cible, dans le cas présent l'ensemble des données ImageNet représente le domaine source et l'ensemble des images du fond d'œil représente le domaine cible.

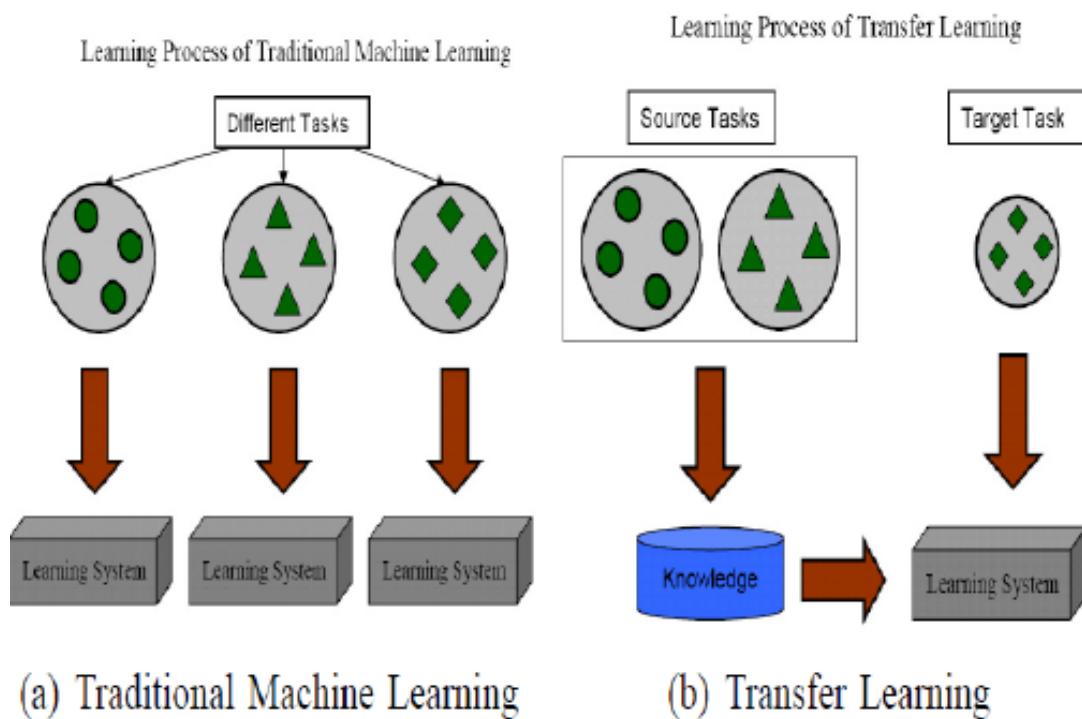


FIGURE 4.10 – La différence entre l'apprentissage traditionnel et l'apprentissage par transfert

4.3.2 Les modèles CNNs pré-entraînés

Plusieurs modèles CNNs pré-entraînés ont été utilisé dans cette recherche parmi ces modèles on peut : AlexNet, Vgg-s, VggNet-vd-16 et VggNet-vd-19. Tous les modèles précédents ont été pré-entraînés sur la même base d'apprentissage ImageNet².

4.3.3 Fine-tunning des modèles pré-entraînés

CNNs fine-tunning consiste à entraîner un réseau de neurones convolutionnels CNN à partir d'un ensemble de poids pré-entraînés. Autrement dit il consiste à initialiser les poids d'un modèle A avec les poids d'un autre modèle B qui a été pré-entraîné sur une autre base d'apprentissage, ensuite le modèle A est entraîné sur la base d'apprentissage cible en l'occurrence les images du fond d'œil, ces deux modèles partagent la même architecture.

Deux techniques de fine-tunning ont été utilisées par Xiaogang Li, et al. La première consiste à fine-tune toutes les couches du modèle CNN pré-entraîné, cette technique est très couteuse en temps d'exécution et est recommandée lorsque les domaines source et cible sont faiblement corrélés. La deuxième technique consiste à fine-tune quelques couches du modèle CNN pré-entraîné ; Le principe de cette technique est le suivant : Nous commençons par entraîner (mettre à jour) la dernière couche entièrement connectée du modèle CNN pré-entraîné et on inclut progressivement les autres couches dans le processus d'apprentissage jusqu'à atteindre le critère d'arrêt, La raison qui justifie la politique d'inclusion des couches dans le processus d'apprentissage (de plus profond au moins profond) est que : **les couches inférieures (moins profondes) peuvent apprendre des caractéristiques d'images de bas niveau communes à toutes les images,tandis que les couches supérieures(plus profondes) peuvent apprendre des représentations de haut niveau applicables à la tâche cible spécifique comme le montre la figure suivante.**

4.3.4 Implémentation et Résultats

Tous les modèles CNNs pré-entraînés décrits précédemment sont utilisés pour opérer une classification **binaire** des images du fond d'œil. Pour cet effet deux bases d'apprentissage DR1 et MESSIDOR contenant 2214 images du fond d'œil annotées sont utilisées.

Les images des deux bases d'apprentissage sont prétraitées avant de lancer le processus d'apprentissage. Le prétraitement de ces images consiste en :

- Un rognage pour éliminer les zones noires autour du fond d'œil.

2. ImageNet est une base de données d'images qui sont organisées hiérarchiquement, chaque nœud de cette hiérarchie est représenté par des centaines de milliers d'images.

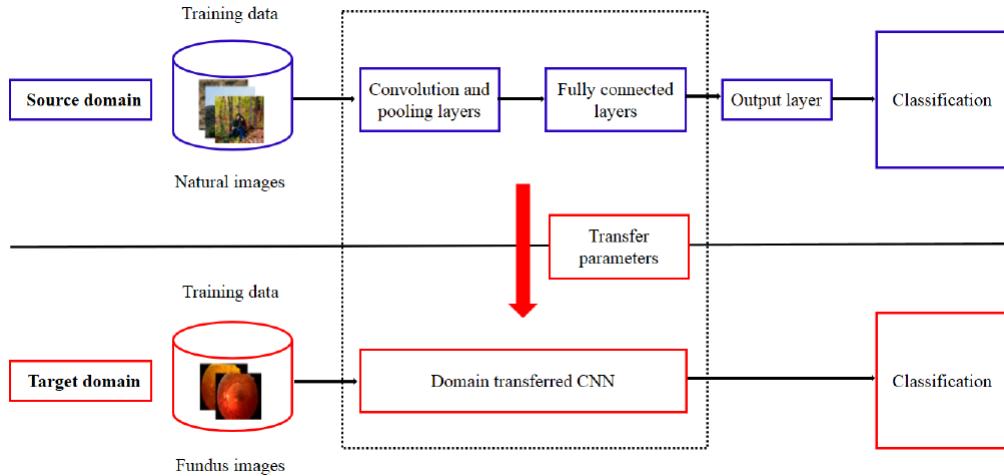


FIGURE 4.11 – Le pipeline de transfert des paramètres.

- Redimensionnement des images pour correspondre à la taille des images acceptées par les CNNs pré-entraînés ($224 \times 224 \times 3$ RGB).

Pour mener l’expérience 4 modèles CNNs pré-entraînés à savoir : AlexNet, Vgg-s, VggNet-vd-16 et VggNet-vd-19 ont servi pour initialiser les 4 modèles CNNs utilisés pour la classification des images du fond d’œil, ces derniers partagent les mêmes architectures avec les modèles pré-entraînés à l’exception de la dernière couche qui est constituée de deux neurones seulement, les couches (ou certaines tout dépend de la politique de fine-tunning) de chaque modèle sont initialisées en utilisant les poids du modèle pré-entraîné correspondant. Comme le fine-tunning nécessite un ensemble d’images de taille moyenne, la base d’apprentissage courante est enrichie en utilisant une technique de data augmentation qui consiste à appliquer quelques transformations sur les images de la base d’apprentissage par la suite ces modèles sont entraînés sur cette dernière jusqu’à atteindre le critère d’arrêt à savoir 30 epochs.

Dans un premier temps c’est le full fine-tunning qui est effectué pour chacun des 4 modèles choisis. Pour évaluer l’efficacité des résultats la technique de validation croisée «5 fold cross-validation [29]» est utilisée. De plus 4 autres métriques d’évaluation sont calculées ; la sensibilité SN, la spécificité SP, la précision ACC et l’aire sous la courbe ROC. Les résultats des tests sont renseignés dans le tableau suivant où on peut constater distinctement que Vgg-s a surpassé les autres modèles.

| Model | DR1 | | | | MESSIDOR | | | |
|--------------|--------------|--------------|--------------|---------------|--------------|--------------|--------------|---------------|
| | SP | SN | ACC | AUC | SP | SN | ACC | AUC |
| AlexNet | 94.07 | 81.27 | 89.75 | 0.9342 | 72.44 | 72.63 | 72.56 | 0.7727 |
| Vgg-s | 97.43 | 86.47 | 93.73 | 0.9786 | 97.11 | 86.03 | 92.01 | 0.9834 |
| VggNet-vd-16 | 94.32 | 90.78 | 93.17 | 0.9616 | 70.86 | 65.78 | 68.67 | 0.7437 |
| VggNet-vd-19 | 96.49 | 89.31 | 94.12 | 0.9684 | 60.30 | 64.70 | 62.19 | 0.6869 |

FIGURE 4.12 – Résultats de la classification basée sur full fine-tunning.

Dans un deuxième temps c'est le fine-tunning partiel qui est appliqué uniquement pour le modèle réalisant les meilleurs résultats dans les tests précédents à savoir Vgg-s. Comme le montre le tableau 1 le modèle Vgg-s comporte 8 couches dont 5 couches de convolution et 3 couches entièrement connectées. Dans l'expérience suivante un modèle Vgg-s est initialisé avec les poids d'un autre Vgg-s ayant la même architecture. Le processus de fine-tunning est appliquée à plusieurs niveau ; dans un premier temps les poids des toutes les couches sont mis à jour lors de l'apprentissage (ligne 1 du tableau 3), dans un deuxième temps les poids de la couche Conv1 sont fixés et tous les autres poids sont mis à jour lors de l'apprentissage (ligne 2 tableau 3), le processus est répété jusqu'à ce que seulement les poids de la dernière couche soient mis à jour et les autres fixés. Comme le montre le tableau 3 où U signifie que les poids de la couche en question sont mis à jour lors de la phase d'apprentissage les poids de la couche en question sont fixés. On constate que les résultats des tests (Tableau4) sont très différents pour les deux bases d'apprentissage ; On remarque que les tests effectués sur la base DR1 donnent des résultats similaires, tandis que pour la base MESSIDOR les meilleurs résultats sont obtenus quand le full fine-tunning est appliqué et la performance du modèle commence à se dégrader au fur et à mesure que les couches du modèle sont exclues du processus d'apprentissage. Une raison probable pouvant expliquer cela est que l'ensemble de donnée source et la base MESSIDOR ne sont pas corrélés contrairement à la base DR1 qui est fortement corrélé avec la source.

| Fine-tuning layers | Domain transferred CNN | | | | | | | |
|--------------------|------------------------|-------|-------|-------|-------|-----|-----|-----|
| | Conv1 | Conv2 | Conv3 | Conv4 | Conv5 | Fc6 | Fc7 | Fc8 |
| Conv1 - Fc8 | U | U | U | U | U | U | U | U |
| Conv2 - Fc8 | F | U | U | U | U | U | U | U |
| Conv3 - Fc8 | F | F | U | U | U | U | U | U |
| Conv4 - Fc8 | F | F | F | U | U | U | U | U |
| Conv5 - Fc8 | F | F | F | F | U | U | U | U |
| Fc6 - Fc8 | F | F | F | F | F | U | U | U |
| Fc7 - Fc8 | F | F | F | F | F | F | U | U |
| Fc8 | F | F | F | F | F | F | F | U |

FIGURE 4.13 – Description du schéma du fine-tunning partiel.

| Fine-tuning layers | DR1 | | | | MESSIDOR | | | |
|--------------------|--------------|--------------|--------------|---------------|--------------|--------------|--------------|---------------|
| | SP | SN | ACC | AUC | SP | SN | ACC | AUC |
| Conv1 - Fc8 | 97.43 | 86.47 | 93.73 | 0.9786 | 97.11 | 86.03 | 92.01 | 0.9834 |
| Conv2 - Fc8 | 97.63 | 88.43 | 94.52 | 0.9655 | 95.23 | 74.11 | 87.46 | 0.9467 |
| Conv3 - Fc8 | 96.94 | 87.45 | 93.79 | 0.9722 | 76.15 | 76.19 | 76.19 | 0.8210 |
| Conv4 - Fc8 | 96.49 | 88.14 | 93.66 | 0.9702 | 79.70 | 69.59 | 75.31 | 0.8134 |
| Conv5 - Fc8 | 96.79 | 87.16 | 93.60 | 0.9686 | 86.02 | 65.46 | 77.06 | 0.8268 |
| Fc6 - Fc8 | 97.04 | 85.69 | 93.27 | 0.9628 | 81.38 | 64.44 | 74.00 | 0.8038 |
| Fc7 - Fc8 | 97.33 | 85.78 | 93.50 | 0.9659 | 74.91 | 72.32 | 73.75 | 0.8094 |
| Fc8 | 97.53 | 82.16 | 92.35 | 0.9610 | 81.23 | 48.25 | 66.78 | 0.7576 |

FIGURE 4.14 – Résultats de la classification basée sur le fine-tunning partiel.

Les résultats précédents montrent que le fine-tunning partiel est très bien adapté aux situations où les domaines source et cible sont corrélés et l'ensemble de donnée disponible est très limité. Ceci permettra au modèle de réaliser de très bons résultats en ayant été entraîné sur nombre limité de données ce qui lui évitera de tomber dans le problème de surapprentissage.

L'un des problèmes qui peuvent survenir lors de l'utilisation de la technique d'apprentissage par transfert est le transfert négatif. En effet le modèle pré-entraîné risque de donner des résultats plus médiocres que s'il n'était pas pré-entraîné.

De plus un tel apprentissage nécessite une base d'apprentissage équilibrée pour éviter qu'un surapprentissage survienne.

Tableau Récapitulatif

Nous avons vu précédemment deux approches complètement différentes pour la détection automatique de la RD. La première approche est basée sur la détection et l'extraction des lésions qui caractérisent la maladie. Tandis que la deuxième approche repose sur les réseaux de neurones convolutionnels pour la détection de la RD. Le tableau suivant est un tableau récapitulatif des méthodes de détection présentées ci-dessus.

TABLE 5.1 – Tableau récapitulatif

| Méthode | Concept de Base | Principe | Taille base de test |
|---------|-------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|--------------------------|
| [30] | Détection des anomalies vasculaires | Filtres Adaptés, Détection des countours, classification selon le diamètre du vaisseau. | petite |
| [18] | Détection des AVs | Suivi des vaisseaux, filtre de Kalman et classification selon la structure du vaisseau. | très petite |
| [19] | Détection des AVs | Filtre de Gabor, classification selon l'orientation des vaisseaux ; une seule orientation dominante => sain, plusieurs => RD. | petite |
| 25 | Détection des microanévrismes | Algorithme croissance de région, Filtres adaptés, opérations morphologiques sur des angiographie fluoroscénique. | petite |
| [20] | Détection des MAs | Elimination des vaisseaux sanguins, Amélioration locale du contraste, croissance de région et K-NN. | NA |
| [22] | Détection des exsudats | Opérations morphologiques, Elimination de la papille, Algorithme d'Otsu. | 40 images |
| [26] | CNN (classification binaire) | Attribution de poids à chacune des deux classes pour prévenir le surapprentissage | 5000 dont 4000 saines |
| [27] | CNN (classification binaire) | Extraction des portions représentatives avec l'algo C-MSER, classification avec CNN | 950 images |
| [28] | CNN et apprentissage par transfert | Utilisation de l'apprentissage par transfert | <300 images |

DEUXIÈME PARTIE

CONCEPTION ET RÉALISATION

Conception du modèle

6.1 Introduction

L'évolution remarquable des performances des machines ces dernières années incité la réapparition des réseaux de neurones dans plusieurs domaines notamment dans le secteur médical. En effet on distingue une utilisation à large échelle des réseaux de neurones convolutionnels dans la détection des maladies, entre autres : La détection des cellules cancéreuses(ex :détection du cancer de sein), détection des symptômes d'Alzheimer sur des images IRM ... etc.

Bien que la RD est une maladie discrète, elle présente des symptômes révélateurs du stade actuel de la maladie.Ces symptômes qui sont présents sur l'image du fond d'oeil se ressemblent tous et est donc difficile à un réseau de neurones convolutionnel simple de les différencier l'un de l'autre.D'autre part la conception des modèles de diagnostic qui concerne le secteur médical doivent être suffisamment fiable pour y fier du moment que ça concerne la santé publique et la moindre erreur peut avoir des répercussions majeures sur la santé du patient.

La conception de tels modèles doit passer par plusieurs étapes pour s'assurer que le modèle final satisfait tous les critères pour pouvoir être déployé dans les établissements de santé. En effet l'efficacité et la fiabilité de ce dernier dépend de plusieurs paramètres et nécessite de faire des choix minutieux au cours des phases de conception et que nous allons détailler dans les paragraphes qui suivent.

6.2 La Base d'apprentissage

6.2.1 Introduction

L'une des raisons qui a fait que l'utilisation des réseaux de neurones dans le domaine médical avançait à petits pas est la rareté des bases d'apprentissage relatives à ce dernier. Ces dix dernières années le domaine de l'imagerie médicale s'est beaucoup développé, on trouve de plus en plus des bases d'apprentissage des images de scanner, IRM, du fond d'œil, ... etc.

La base d'apprentissage constitue le rouage du processus d'apprentissage, il est donc impératif de bien choisir sa base d'apprentissage, l'équilibrer et la traiter pour aboutir aux résultats escomptés.

Dans le présent travail nous avons utilisé une base d'apprentissage publié par la plateforme Kaggle¹ dans le cadre d'une compétition de détection de la RD. C'est une base d'images du fond d'œil annotées et classées en 5 catégories.

6.2.2 Description

Les images du fond d'œil sont fournies par EyePACS², c'est une plateforme pour le dépistage de la rétinopathie diabétique. EyePACS a mis en place un système pour la récolte des images du fond d'œil des patients diabétiques à travers le monde. Ces images ont été capturées sous différentes conditions de lumière, par différents rétinographes. Ces dernières ont été annotées et classifiées par des spécialistes en 5 catégories : œil sain, RD non proliférante minime, RD non proliférante modérée, RD non proliférante sévère ,RD proliférante.

6.2.3 Étude et statistiques

La base d'apprentissage contient 35126 images catégorisées en 5 classes. La distribution des images du fond d'œil dans la base d'apprentissage est consignée dans le tableau 6.1 suivant.

1. Kaggle est une plateforme web organisant des compétitions en science des données. Sur cette plateforme, les entreprises proposent des problèmes en science des données et offrent un prix aux datalogistes obtenant les meilleures résultats.

2. <https://www.eyepacs.com/>

| Stade | Nombre d'images | Fréquence relative |
|-------|-----------------|--------------------|
| 0 | 25810 | 73.47 % |
| 1 | 2443 | 6.95 % |
| 2 | 5292 | 15.07 % |
| 3 | 873 | 2.49 % |
| 4 | 708 | 2.02 % |

TABLE 6.1 – Le nombre et la fréquence des images dans chaque classe

6.2.4 Problème de déséquilibre de la base d'apprentissage

Le problème de base d'apprentissage déséquilibrée est propre aux problèmes de classification, il survient lorsque les nombres de données des classes sont largement espacés ; par exemple si une classe prédomine toutes les autres classes.

Ce problème est très courant dans les problèmes de classification des maladies où le nombre des images saines est largement supérieur aux nombres des images présentant une maladie. Ceci a un impact direct sur la qualité de l'apprentissage et donc l'efficacité du modèle. En effet un modèle entraîné sur une base d'apprentissage très déséquilibrée est biaisé et inefficace car l'objectif de tout modèle est d'améliorer la précision en minimisant la fonction objective sans tenir compte de la proportion des images dans la base d'apprentissage. Exemple : Supposant un problème de classification binaire où la classe 0 contient 90% des données, le modèle s'aperçoit donc que la meilleure chose à faire c'est de classer toutes les images dans la classe 0 ce qui donne donc une précision de 90%.

Il existe plusieurs techniques pour remédier à ce problème, ci-dessous une liste non exhaustive des techniques employées :

- **sous-échantillonnage (Undersampling)** : consiste à supprimer aléatoirement des images des classes les plus importantes.
 - **sur-échantillonnage (Oversampling)** : consiste à rajouter de nouvelles images aux classes les moins nombreuses pour être à pied égal avec les classes nombreuses.
- Pour ce faire il existe plusieurs techniques :
- **Technique de suréchantillonnage des minorités synthétiques.**³ : Cette technique consiste à générer de nouvelles images en rajoutant des bruits sur les images existantes.[31]
 - **GAN Réseaux antagonistes génératifs (Generative Adversarial Network)** : Elle consiste à générer des images complètement nouvelles. Toutefois cette technique est très gourmande en temps d'exécution[32]

3. Synthetic Minority Over-sampling Technique (SMOTE)

6.2.5 Choix de technique d'équilibrage

On peut constater clairement à partir du tableau précédent que la base d'apprentissage utilisée est fortement déséquilibrée ; on voit que la classe saine domine toutes les autres classes. En fait ce déséquilibre favorise le surapprentissage du modèle sur la première classe. Par conséquent le modèle devient complètement stérile et biaisé. Pour remédier à cela nous avons opté pour une technique d'undersampling en temps réel. À chaque itération du processus d'apprentissage on prend m images tirées aléatoirement de chaque classe, où m désigne le nombre d'images de la moins peuplée. Cette technique nous évite de perdre des images qui peuvent être utiles dans l'apprentissage comme c'est le cas de la technique undersampling tout en évitant le surapprentissage.

De plus pour s'assurer de la qualité du modèle, on utilise une base de test complètement équilibré lors de l'évaluation du modèle pour éviter l'accuracy paradox. En outre plusieurs d'autres métriques sont utilisées pour évaluer le modèle : matrice de confusion, sensibilité et spécificité.

6.3 Processus général de la classification de la RD

L'ultime objectif de ce travail étant de classifier l'image du fond d'œil d'un patient en déterminant le stade actuel de la maladie. Il est donc primordial de définir un processus général de classification de l'image du fond d'œil d'un patient.

Le schéma dans la figure 6.1 suivant illustre le processus général de la classification d'une image du fond d'œil.

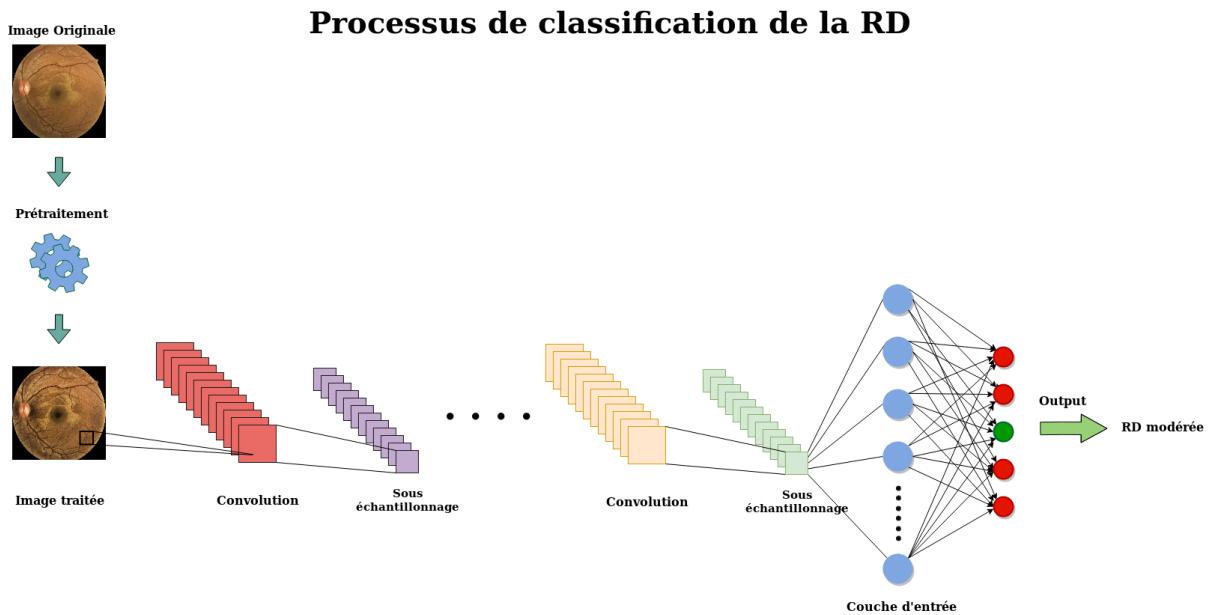


FIGURE 6.1 – Processus de classification de la RD

6.3.1 Pré-traitement

La phase de prétraitement est d'une grande utilité et permet d'améliorer significativement l'efficacité de la classification. Il est donc primordial de prétraiter minutieusement les images de notre base d'apprentissage de façon à éliminer les informations inutiles dans les images et pouvoir mettre l'accent sur les régions qui contiennent des informations importantes pour la classification telles que les lésions. Elle passe par 4 étapes essentielles :

1. Amélioration du contraste

Les images de notre base d'apprentissage ont été prises sous différentes conditions de lumière, avec des rétinographes différents et dans des sociétés complètement différentes. Par conséquent on trouve un grand nombre d'images du fond d'œil peu éclairées ou trop éclairées, ceci détériore significativement la qualité de l'image et donc la classification.

Le but de cette étape est d'améliorer le contraste des images, ce qui permet de mettre en lumière les lésions qui caractérisent les stades de la maladie.

Nous utilisons la technique de Contrast Limited Adaptive Histogram Equalization (CLAHE)[33]. C'est une méthode d'ajustement local du contraste en utilisant les histogrammes des régions différentes de l'image. Ceci permet en effet d'améliorer considérablement le contraste de l'image et mettre en évidence des détails même dans les régions les plus lumineuses ou sombres.

Les deux images suivantes montrent le résultat avant et après l'amélioration du contraste.

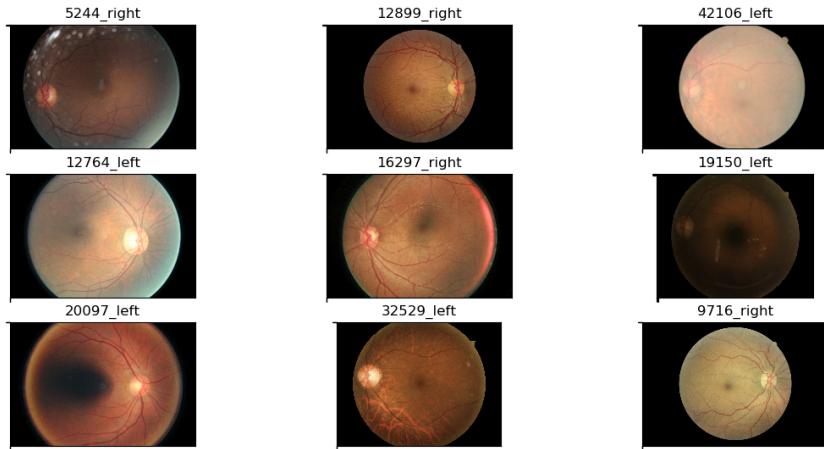


FIGURE 6.2 – Images extraites de la base d'apprentissage

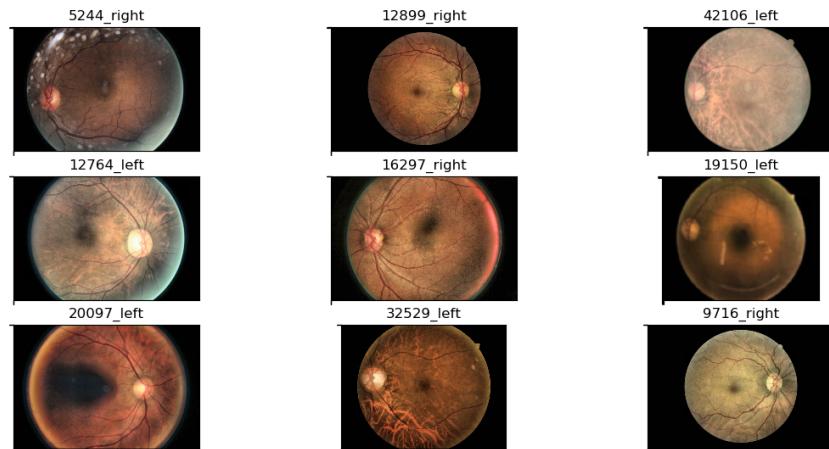


FIGURE 6.3 – Le résultat après l'amélioration du contraste

2. Élimination de bruit

En effet la présence des bruits dans les images du fond d'œil peut être nuisible lors de l'apprentissage ; le modèle peut les considérer comme des lésions (ex : microanévrismes), ce qui peut fausser la classification. A cet effet on applique une technique de BLUR avec un filtre médian pour flouter les images et éliminer tous les bruits présents dans l'image.

La figure 6.4 suivante montre le résultat après le floutage.

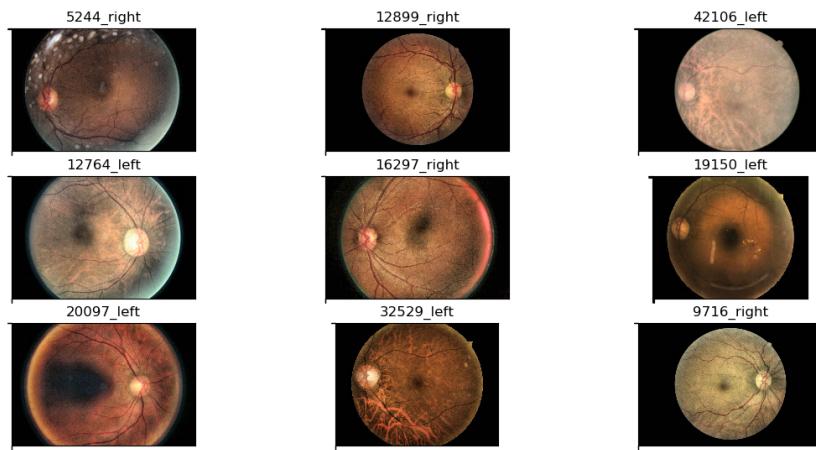


FIGURE 6.4 – Le résultat après le floutage

3. Isoler le fond d'œil du fond noir

Comme on peut le constater dans les images précédentes, les images de fond d'œil sont entourées par une zone noire. La présence de ce fond noir est inutile dans la classification car il ne contient aucune information. De ce fait, il doit être supprimé.

Pour ce faire on utilise une technique qui détermine approximativement le rayon du fond d'œil et élimine toute la zone noire qui l'entoure.

La figure 6.5 suivante montre le résultat après la suppression du fond noir.

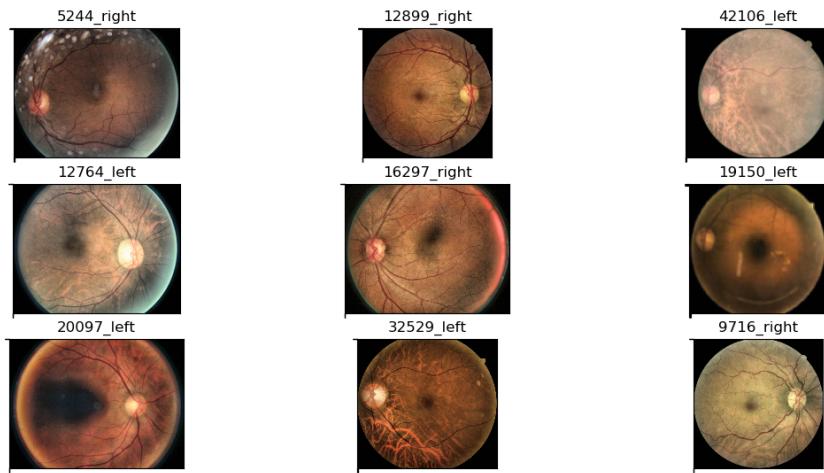


FIGURE 6.5 – Le résultat après la suppression du fond noir.

4. Redimensionnement de l'image

Cette étape consiste à redimensionner l'image en réduisant sa taille tout en minimisant la perte d'informations. L'objectif est de réduire le temps d'apprentissage et de fournir au modèle des images de même taille.

En effet toutes les images du fond d'œil sont réduites à la taille $224 \times 224 \times 3$, ces dimensions correspondent aux dimensions des images attendus par le modèle présenté dans le paragraphe suivant.

La figure 6.6 suivante illustre le processus de prétraitement et ses étapes.

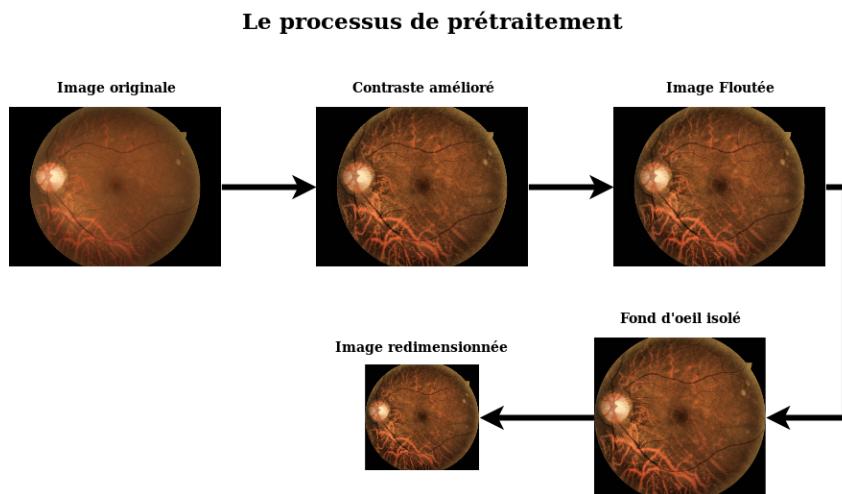


FIGURE 6.6 – illustration du processus de prétraitement.

6.3.2 L'architecture du réseau de neurones convolutionnel proposé

6.3.2.1 Introduction

Le réseau de neurones convolutionnel constitue le cœur du processus de détection de la RD. Il doit être suffisamment robuste pour pouvoir différencier les différents i.e. reconnaître les lésions présentes dans l'image du fond d'œil sans qu'il soit explicitement programmé à le faire. Cette faculté de pouvoir distinguer entre les différents stades de la RD est acquise lors de la phase d'apprentissage.

Le choix de l'architecture du modèle s'avère donc une étape difficile et dépend de plusieurs paramètres pour aboutir aux résultats attendus.

Notre choix d'architecture s'est porté sur le ResNet50 pour plusieurs raisons que nous allons détailler un peu plus bas.

6.3.2.2 ResNet50

Le ResNet50 ou le ‘Residual Network’ est un réseau de neurones convolutionnel proposé par Microsoft en 2015, dans la même année il a décroché la première place dans la compétition d’ImageNet en surpassant AlexNet et Inception [34].

Après le succès qu'il a connu, ResNet est devenu un modèle très populaire et largement utilisé à cause des avantages qu'il offre.

Le principe de l'architecture de ResNet repose sur les blocs résiduels qui est un nouveau concept introduit par cette dernière. Ils sont caractérisés par des liaisons entre des couches non successives appellées "skip connections". Elles permettent au modèle de remédier à un problème très connu dans les réseaux de neurones profonds : l'évanescence du gradient.

L'évanescence du gradient

C'est un problème très classique et courant dans les réseaux de neurones profonds. Il survient lors de l'apprentissage de ces derniers ; en effet la mise à jour des poids synaptiques nécessite le calcul des dérivées partielles relatives à ces derniers. Lors de la rétropropagation du gradient, ce dernier est démultiplié autant de fois qu'il y a de couche dans le réseau de neurones. Le gradient se trouve donc réduit à chaque multiplication et aura une infime valeur lorsqu'il atteint la première couche. Ainsi, les poids de cette dernière ne sont plus mis à jour ce qui influence l'apprentissage. Par conséquent plus le réseau est profond plus ce problème aura d'impact sur l'apprentissage. Il est important de souligner le fait que

plusieurs facteurs peuvent provoquer ce problème entre autres : la fonction d'activation telle que sigmoid qui retourne toujours une valeur entre 0 et 1 quel que soit la valeur en entrée les valeurs des dérivées partielles se trouvent donc atténuées.

Le ResNet50 pallie à ce problème d'évanescence du gradient avec l'introduction des blocs résiduels.

Les blocs Résiduels

C'est un nouveau concept introduit par le ResNet qui est caractérisé par des liaisons résiduelles ‘Residual connections’ comme illustré dans la figure suivante. En effet la liaison résiduelle permet de véhiculer la valeur à l'entrée du bloc vers la sortie de ce dernier et la rajouter avec sa sortie ($F(x) + x$). Aucune fonction d'activation n'est appliquée à cette liaison ce qui évite que les gradients s'atténuent à cause de la fonction d'activation.

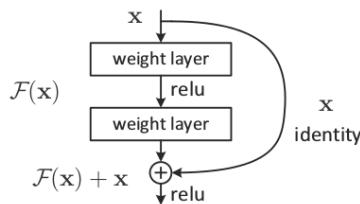


FIGURE 6.7 – Un bloc résiduel.

Les deux principales raisons qui font que les blocs résiduels sont très efficaces pour remédier au problème d'évanescence du gradient sont :

- Ils permettent aux dérivées partielles de remonter jusqu'au couches inférieures sans que leurs valeurs soient atténuées, ceci est possible grâce aux skip connections.
- L'ajout de nouvelles couches au modèle ne peut pas détériorer les résultats de ce dernier car ils ne peuvent être que meilleurs et dans le pire cas identiques aux résultats obtenus avec le modèle initial, ceci est possible grâce aux skip connections qui peuvent détourner n'importe quelle couche si cette dernière détériore les résultats.

Comme son nom l'indique le ResNet50 est constitué de 50 couches de convolution et sous-échantillonnage séparées par des fonctions d'activation et de batch normalization. Le ResNet50 est constitué de successions de deux types de blocs :

1. bloc identité (Identity block)

C'est le plus simple bloc dans l'architecture ResNet50, il est constitué de 3 couches de convolution séparées par des couches de batch normalization et des fonctions d'activation ainsi qu'une skip connection. Ces blocs préservent la taille de l'image en entrée du bloc, la valeur de la skip connection est donc directement rajouté à la sortie du bloc sans aucun traitement nécessaire. Le schéma 6.8 suivant illustre la composition d'un bloc identité.

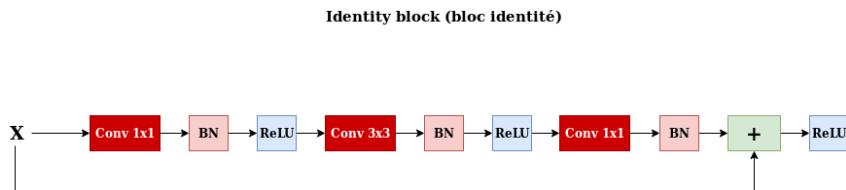


FIGURE 6.8 – bloc identité.

2. bloc de convolution (Conv block)

C'est le deuxième type de bloc qu'on trouve dans l'architecture ResNet50, il se place entre deux stages différents, son utilité est de réduire la taille de l'image en entrée. Il est constitué lui aussi de 3 couches de convolution séparées par des couches de batch normalization et des fonctions d'activation ainsi qu'un skip connection qui relie l'entrée directement à la sortie du bloc. Toutefois comme la taille de l'image change à l'intérieur du bloc la valeur véhiculée par la skip connection doit être de même taille que la valeur en sortie, pour cela on place une couche de convolution supplémentaire dans la skip connection pour pouvoir sommer les deux valeurs à la sortie du bloc. Le schéma 6.9 suivant illustre la composition d'un bloc de convolution.

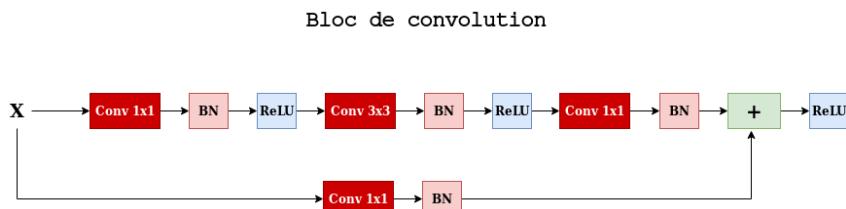


FIGURE 6.9 – bloc de convolution.

A présent que les deux briques essentielles du ResNet50 sont connues, on peut dévoiler l'intégralité de l'architecture.

Pour des soucis de compréhension et de visualisation nous avons jugé qu'il est préférable de diviser l'intégralité de l'architecture en 3 parties.

1. Partie 1

Elle est constituée d'une couche d'entrée qui représente l'image initiale. Cette dernière est bournée avec des zéros de chaque côté, la taille de l'image devient donc $230 \times 230 \times 3$. L'objectif de cette étape est d'obtenir une image de taille 112×112 après la couche de convolution suivante. Le résultat obtenu est passé dans une couche de convolution avec 64 filtres de taille 7×7 qui permettent d'extraire les caractéristiques les plus importantes de l'image. Le résultat obtenu est une image de taille $112 \times 112 \times 64$ qui est passée dans une couche de batch normalization suivie

par une fonction d’activation de type ReLU suivie par un sous échantillonnage de type max pooling. Le schéma 6.10 suivant illustre les composants de cette partie.

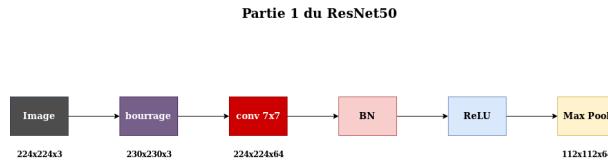


FIGURE 6.10 – Partie 1 du ResNet50.

2. Partie 2

Cette partie est constitué de 4 stages, où chaque stage contient un bloc de convolution suivies par n_i blocs d’identité, chacune des couches de convolution qui constituent les blocs d’un stage i est caractérisée par un nombre de filtres de convolution donné.

Ces 4 stages sont suivis d’une couche de sous échantillonnage de type Global Average pooling qui permet de réduire la taille de l’image à vecteur avec 2048 valeurs. Le schéma 6.11 suivant illustre la succession de ces stages.

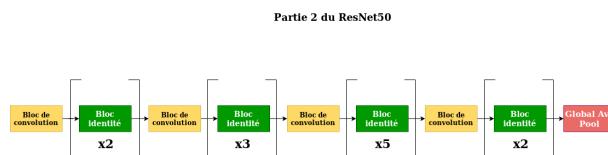


FIGURE 6.11 – Partie 2 du ResNet50.

3. Partie 3

Elle représente le réseau de neurones entièrement connecté qui permet de classifier l’image en entrée en se basant sur les caractéristiques de l’image extraites dans les deux parties précédentes.

Le réseau de neurones est constitué de 3 couches séparées par des dropout pour réduire le risque de surapprentissage.

La première couche contient 2048 neurones et qui correspond à la taille de l’image à l’issue de la partie 2 suivie d’une fonction d’activation ReLU et un dropout avec une probabilité de 0.4, la couche cachée quant à elle contient 256 neurones suivie par une ReLU et un dropout avec la même probabilité, et enfin la couche de sortie contient 5 neurones qui correspondent aux stades de la RD avec une fonction d’activation de type softmax. Le schéma 6.12 suivant illustre la 3ème partie qui correspond au réseau entièrement connecté.

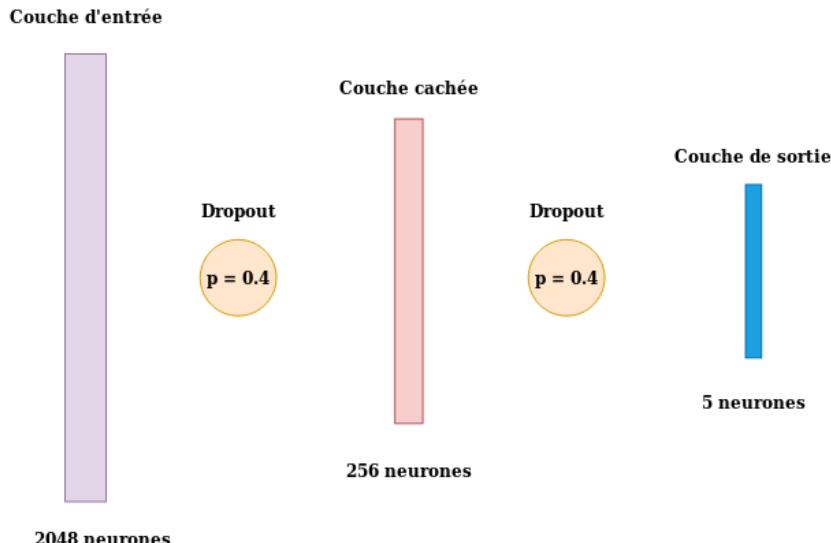
Partie 3 du ResNet50 : le réseau entièrement connecté

FIGURE 6.12 – Partie 3 du ResNet50 : réseau entièrement connecté.

6.4 Pourquoi avoir choisi ResNet50 ?

Notre choix s'est porté sur ResNet50 pour plusieurs raisons :

- C'est un modèle qui a fait ses preuves dans les compétitions et est très utilisé dans la détection des maladies.
- Les blocs résiduels qui constituent le réseau permettent de palier aux problèmes d'évanescence du gradient.
- C'est un réseau assez profond pour pouvoir apprendre à extraire des images des caractéristiques importantes dans la classification. C'est grâce aux blocs résiduels qu'on peut étendre le réseau sans se soucier du problème d'évanescence du gradient.
- Son temps d'apprentissage est moins important que plusieurs autres modèles(en raison du nombre de paramètres à apprendre) tel que :InceptionResNet [35].
- Possibilité de le combiner avec d'autres modèles dans le cadre d'un apprentissage ensembliste.
- Sa profondeur permet d'extraire les caractéristiques les plus importantes et de réduire au maximum la taille de l'image en entrée tout en gardant les informations déterminantes dans la classification.

Malgré ses multiples avantages il présente tout de même certains inconvénients :

- Le nombre de paramètres à entraîner qui dépasse les 27 741 189 paramètres, le temps nécessaire pour l'apprentissage s'accroît considérablement pour les bases d'apprentissages importantes.
- Son architecture complexe peut parfois mener à un surapprentissage, il est donc nécessaire d'implémenter des batch normalization et des dropout pour remédier à

cela, tel est décrit dans le paragraphe suivant.

6.5 Les techniques utilisées pour éviter le surapprentissage

Le surapprentissage ou l’overfitting est très courant dans les réseaux de neurones, il survient lorsque le modèle dépend beaucoup de la base d’apprentissage et n’arrive pas à bien généraliser la classification i.e. la précision de la classification décroît considérablement sur les images du test set.

Nous avons employé les techniques suivantes pour minimiser le risque de surapprentissage.

6.5.1 Normalisation du lot (batch normalization)

L’entraînement des réseaux de neurones profonds est beaucoup plus compliqué que ça en a l’air. En effet la distribution des données en entrée de chaque couche change en permanence vu que les paramètres des couches précédentes changent régulièrement lors de l’apprentissage. Par conséquent le modèle doit avoir un faible taux d’apprentissage (learning rate) mais surtout une bonne initialisation des paramètres pour pouvoir converger, ceci ralentit considérablement le temps d’apprentissage [36].

La technique de batch normalization est venue donc pour remédier à ce problème en normalisant les valeurs rentrées par les fonctions d’activations de chaque couche. Par conséquent la moyenne des valeurs rentrées par toutes les fonctions d’activation est égale à 0 et leur variance à 1. Le grand avantage qu’offre cette technique c’est qu’elle permet d’accélérer considérablement l’apprentissage et donc de réduire le nombre d’itérations par conséquent le temps d’exécution.

Le passage des données dans le réseau se fait par des sous ensembles appelés batch, ils sont généralement de taille 32 ou 64. Le principe de batch normalization est de soustraire la moyenne des valeurs que contient le batch et en divisant le tout par l’écart type rajouté à une infime constante pour éviter la division par zéro. Le résultat de normalisation est multiplié par un paramètre appelé gamma et rajouté à un autre paramètre beta, ces deux derniers sont des paramètres que le modèle apprenne au fur et à mesure. Les lignes suivantes représentent les formules mathématiques du batch normalization telles quelle sont décrites dans l’article original.

Entrée : Valeurs de x par lot : $\mathcal{B} = \{x_{1...m}\}$

Sortie : $y_i = BN_{\gamma,\beta}(x_i)$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m} \sum_i^m x_i$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m} \sum_i^m (x_i - \mu_{\mathcal{B}})^2$$

$$\hat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}}$$

$$y_i \leftarrow \gamma \hat{x}_i + \beta \equiv BN_{\gamma,\beta}(x_i)$$

où γ et β sont appris au fur et à mesure.

Dans notre modèle nous avons rajouté une couche de batch normalization après chaque produit de convolution et avant la fonction d'activation.

6.5.2 Dropout

Le terme Dropout désigne que certains neurones d'une couche donnée sont désactivés (leurs valeurs ne sont pas considérées) lors du passage d'un batch dans le réseau, ces neurones sont choisis aléatoirement avec une probabilité p . Cette technique pousse le modèle à considérer uniquement les caractéristiques importantes et limiter donc le risque d'apprentissage. Toutefois cette technique ralentit l'apprentissage car le modèle nécessite plus d'itérations pour converger.

Dans notre modèle nous avons eu recours à cette technique pour éviter le risque de surapprentissage, nous avons rajouté deux couches de dropout dans le réseau de neurones entièrement connecté avec une probabilité de 0.4.

6.5.3 Régularisation

La régularisation est une technique très efficace pour éliminer le surapprentissage. Il existe plusieurs types de régularisation la plus connue c'est **L2 Regularization** qui consiste à rajouter la somme de tous les poids synaptiques du modèle multipliée par une constante lambda à la valeur de la fonction objectif pour pénaliser les poids synaptiques ayant une grande valeur.

Dans notre travail nous avons opté pour un type un peu différent de régularisation qui est intégrée dans l'algorithme d'optimisation **AdamW** [37].

6.6 Data Augmentation

Le Data Augmentation est une technique utilisée principalement pour agrandir la base d'apprentissage et limiter le risque de surapprentissage en générant de nouvelles images à partir des images de la base d'apprentissage. Ex : en faisant des rotations, en améliorant le contraste, en réduisant la luminosité, en inversant l'image horizontalement et/ou verticalement .etc..Il existe deux types de data augmentation : data augmentation passive ;

des nouvelles images sont générées avant l'apprentissage et rajoutées à la base d'apprentissage. Le deuxième type c'est le data augmentation en temps réel ; les nouvelles images sont générées lors de l'apprentissage, l'avantage de cette dernière c'est qu'elle offre plus de liberté pour générer de nouvelles images du moment qu'on n'est pas contraints par l'espace de stockage, toutefois elle ralentit considérablement l'apprentissage.

Dans notre solution nous avons opté pour le data augmentation en temps réel en raison d'espace de stockage et la liberté qu'offre cette dernière. Nous avons utilisé 4 types d'augmentation au cours de l'apprentissage :

1. Rotation aléatoire

Les images du batch initial (de la couche d'entrée) sont aléatoirement pivotées par un angle α compris entre 0 et 40° .

2. Agrandissement et rognement

Les images du batch sont aléatoirement agrandies par un facteur de 0.4 et rognées à partir du centre.

3. Flip (inversion)

Les images du batch sont aléatoirement inversées horizontalement et/ou verticalement.

4. Saturation,luminosité et contraste

Ces trois paramètres sont aléatoirement modifiés dans les images du batch.

Un exemple d'utilisation du data augmentation sur un batch de 9 images est illustré dans les figures 6.13 et 6.14 suivantes.

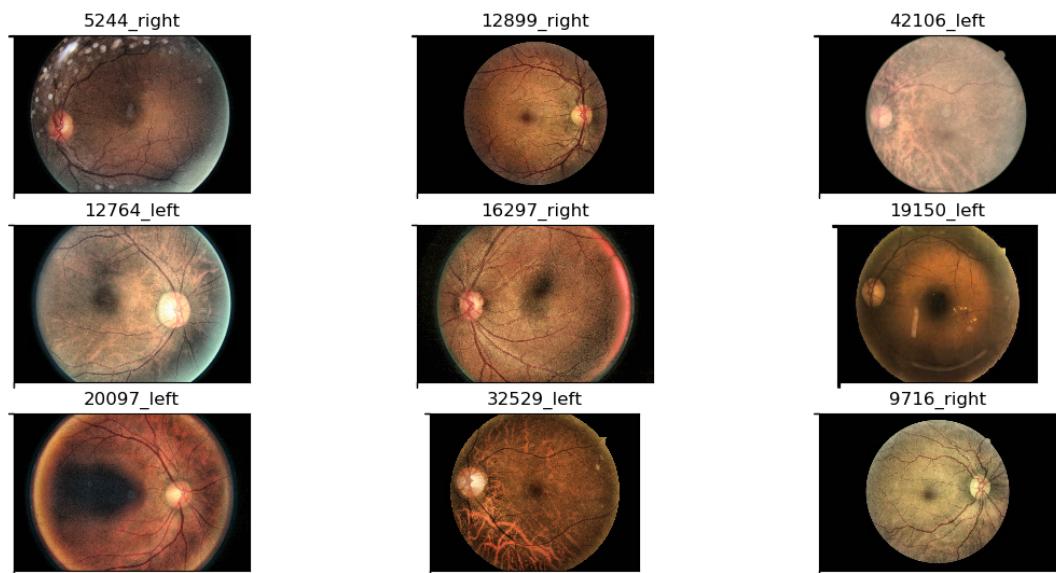


FIGURE 6.13 – Les images originales

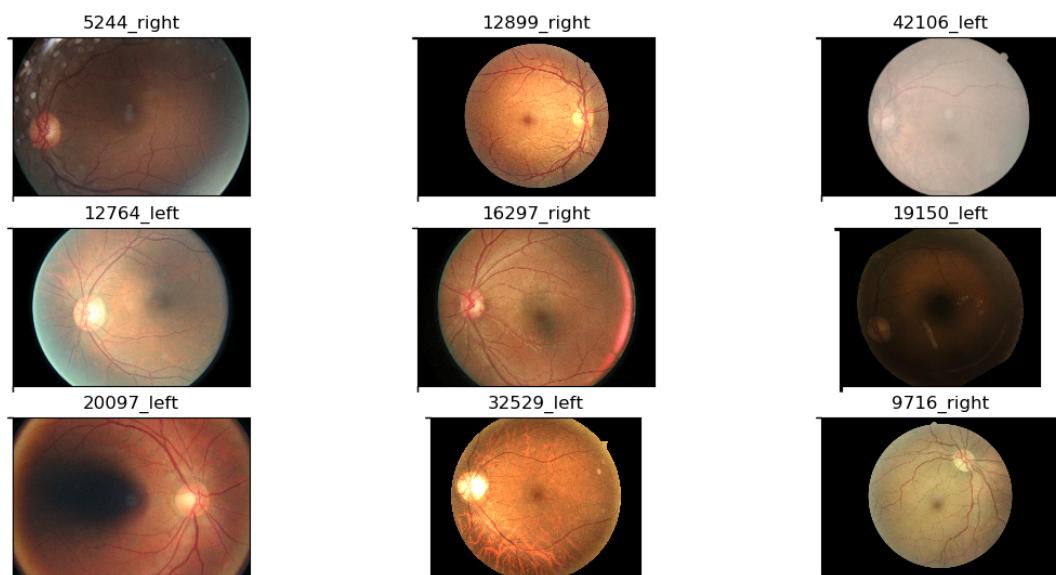


FIGURE 6.14 – Exemple de data augmentation

Réalisation et Implémentation de l'outil RetinoCheck

7.1 Introduction

Dans ce chapitre nous allons tout d'abord présenter notre plateforme professionnel et pédagogique pour la détection de la RD et ses différentes stades appelée “**RetinoCheck**”. Dans un second temps nous allons exposer les outils et technologies utilisés pour la mise en place de cette plateforme. Nous abordons ensuite l'architecture d'implémentation ainsi que l'organisation des différentes parties de “RetinoCheck”. Enfin, nous détaillerons un scénario d'utilisation de cette dernière.

7.2 Présentation générale de la plateforme

Afin de mettre notre modèle en valeur, nous avons mis en place un outil/plateforme pour rendre la prédiction du modèle d'apprentissage profond plus simple et facile à utiliser pour des simples utilisateurs ainsi que des ophtalmologues ou même des étudiants en médecine.

La principale fonctionnalité de la plateforme en question est la détection et la classification des images du fond d'œil, cette fonctionnalité est accessible à tous les types d'utilisateurs (utilisateurs simples / ophtalmologues). Le modèle accepte en entrée uniquement les images produites par un rétinographe (format : JPEG/JPG).

Comme tous les modèles d'apprentissage profond, notre modèle a besoin de beaucoup plus de données, qui dans notre cas sont des images de la rétine pour atteindre une précision excellente et fiable, l'idée est venue d'établir une fonctionnalité de collaboration entre médecins et simples utilisateurs, où les utilisateurs simples et médecins peuvent tester leurs images de la rétine sur notre modèle, ces images du fond de l'œil seront rajoutées à notre base d'apprentissage automatiquement et étiquetées par des ophtalmologues expérimentés à travers la plateforme.

7.3 Fonctionnalités de l’outil

Nous listerons ci-dessous les principales fonctionnalités fournies par la plateforme.

1. Gestion des utilisateurs

la plate-forme propose deux types d’utilisateurs, les utilisateurs simples sans privilège de modification sur la base d’apprentissage et les utilisateurs privilégiés (ophtalmologues) qui ont le privilège de modifier les étiquettes de l’ensemble de données et corriger un résultat erronée de la plateforme.

2. Détection et classification de la maladie

La principale fonctionnalité est la détection de la maladie (œil sain ou œil malsain) et d’indiquer le stade de la maladie en cas d’œil malsain. Le résultat retourné par le modèle de classification est de forme d’un vecteur de 5 valeurs de probabilité. Elles représentent en fait les probabilités que l’image du fond d’œil en entrée appartient à la classe i . Ces probabilités sont représentées sous forme de diagramme circulaire au niveau de la plateforme.

3. Génération d’un rapport détaillé

Après chaque test effectué par l’utilisateur, que ça soit un docteur ou un simple utilisateur, on a la possibilité de télécharger un rapport détaillé du test contenant l’image et le résultat obtenu par le modèle ainsi que d’autres informations concernant l’image et les paramètres utilisés pour faire le test.

4. Étiquetage des images du fond de l’œil

Les utilisateurs ayant des comptes de type “docteur” peuvent étiqueter des images non étiquetées, ces dernières seront donc ajoutées à notre base d’apprentissage pour les futures sessions d’entraînement du modèle. Ces utilisateurs peuvent aussi corriger le résultat du suggéré par le modèle.

5. Statistiques sur le modèle et la plateforme

La plateforme offre à travers une API toutes les informations sur le modèle courant dont la précision (accuracy), courbe d’apprentissage et l’architecture courante du modèle. De plus, on trouve des informations sur le nombre d’utilisateurs de la plateforme.

7.4 Présentation des outils et technologies utilisés

Dans cette première partie, nous allons expliquer nos choix des technologies utilisées afin de mettre en place l'outil conçu dans la phase de conception.

7.4.1 Langage de programmation

Le langage de programmation choisi est Python (version 3.7) en raison du succès considérable qu'il a eu ces dernières années sachant qu'il a été classé en première position des langages de programmation les plus utilisés en Juin 2020 dans le PYPL PopularitY of Programming Language (avec une part de 31.17 %). Du développement au déploiement et à la maintenance, Python aide les développeurs à être productifs et confiants quant au logiciel qu'ils construisent. Les avantages qui font de Python le meilleur choix pour l'apprentissage automatique et les projets basés sur l'IA incluent la simplicité et la cohérence, l'accès à d'excellentes bibliothèques et des Frameworks pour l'IA et l'apprentissage automatique (ML), la flexibilité, l'indépendance de la plateforme et une large communauté. Ceux-ci ajoutent à la popularité globale du langage.

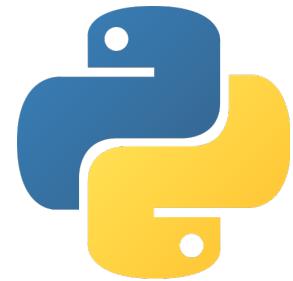


FIGURE 7.1 – Logo de python

7.4.2 Environnement de développement

Nous avons choisi de développer notre modèle dans Google Colab ou Colaboratory, qui est un service cloud, offert par Google (gratuit), basé sur Jupyter Notebook et destiné à la formation et à la recherche dans l'apprentissage automatique, car Google Colab contient toutes les bibliothèques python préinstallées. Cette plateforme permet d'entraîner des modèles de Machine Learning directement dans le cloud. Sans donc avoir besoin d'installer quoi que ce soit sur notre ordinateur à l'exception d'un navigateur. Pour le développement de la plateforme web qui relie le model avec les utilisateurs, nous avons choisi Visual Studio Code qui facilite énormément le processus de développement grâce aux différentes fonctionnalités offertes par ce dernier.



FIGURE 7.2 – Logo de Google Colab

7.4.3 Frameworks

Nous avons choisi de développer notre modèle d’apprentissage automatique avec Tensorflow. TensorFlow est une bibliothèque logicielle gratuite et open-source pour le flux de données et la programmation différentiable à travers une gamme de tâches. C’est une bibliothèque mathématique symbolique, et est également utilisée pour des applications d’apprentissage automatique telles que les réseaux de neurones. Il est utilisé à la fois pour la recherche et la production chez Google. De plus, nous avons choisi d’utiliser le Framework Python “Django” (version 2.1.15) qui nous permettra le développement accéléré, simple et organisé de notre outil en respectant au mieux les principes de développement tels que DRY (**Don’t Repeat Yourself**), le faible couplage et le **Open Close Principle**. Il nous permettra aussi d’intégrer le modèle avec la plateforme. Django est connu comme étant un projet open source mature, stable et sûr, disposant d’une large communauté de contributeurs et une très riche documentation. Il est de plus utilisé en production par de grands sites tels que Instagram, Spotify, OpenStack, Mozilla et même le fameux YouTube.



FIGURE 7.3 – Logo de Django

7.4.4 Stockage de la base d’apprentissage

Afin de pouvoir accéder aux images dans la base d’apprentissage qui est relativement volumineuse (85 GB), on a utilisé Google Drive. Google Drive est un service de stockage et de synchronisation de fichiers développé par Google. Google Drive permet aux utilisateurs de stocker des fichiers sur leurs serveurs, de synchroniser des fichiers sur plusieurs appareils et de partager des fichiers.



FIGURE 7.4 – Logo de Google Drive

7.4.5 Bibliothèques

Afin d’implémenter certaines fonctionnalités offertes par notre outil, nous avons eu recours à un ensemble de bibliothèques Python. Nous présentons les plus importantes (principales) dans le tableau 7.1 ci-dessous

| Bibliothèques | Brève description | Utilisation |
|----------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| JSON (2.0.9) | JSON : permet de convertir un contenu JSON en dictionnaire Python et vice-versa. | |
| requests (2.23.0) | requests : permet l'envoie de tout type de requêtes HTTP. | |
| djangorestframework (3.11.0) | Djangorestframework : Permet la création des RESTful APIs flexibles et sécurisées. | Implémentation d'une RESTful API pour gérer les requêtes utilisateurs venant de la plateforme et permettre l'intégration du modèle avec la plateforme ainsi que l'utilisateur. |
| Random | Random : permet la génération pseudo-aléatoire de nombres tels que les entiers ou le choix aléatoire de valeurs à partir d'un ensemble initial. | Générer les poids initiaux pour le modèle |
| Numpy (1.18.5) | Numpy : Permet une manipulation rapide et efficace sur les tableaux. | |
| scikit-learn (0.23.1) | Scikit-learn : Scikit-learn est une bibliothèque d'apprentissage automatique de logiciels gratuits pour le langage de programmation Python. Elle propose différents algorithmes de classification, de régression et de clustering, y compris SVM, random forest, gradient boosting, k-means. | - Implémentation des parties dans l'architecture du Resnet50 - Validation du modèle sur le test set |
| Sickit-image (0.17.2) | Scikit-Image : scikit-image est une bibliothèque de traitement d'images open-source pour le langage de programmation Python. Elle comprend des algorithmes de segmentation, de transformations géométriques, de manipulation de l'espace colorimétrique, d'analyse, de filtrage, de morphologie, de détection de caractéristiques, etc. | - Pré-traitement de l'image en entrée. |
| Pillow (7.1.2) | Pillow : Permet la manipulation des images et contient plusieurs algorithmes de traitement d'images. | Effectuer l'étape du prétraitement qui consiste à recadrer l'image du fond d'œil et mettre en valeurs les détails importants de l'image en question. |
| reportlab (3.5.42) | Reportlab : Une bibliothèque Open Source Python pour générer des PDF et des graphiques. | Générer un rapport détaillé sur le test de rétinopathie diabétique. |

TABLE 7.1 – Bibliothèques utilisées dans l'implementation

7.5 Présentation de l’architecture de Django

Notre outil suit une architecture MVT (Model View Template) qui est l’architecture caractérisant le Framework Django. Elle est composée de trois éléments :

— **Model :**

Un modèle est une représentation des données stockées permettant de faciliter l’interaction avec différentes BDDs à travers l’ORM (Object Relational Mapping) de Django, dont le rôle est de traduire les classes Python en requêtes (et vice versa).

— **View :**

Une vue est une fonction Python recevant en entrée une requête HTTP et retournant en sortie une réponse compréhensible par le navigateur. Son rôle est de traiter la requête, elle peut aussi récupérer des objets de la BDD à travers les modèles lorsque cela est nécessaire. Une vue est exécutée si l’URL de la requête correspond à l’une des règles de routage qui lui sont associées.

— **Template :**

Un Template est un fichier HTML pouvant accéder à des objets Python à travers un contexte lié à une vue. Il permet également de définir des variables et des balises grâce au langage de gabarit offert par Django.

L’un des principaux objectifs de Django est le faible couplage et la forte cohésion entre les couches d’un système. Ceci est assuré par l’architecture MVT et par la structure particulière des projets Django ; un projet Django est constitué d’une ou de plusieurs applications Django, chacune d’entre elles est indépendante et peut être réutilisée dans d’autres projets. De manière concrète, une application Django est un package Python comprenant des modèles, des vues et des Templates (entre autres) et destinée à réaliser un ensemble de tâches.

Nous détaillerons dans la section suivante l’architecture de notre outil.

7.6 Présentation de l'architecture de la plateforme

Notre outil se compose de quatre applications Django :

1. L'application "model_prediction" :

cette application gère l'alimentation des images du fond d'œil dans le modèle d'apprentissage automatique et renvoie les probabilités résultantes de l'opération de prédiction à l'interface utilisateur ainsi que d'autres informations concernant le test.

2. L'application "REST" :

une application RESTful fournissant des statistiques sur notre modèle, y compris la précision du modèle, la courbe d'apprentissage et l'historique de l'apprentissage, en plus d'autres statistiques sur les utilisateurs et leurs activités dans la plate-forme.

3. L'application "authentication" :

Elle s'occupe de tout le processus d'authentification et les autorisations des utilisateurs.

4. L'application "CORE_APP" :

Cette application est le cœur de notre plateforme, elle relie les trois applications ci-dessus. Elle gère les utilisateurs et leurs rôles à l'aide de l'application d'authentification, affiche différentes statistiques sur le modèle en utilisant l'application REST et surtout, elle affiche un résultat détaillé du test de la rétinopathie diabétique à l'utilisateur ainsi que la génération de rapports.

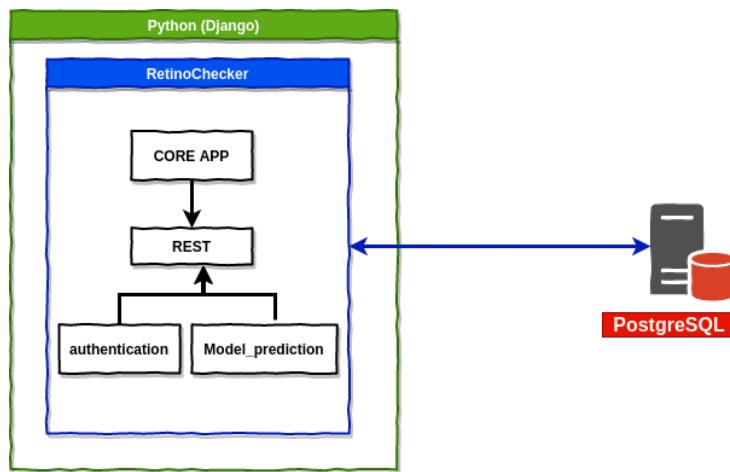


FIGURE 7.5 – architecture de la plate-forme RetinoCheck

Ces quatre applications Django partagent tous une BDD commune. Notre choix s'est porté sur PostgreSQL, c'est la BDD la plus utilisée dans la communauté Django pour les projets qui passe à la phase de production.

Dans ce qui suit on présentera les différentes structures de données (modèles) manipulées par notre outil.

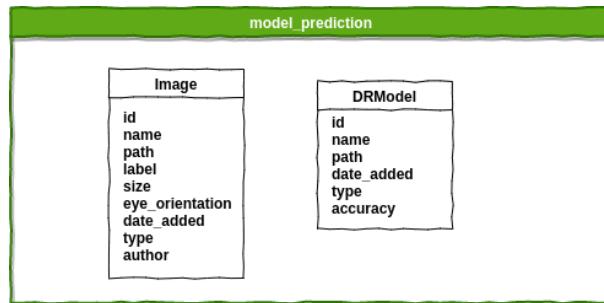


FIGURE 7.6 – Modèles de l’application model_prediction

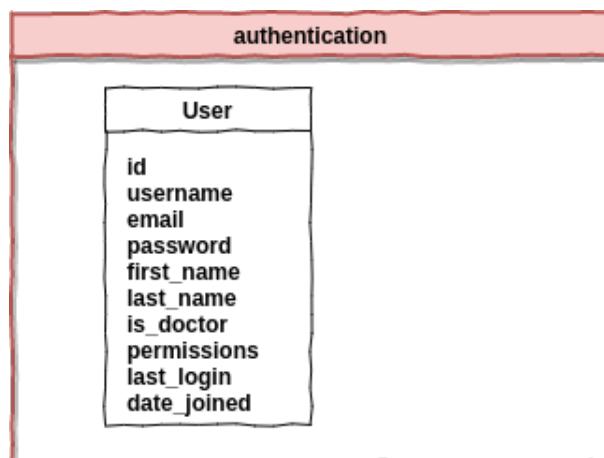


FIGURE 7.7 – Modèles de l’application authentication

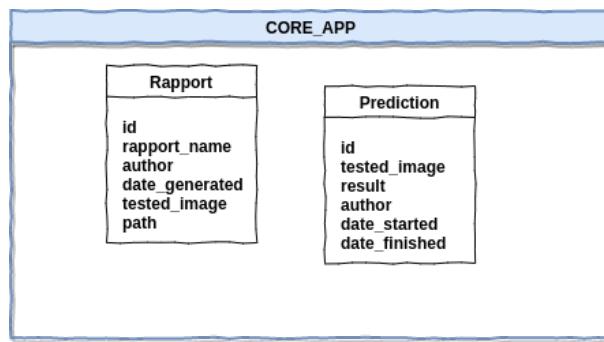


FIGURE 7.8 – Modèles de l’application CORE_APP

7.7 Implémentation du modèle de classification

Afin que nous puissions utiliser le modèle ResNet50, nous avons décidé de mettre en œuvre l’architecture à partir de zéro. On a implémenté cette architecture en utilisant les tensors de tensorflow et quelques bibliothèques de gestions de tableaux et la génération des nombres aléatoires. Toutes les couches du Resnet50 ont été implémentées à partir de zéro ainsi que la fonction du forwarding, l’initialisation des poids initiaux et le concept du Skip Connections.

Code Source : https://github.com/omega-coder/PFE_RD

7.7.1 Pré-traitement des images

Pour l’implémentation de l’algorithme de pré-traitement, on a opté pour l’algorithme CLAHE afin de préparer les images avant l’entraînement. On a implémenté l’algorithme CLAHE en utilisant des bibliothèques de traitement des images tel que Pillow et sickit-image.

7.7.2 Validation du modèle

Pour avoir la précision (accuracy) du modèle après l’entraînement, on a utilisé Sklearn et tensorflow afin d’évaluer le modèle sur un ensemble de données pour la validation et un autre pour le test.

7.8 Scénarios d’utilisation

Avant de détailler les différents scénarios de l’utilisation de notre outil, il est important de préciser que RetinoCheck, qui est essentiellement notre modèle en mode production est muni d’une interface graphique facilitant l’utilisation de l’outil, mais il est aussi à noter que le but essentiel de notre travail est l’élaboration d’un modèle d’apprentissage profond pouvant détecter la maladie en question ainsi que ses différents stades d’une manière

automatique et précise. On suppose donc que notre modèle sera utilisé par des personnes expérimentées du domaine qui auront toutes les compétences nécessaires pour pouvoir comprendre les différents résultats du modèle, entraîner le modèle sur d’autres ensembles de données et même changer quelques paramètres du modèle.

Nous présentons sous forme de captures d’écrans les cas d’utilisations et fonctionnalités de la plateforme.

1. Tester une image du fond d’œil sur le modèle

Dans ce scénario, l’utilisateur (simple ou docteur) va choisir l’opération nommée “prédiction”, il va ensuite charger une image du fond d’œil dans l’application web. Enfin, après avoir lancé la détection, il récupérera les résultats de la détection et sauvegardera des résultats plus détaillés grâce à la fonctionnalité de génération de rapport.

— Choix de l’image du fond d’œil

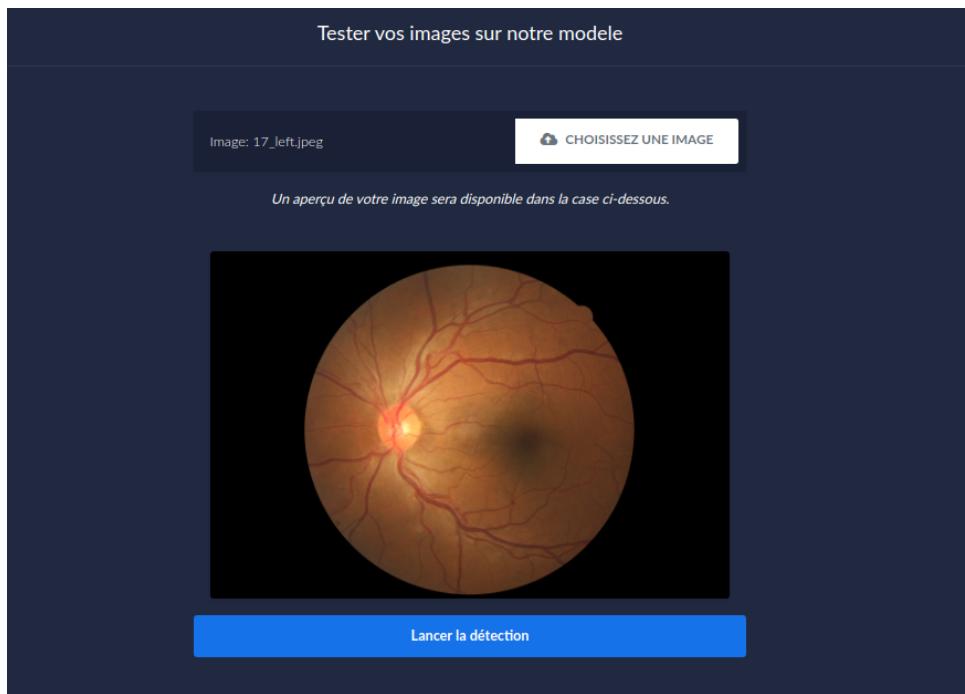


FIGURE 7.9 – Choix de l’image du test

— Lancement de la détection et récupération du rapport de tests

Dans cette étape, l’utilisateur récupère le résultat du test sous forme d’un graphique circulaire contenant les probabilités des différents stades et un message contenant le stade le plus probable, en plus, un docteur peut corriger le résultat du modèle, si l’image n’est pas encore étiquetée et que le modèle donne un résultat erroné.

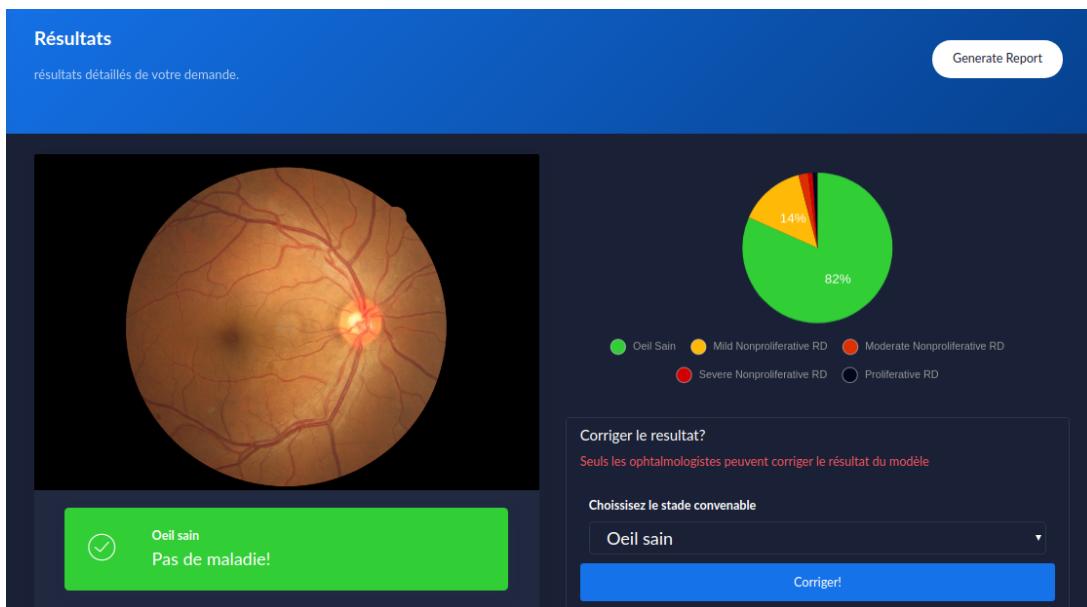


FIGURE 7.10 – Résultats du test sur la plateforme

Après consultation des résultats, l'utilisateur peut maintenant télécharger un rapport PDF plus détaillé.

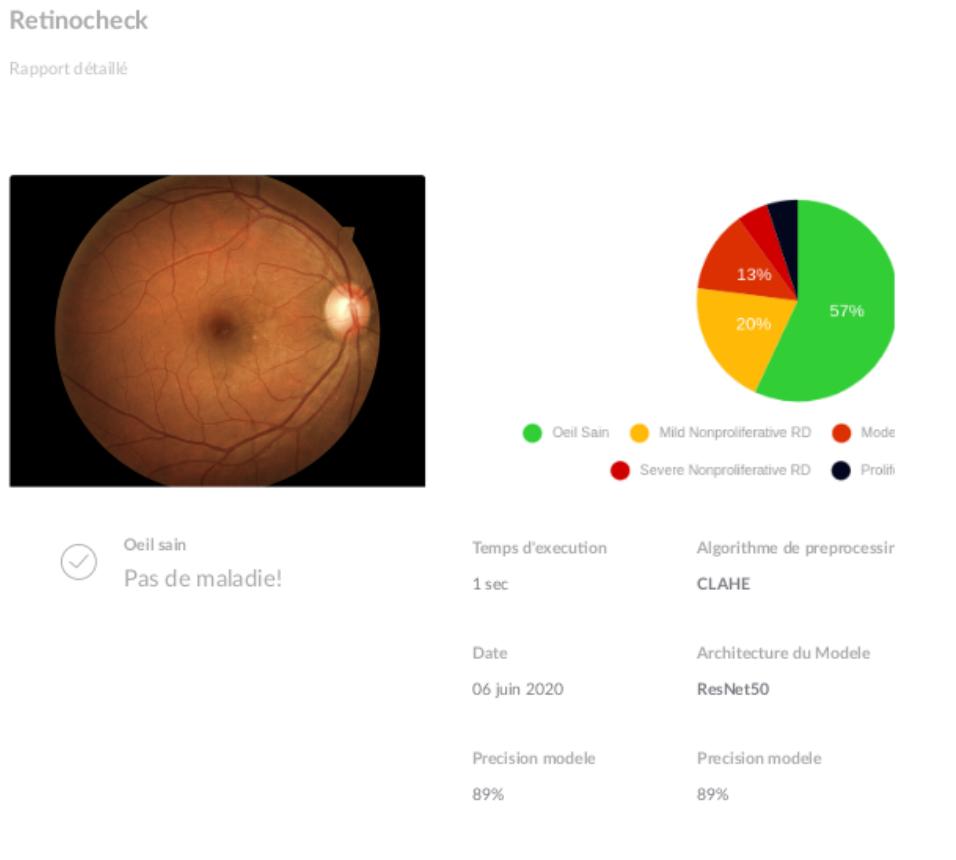


FIGURE 7.11 – Rapport du test sur la plateforme

2. Etiquetez des images du fond d’œil

La plateforme offre la possibilité d’étiqueter les images chargées par d’autres utilisateur.

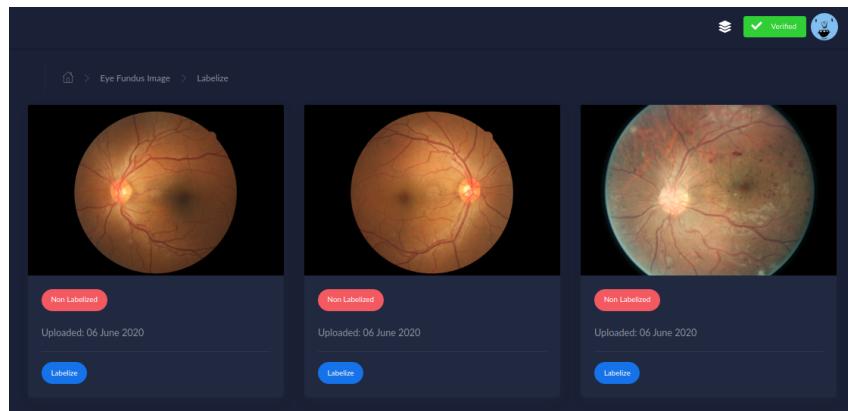


FIGURE 7.12 – Étiquetage des images du fond d’œil

Dans ce scénario, le docteur (Ophtalmologue), peut choisir le stade adéquat de la maladie, ou choisir que c'est un œil sain. Cette opération de collaboration est essentielle pour enrichir notre base d'apprentissage et améliorer la précision et la spécifité du modèle et de la détection de la RD dans les futurs tests.

NOTE

Seuls les utilisateurs avec un compte médecin vérifié ont accès à cette fonctionnalité

7.9 Conclusion

Dans ce chapitre, nous avons présenté les différentes technologies utilisées pour développer notre outil ainsi que l’architecture suivie pour son implémentation. Un scénario d’utilisation a ensuite été exposé pour décrire son fonctionnement à travers des captures d’écran en mettant en évidence les différentes fonctionnalités de la plateforme.

TROISIÈME PARTIE

TESTS ET RÉSULTATS

Entrainement du modèle

8.1 Introduction

La phase d'apprentissage est une étape cruciale et très importante. En effet l'efficacité du modèle dépend entièrement de cette phase, il est donc nécessaire de bien choisir les paramètres du modèle. Le choix de ces derniers est une tache fastidieuse, il nous a fallu tester plusieurs combinaisons de valeurs pour aboutir à un résultat satisfaisant.

8.1.1 Valeurs des hyperparamètres

Les valeurs des hyperparamètres ci-dessous ont été obtenues après plusieurs exécutions et tests. Elles représentent donc la meilleure combinaison qu'on a pu trouver après plusieurs exécutions.

- **Nombre d'itérations :** Le nombre d'itérations est déterminée par **la courbe d'apprentissage**; lorsque la direction de la courbe des valeurs de la fonction objectif s'inverse on arrête l'apprentissage.
- **Taux d'apprentissage :** Le taux d'apprentissage décroît au fur et à mesure que le modèle avance. la valeur initiale du taux d'apprentissage est **0.001**.
- **L'algorithme d'optimisation :** L'algorithme d'optimisation utilisé est **AdamW**. Nous avons opté pour ce dernier car il intègre un type de régularisation pour prévenir le surapprentissage.
- **La fonction objectif :** Nous avons opté pour la fonction **categorical cross-entropy**, cette dernière pénalise significativement les mauvaises classifications.
- **La fonction d'activation de la couche de sortie :** Notre avons utilisé la fonction **softmax** qui est la généralisation de la fonction sigmoid pour une classification multiclass. Elle retourne une probabilité que l'image appartient à une classe i.
- **Initialisation des poids :** L'initialisation des poids synaptiques est faite avec la technique de **glorot uniform**¹.

1. Appelé aussi *Xavier initializer*, Il retourne des valeurs uniformément distribuées et comprises entre $-c$ et c où $c = \sqrt{\frac{6}{fan_in + fan_out}}$, où fan_in est le nombre de neurones de la couche i et fan_out est le nombre de neurones de la couche i+1

8.1.2 Courbes d'apprentissage

La courbe d'apprentissage est une représentation graphique des variations des fonctions objectif d'apprentissage et de validation (Loss function) en fonction des épisodes (epochs). Elle permet entre autres de détecter le point de saturation (lorsque la valeur de la fonction objectif d'apprentissage cesse de décroître), le point où le modèle commence à surapprendre (lorsque la courbe de la fonction objectif de validation évolue dans le sens inverse de la fonction objectif d'apprentissage) et c'est à ce moment-là que l'apprentissage doit être arrêté.

La figure 8.1 suivante représente la courbe d'apprentissage de notre modèle.

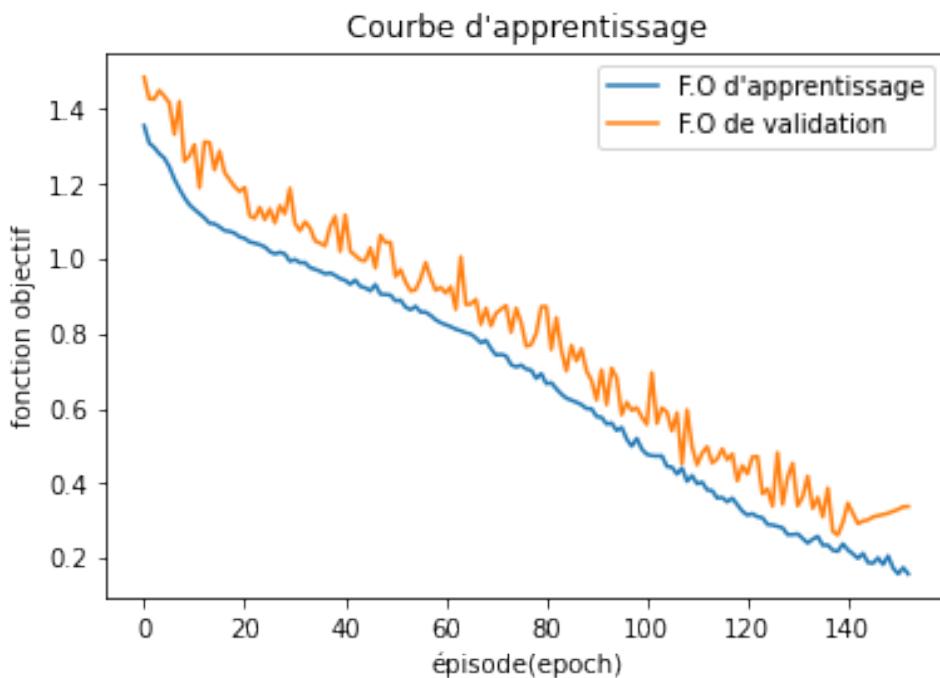


FIGURE 8.1 – La courbe d'apprentissage.

Le graphe ci-dessus est une représentation de la courbe d'apprentissage de notre modèle, où les variations des valeurs des fonctions objectif de l'apprentissage et de validation en fonction du nombre d'épisodes (epochs) sont représentées.

Nous remarquons que les valeurs des fonctions objectifs décroissent au fil de l'apprentissage. Les courbes montrent également une forte décroissance des valeurs de fonctions objectifs dans les 100 premières épisodes. La vitesse de décroissance commence à ralentir après les 110 épisodes pour arriver au point de saturation vers la 140ème épisode où nous remarquons un changement de direction de la courbe de validation qui révèle donc un début de surapprentissage. L'entraînement du modèle est donc arrêté après 140 épisodes.

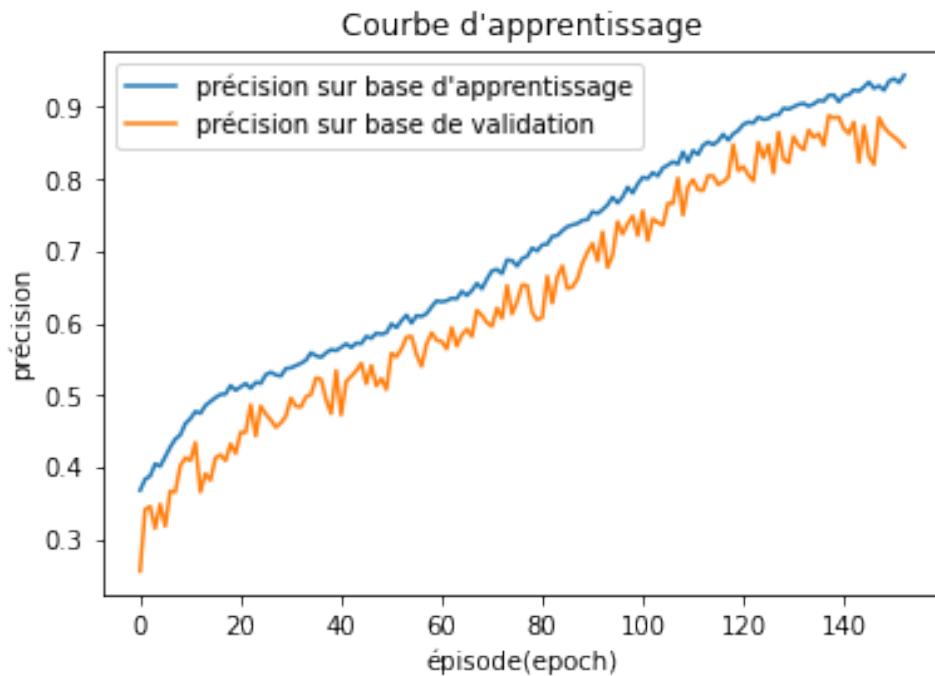


FIGURE 8.2 – La courbe d'apprentissage (précision).

La figure 8.2 ci-dessus représente les variations de la précision du modèle et sur la base d'apprentissage et sur la base de validation. La précision du modèle s'améliore au fur et à mesure que ce dernier apprend à extraire les caractéristiques déterminantes dans la classification. La précision du modèle sur la base de validation continue à progresser jusqu'à arriver au point de saturation au bout de la 140ème épisode où le modèle commence à surapprendre.

Tests et Résultats

9.1 Introduction

Une fois la phase d'apprentissage terminée et le modèle entraîné, ce dernier doit être évalué et testé sur une base de test contenant des images du fond d'oeil de différentes classes. En effet plusieurs métriques d'évaluation peuvent être utilisées pour juger l'efficacité du modèle. Dans le paragraphe suivant nous allons énumérer les différentes métriques utilisées.

Dans le cas d'une classification multiclasse, la signification des termes : vrai positif (VP), vrai négatif (VN), faux positif (FP) et faux négatif (FN) change légèrement du cas d'une classification binaire.

VP_i : Les images de la classe i et qui sont correctement classées dans la classe i.

VN_i : Les images des classes autre que i et qui ne sont pas classées dans la classe i.

FP_i : Les images des classes autre que i et qui sont faussement classées dans la classe i.

FN_i : Les images de la classe i et qui ne sont pas classées dans la classe i.

1. **Précision (Accuracy)** : Elle représente la portion des images correctement classifiées sur le nombre total des images.

$$precision = \frac{1}{5} \sum_i \frac{VP_i + VN_i}{VP_i + VN_i + FP_i + FN_i} \quad (9.1)$$

2. **Sensibilité** : La sensibilité du modèle est sa capacité à détecter un maximum de malades (c'est-à-dire à avoir le moins de faux négatifs). La valeur de sensibilité d'une classe i est calculée avec la formule ci-dessous.

$$sensibilite_i = \frac{VP_i}{VP_i + FN_i} \quad (9.2)$$

3. **Spécificité** : La spécificité du modèle est sa capacité à ne détecter que les malades (avoir le moins de faux positifs). La valeur de spécificité d'une classe i est calculée avec la formule ci-dessous.

$$specifique_i = \frac{VN_i}{VN_i + FP_i} \quad (9.3)$$

9.2 Résultats

Avec notre modèle entraîné à partir du zéro nous avons pu obtenir une précision de 86.95% sur une base de test avec 2000 images réparties sur 5 classes. Le tableau 9.1 et la figure 9.1 suivants résument les résultats obtenus sur la base de test.

| | Stade | Spécificité | Sensibilité |
|-----------------|-------|-------------|-------------|
| Œil sain | 90 % | 84 % | |
| RD minime | 89 % | 83 % | |
| RD modérée | 88 % | 91 % | |
| RD sévère | 94 % | 90 % | |
| RD proliférante | 92 % | 90 % | |

TABLE 9.1 – Spécificité et sensibilité.

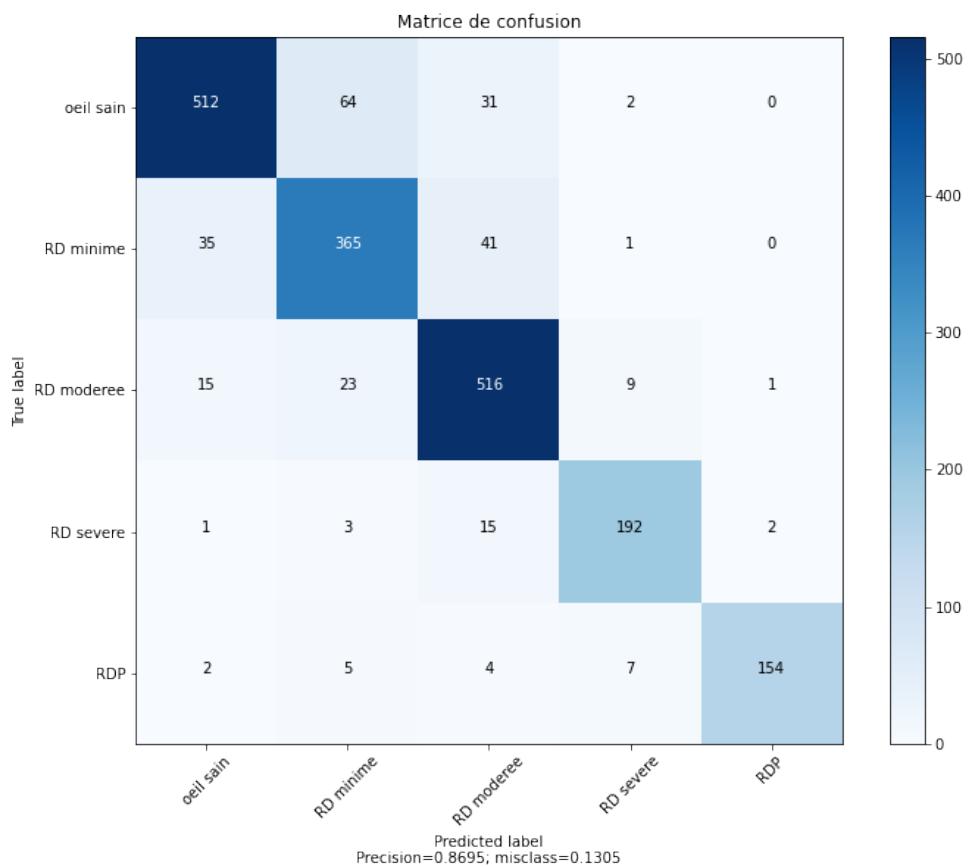


FIGURE 9.1 – Matrice de confusion.

Apprentissage par transfert

10.1 Introduction

L'apprentissage par transfert est une technique de Machine Learning qui consiste à transférer les valeurs des poids synaptiques d'un modèle pré entraînée sur base d'apprentissage A vers un autre modèle possédant la même architecture qui sera entraînée sur une base d'apprentissage B.

Ce transfert peut être d'une manière sélective ; on peut opter pour un transfert partiel de paramètres, ex : seules les valeurs des premières couches qui sont transférées vers le second modèle.

Cette technique est très utile dans le cas où la base d'apprentissage en autre faveur n'est pas suffisante pour un bon apprentissage.

Dans ce qui suit nous allons appliquer la technique d'apprentissage par transfert et voir son impact sur la vitesse d'apprentissage et sur les résultats de tests. Dans un second lieu nous allons l'appliquer d'une manière sélective et comparer les résultats.

10.2 Impact d'apprentissage par transfert sur la vitesse d'apprentissage.

Dans un premier temps, nous avons importé les poids synaptiques d'un modèle ResNet50 préentraîné sur IMAGENET. Tous les poids de notre modèle prennent donc les valeurs du modèle à l'exception des poids du réseau entièrement connecté qui sont initialisés avec l'algorithme glorot_uniform.

L'apprentissage est donc relancé à nouveau sur la base d'apprentissage des images du fond d'oeil. Les deux figures 10.1 et 10.2 suivantes représentent les deux courbes d'apprentissage relatives à la fonction objective et à la précision respectivement.

Le graphe 10.1 ci-dessus représente les variations des valeurs des fonctions objectif de l'apprentissage et de validation en fonction du nombre d'itération.

On constate clairement que l'apprentissage est beaucoup plus rapide que l'apprentissage à zéro (voir 8.1) ceci est traduit par l'allure des courbes. On voit une allure descendante très prononcée au début de l'apprentissage qui commence à se stabiliser après 60

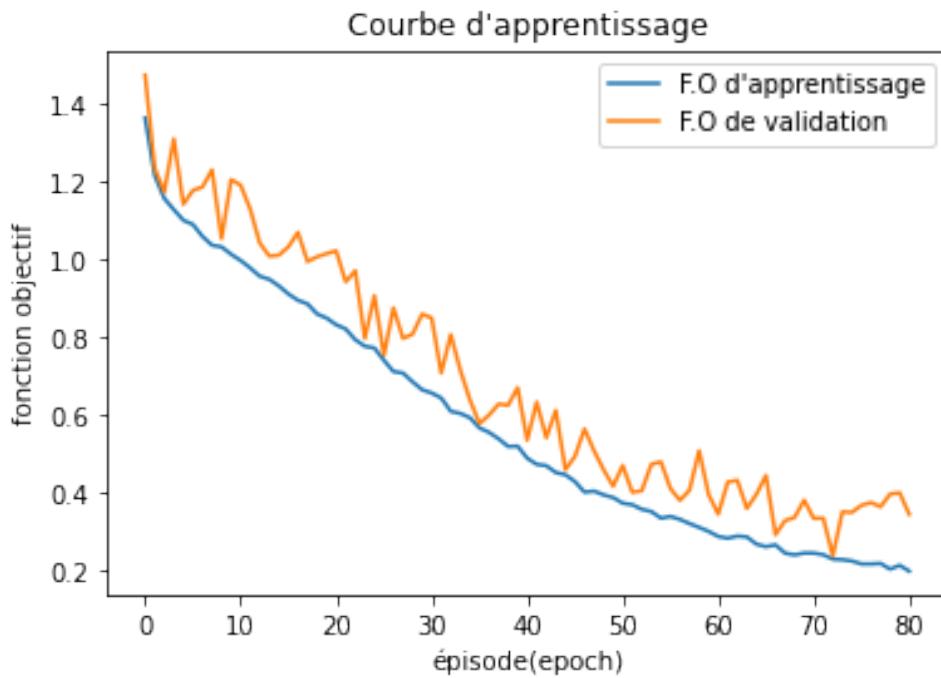


FIGURE 10.1 – La courbe d'apprentissage dans le cas d'un apprentissage par transfert.

épisodes. Après quoi, les valeurs des fonctions objectifs continuent de descendre à petite vitesse. Au bout des 80 épisodes le modèle atteint le point de saturation.

On distingue donc une nette accélération de l'apprentissage par rapport à l'initialisation à froid du modèle.

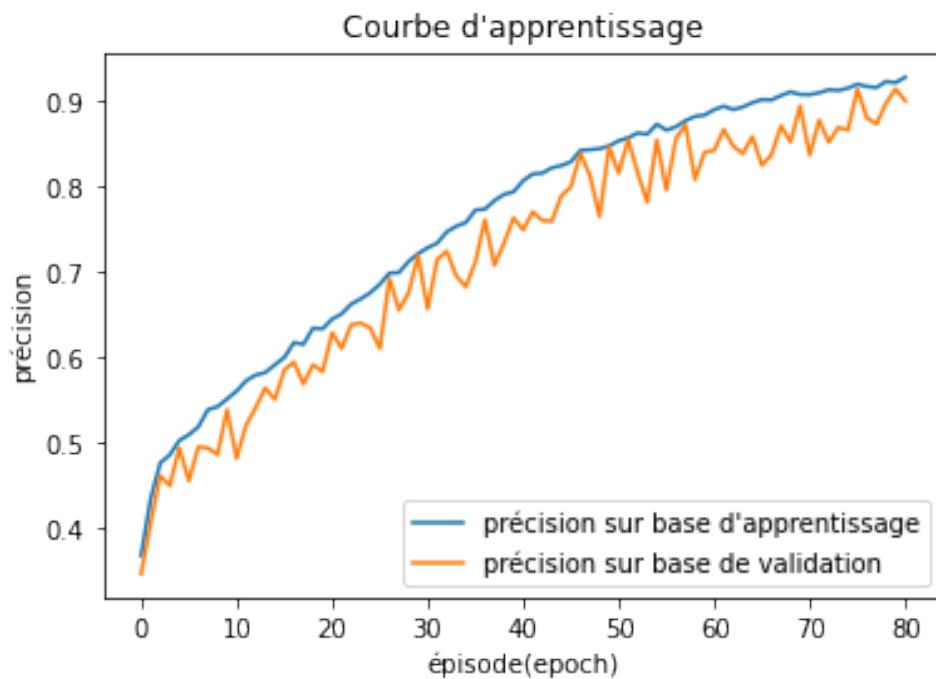


FIGURE 10.2 – La courbe d'apprentissage (précision) dans le cas d'un apprentissage par transfert.

10.2. Impact d'apprentissage par transfert sur la vitesse d'apprentissage.

La figure 10.2 ci-dessus représente les variations de la précision du modèle sur la base d'apprentissage et sur la base de validation.

Le tableau 10.1 et la figure 10.3 suivants représentent les résultats obtenus sur la base de test.

| Stade | Apprentissage par transfert | | Apprentissage à zéro | |
|-----------------|-----------------------------|-------------|----------------------|-------------|
| | Spécificité | Sensibilité | Spécificité | Sensibilité |
| Œil sain | 92 % | 82 % | 90 % | 84 % |
| RD minime | 91 % | 96 % | 89 % | 83 % |
| RD modérée | 88 % | 90 % | 88 % | 91 % |
| RD sévère | 94 % | 88 % | 94 % | 90 % |
| RD proliférante | 92 % | 95 % | 92 % | 90 % |

TABLE 10.1 – Spécificité et sensibilité du modèle dans le cas d'un apprentissage par transfert et dans le cas d'un apprentissage à zéro.

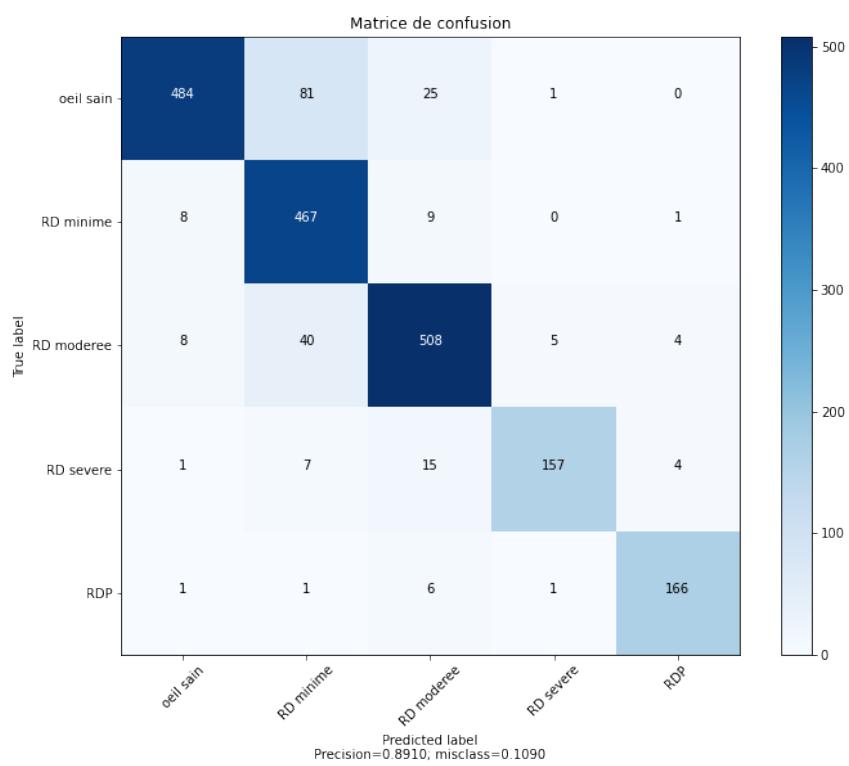


FIGURE 10.3 – Matrice de confusion du modèle dans le cas d'un apprentissage par transfert.

Les résultats consignés dans le tableau et la figure 10.3 ci-dessus révèle une légère amélioration des résultats par rapport aux résultats précédents. Nous pouvons donc déduire que l'initialisation du modèle joue un rôle très important dans l'apprentissage de ce dernier.

10.3 Apprentissage par transfert sélectif

Dans cette partie nous allons appliquer un apprentissage par transfert d'une manière sélective et voir l'utilité des différentes couches du modèle dans la classification finale.

Dans un premier lieu nous allons transférer les poids du modèle pré entraîné sur IMAGENET, et par la suite figer les poids du premier stage du ResNet50. Les valeurs des poids restants sont mises à jour au fur et à mesure de l'apprentissage.

La figure 10.4 suivante représente la matrice de confusion obtenu après l'évaluation du modèle sur la même base de test.

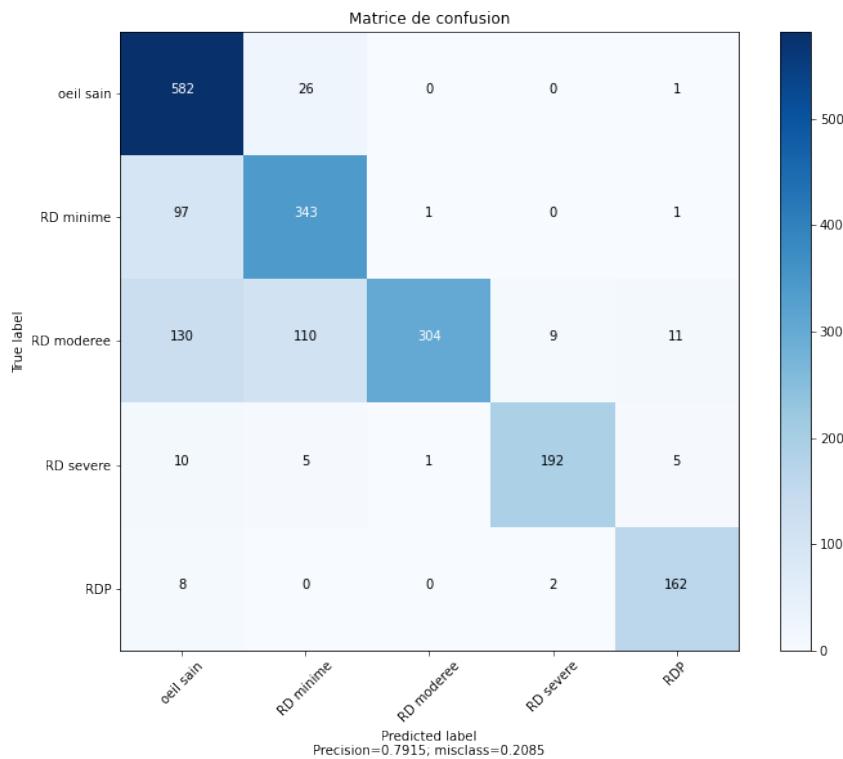


FIGURE 10.4 – Matrice de confusion du modèle dans le cas d'un apprentissage par transfert sélectif.

| Stade | Spécificité | Sensibilité |
|-----------------|-------------|-------------|
| Oeil sain | 82 % | 96 % |
| RD minime | 89 % | 78 % |
| RD modérée | 91 % | 54 % |
| RD sévère | 92 % | 90 % |
| RD proliférante | 88 % | 94 % |

TABLE 10.2 – Spécificité et sensibilité dans le cas d'un apprentissage par transfert sélectif.

On note que l'apprentissage du modèle est beaucoup plus rapide que les derniers apprentissages vu que le nombre des poids synaptiques à apprendre a diminué, (Voir figures 10.5 et 10.6).

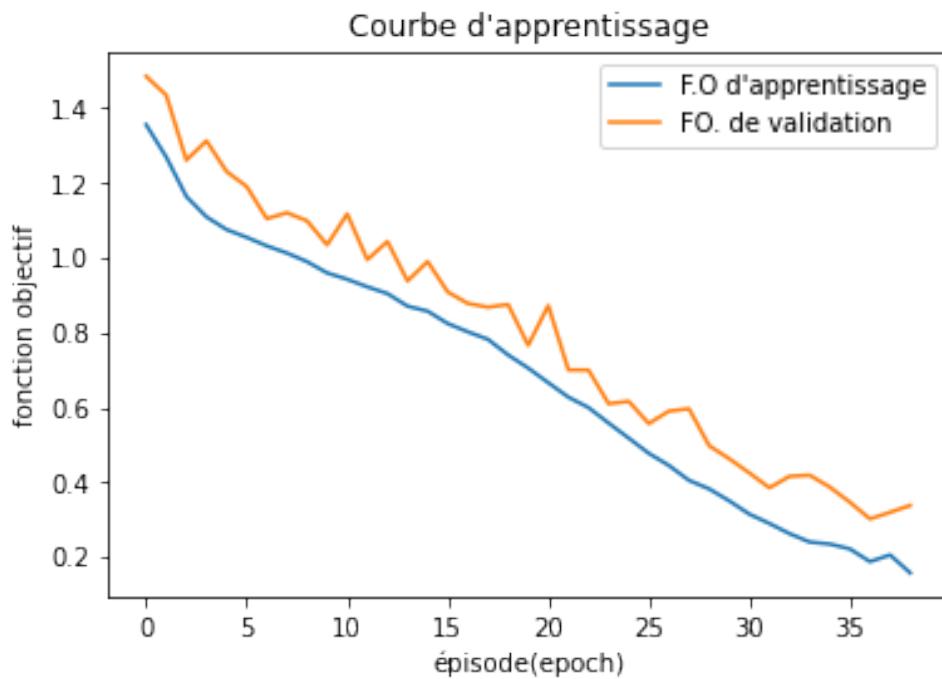


FIGURE 10.5 – Courbe d'apprentissage dans le cas d'un apprentissage par transfer sélectif.

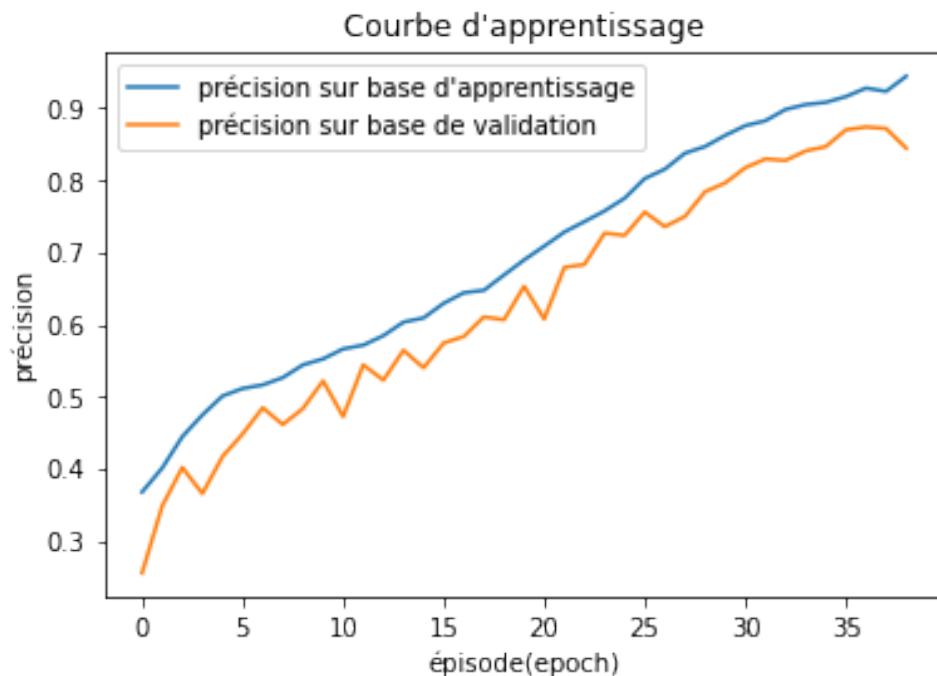


FIGURE 10.6 – Courbe d'apprentissage(précision) dans le cas d'un apprentissage par transfer sélectif.

On constate que le modèle a du mal à extraire les caractéristiques les plus importantes dans la classification de la RD. En outre le modèle atteint le point de saturation très vite lors de l'apprentissage avec de mauvais résultats par rapport aux deux derniers.

Il est donc inutile donc d'adopter la technique d'apprentissage par transfert sélectif pour les stages restants du modèle pour les raisons suivantes :

- Les résultats obtenus en figeant les poids du premier stage n'apportent aucune amélioration.
- Les couches profondes des modèles CNN sont spécifiques au problème donné i.e. plus on s'enfonce dans le modèle plus l'extraction des caractéristiques à partir de l'image est spécifique au problème. Par conséquent figer les poids des stages avancés du modèle ne permettra pas d'améliorer les résultats précédents, car ces poids sont propres à la base source (IMAGENET).

10.4 Conclusion

Dans ce chapitre nous avons exposé l'utilité d'apprentissage par transfert et l'amélioration qu'il peut apporter à notre solution. En effet l'utilisation de l'apprentissage par transfert a permis un apprentissage très rapide par rapport à l'apprentissage à froid, ainsi qu'une légère amélioration des résultats du modèle.

Nous avons également montré que l'apprentissage par transfert d'une manière sélective n'a apporté aucune amélioration au modèle, car les poids des couches supérieures transférés ne correspondent pas à notre problème. De ce fait le modèle était dans l'incapacité d'extraire les caractéristiques importantes dans la classification. Ceci s'est traduit par la dégradation des résultats obtenus.

Conclusion

L'ultime objectif de ce travail était de proposer dans un premier temps une méthode de détection et de classification automatique de la rétinopathie diabétique en utilisant des techniques de traitement d'images et un modèle basé sur les réseaux de neurones convolutionnels. Dans un second temps, nous avons exploré les améliorations que l'apprentissage par transfert peut apporter à notre modèle de base. Nous avons également conçu une plateforme dotée de plusieurs fonctionnalités pour faciliter l'interaction avec le modèle de classification et pour enrichir une future base d'apprentissage.

Par ailleurs l'application d'apprentissage par transfert nous a permis d'améliorer la précision de notre modèle de base et surtout de réduire le temps d'apprentissage. Nous avons montré ainsi que la détection de la rétinopathie diabétique peut se faire avec des réseaux de neurones,sans avoir recours aux méthodes traditionnelles qui sont basées sur la détection des lésions de la maladie.

Les résultats obtenus avec le modèle utilisé sont très satisfaisants et témoignent donc l'efficacité de ces réseaux dans la détection de la maladie,En effet on a pu atteindre des moyennes de spécificité et de sensibilité respectivement égales à 91.4% et 90.2%.

Cependant les résultats de ce modeste travail constituent la première pierre d'un travail à poursuivre et à améliorer dans une étude beaucoup plus approfondie qui pourra faire l'objet d'une thèse de doctorat.

Nous proposons également quelques perspectives autour de la détection de la RD avec les réseaux de neurones convolutionnels dans la liste ci-dessous :

- Adopter l'une des technique d'apprentissage ensembliste,qui consiste à utiliser plusieurs modèles de détection simultanément, ces modèles se complémentent pour constituer un ensemble très consistant et efficace dans la détection de la RD.
- Possibilité d'introduire des données supplémentaires dans la classification,telles que : l'age,sex,polarité de l'oeil(droit,gauche),des métadonnées extraites de l'image elle-même (moyenne,variance...etc.).
- Etudier la possibilité de déployer le système de détection dans les Rétinographes, ainsi le dépistage peut se faire en temps réel.
- Implémentation d'un algorithme de ségmentation des lésions de la RD sur l'image du fond d'oeil.

- Amélioration du rapport détaillé généré par la plateforme en marquant les zones contestant les lésions qui ont motivé le résultat de la classification.

Bibliographie

- [1] K. ALBERTI et P. ZIMMET, « Definition, diagnosis and classification of diabetes mellitus and its complications. Part 1 : diagnosis and classification of diabetes mellitus. Provisional report of a WHO Consultation », *Diabetic Medicine*, t. 15, 7, p. 539-553, 1998. DOI : 10.1002/(sici)1096-9136(199807)15:7<539::aid-dia668>3.0.co;2-s.
- [2] H. V. DOCTOR, S. NKHANA-SALIMU et M. ABDULSALAM-ANIBILOWO, « Health facility delivery in sub-Saharan Africa : successes, challenges, and implications for the 2030 development agenda », *BMC Public Health*, t. 18, 1, 2018. DOI : 10.1186/s12889-018-5695-z.
- [3] N. PEER, A.-P. KENGNE, A. A. MOTALA et J. C. MBANYA, « Diabetes in the Africa region : An update », *Diabetes Research and Clinical Practice*, t. 103, 2, p. 197-205, 2014. DOI : 10.1016/j.diabres.2013.11.006.
- [4] P. J. WATKINS, « ABC of diabetes : Retinopathy », *Bmj*, t. 326, 7395, p. 924-926, 2003. DOI : 10.1136/bmj.326.7395.924.
- [5] G. L. ONG, L. G. RIPLEY, R. S. NEWSOM, M. COOPER et A. G. CASSWELL, « Screening for sight-threatening diabetic retinopathy : comparison of fundus photography with automated color contrast threshold test », *American Journal of Ophthalmology*, t. 137, 3, p. 445-452, 2004. DOI : 10.1016/j.ajo.2003.10.021.
- [6] S. RESNIKOFF, W. FELCH, T.-M. GAUTHIER et B. SPIVEY, « The number of ophthalmologists in practice and training worldwide : a growing gap despite more than 200000 practitioners », *British Journal of Ophthalmology*, t. 96, 6, p. 783-787, 2012. DOI : 10.1136/bjophthalmol-2011-301378.
- [7] D. S. W. TING, P. H. YI et F. HUI, « Clinical Applicability of Deep Learning System in Detecting Tuberculosis with Chest Radiography », *Radiology*, t. 286, 2, p. 729-731, 2018. DOI : 10.1148/radiol.2017172407.
- [8] J. A. GOLDEN, « Deep Learning Algorithms for Detection of Lymph Node Metastases From Breast Cancer », *Jama*, t. 318, 22, p. 2184, déc. 2017. DOI : 10.1001/jama.2017.14580.
- [9] P. COPPO et A. VEYRADIER, « Microangiopathies thrombotiques : physiopathologie, diagnostic et traitement », *Réanimation*, t. 14, 7, p. 594-603, 2005. DOI : 10.1016/j.reaurg.2005.10.008.

- [10] E. ZANDER, « Maculopathy in patients with diabetes mellitus type 1 and type 2 : associations with risk factors », *British Journal of Ophthalmology*, t. 84, 8, p. 871-876, jan. 2000. DOI : 10.1136/bjo.84.8.871.
- [11] « Grading Diabetic Retinopathy from Stereoscopic Color Fundus Photographs—An Extension of the Modified Airlie House Classification », *Ophthalmology*, t. 98, 5, p. 786-806, 1991. DOI : 10.1016/s0161-6420(13)38012-9.
- [12] Z. OCKRIM et D. YORSTON, « Managing diabetic retinopathy », *Bmj*, t. 341, oct25 1, p. c5400-c5400, 2010. DOI : 10.1136/bmj.c5400.
- [13] W. S. MCCULLOCH et W. PITTS, « A logical calculus of the ideas immanent in nervous activity », *The Bulletin of Mathematical Biophysics*, t. 5, 4, p. 115-133, 1943. DOI : 10.1007/bf02478259.
- [14] D. O. HEBB, *The organization of behavior : a neuropsychological theory*. John Wiley Sons, 1949.
- [15] P. J. WERBOS, « Applications of advances in nonlinear sensitivity analysis. », *System modeling and optimization*, p. 762-770, 1982.
- [16] J. W. J. KIEFER, « Stochastic Estimation of the Maximum of a Regression Function », *The Annals of Mathematical Statistics*, t. 23, p. 462-466, 1952.
- [17] Y. LECUN, L. BOTTOU, Y. BENGIO et P. HAFFNER, « Gradient-based learning applied to document recognition », *Proceedings of the IEEE*, t. 86, 11, p. 2278-2324, 1998. DOI : 10.1109/5.726791.
- [18] O. CHUTATAPE, L. ZHENG et S. KRISHNAN, « Retinal blood vessel detection and tracking by matched Gaussian and Kalman filters », *Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. Vol.20 Biomedical Engineering Towards the Year 2000 and Beyond (Cat. No.98CH36286)*, DOI : 10.1109/tembs.1998.746160.
- [19] D. VALLABHA, R. DORAIRAJ, K. NAMUDURI et H. THOMPSON, « Automated detection and classification of vascular abnormalities in diabetic retinopathy », *Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers, 2004.*, DOI : 10.1109/acssc.2004.1399432.
- [20] A. FLEMING, S. PHILIP, K. GOATMAN, J. OLSON et P. SHARP, « Automated microaneurysm detection using local contrast normalization and local vessel detection », *IEEE Transactions on Medical Imaging*, t. 25, 9, p. 1223-1232, 2006. DOI : 10.1109/tmi.2006.879953.
- [21] P. F. SHARP, « Automated assessment of retinal image field of view », *Med.Image Understand. Anal*, p. 129-132, sept. 2004.

- [22] A. SOPHARAK, B. UYYANONVARA, S. BARMAN et T. H. WILLIAMSON, « Automatic detection of diabetic retinopathy exudates from non-dilated retinal images using mathematical morphology methods », *Computerized Medical Imaging and Graphics*, t. 32, 8, p. 720-727, 2008. DOI : 10.1016/j.compmedimag.2008.08.009.
- [23] N. OTSU, « A Threshold Selection Method from Gray-Level Histograms », *IEEE Transactions on Systems, Man, and Cybernetics*, t. 9, 1, p. 62-66, 1979. DOI : 10.1109/tsmc.1979.4310076.
- [24] G. G. GARDNER, D. KEATING, T. H. WILLIAMSON et A. T. ELLIOTT, « Automatic detection of diabetic retinopathy using an artificial neural network : a screening tool. », *British Journal of Ophthalmology*, t. 80, 11, p. 940-944, jan. 1996. DOI : 10.1136/bjo.80.11.940.
- [25] J. NAYAK, P. S. BHAT, R. A. U, C. M. LIM et M. KAGATHI, « Automated Identification of Diabetic Retinopathy Stages Using Digital Fundus Images », *Journal of Medical Systems*, t. 32, 2, p. 107-115, 2007. DOI : 10.1007/s10916-007-9113-9.
- [26] H. PRATT, F. COENEN, D. M. BROADBENT, S. P. HARDING et Y. ZHENG, « Convolutional Neural Networks for Diabetic Retinopathy », *Procedia Computer Science*, t. 90, p. 200-205, 2016. DOI : 10.1016/j.procs.2016.07.014.
- [27] « Transformed representations for convolutional neural networks in diabetic retinopathy screening. », 2014.
- [28] X. LI, T. PANG, B. XIONG, W. LIU, P. LIANG et T. WANG, « Convolutional neural networks based transfer learning for diabetic retinopathy fundus image classification », *2017 10th International Congress on Image and Signal Processing, Bio-Medical Engineering and Informatics (CISP-BMEI)*, 2017. DOI : 10.1109/cisp-bmei.2017.8301998.
- [29] M. STONE, « Cross-Validatory Choice and Assessment of Statistical Predictions », *Journal of the Royal Statistical Society : Series B (Methodological)*, t. 36, 2, p. 111-133, 1974. DOI : 10.1111/j.2517-6161.1974.tb00994.x.
- [30] L. GANG, O. CHUTATAPE et S. KRISHNAN, « Detection and measurement of retinal vessels in fundus images using amplitude modified second-order Gaussian filter », *IEEE Transactions on Biomedical Engineering*, t. 49, 2, p. 168-172, 2002. DOI : 10.1109/10.979356.
- [31] N. V. CHAWLA, K. W. BOWYER, L. O. HALL et W. P. KEGELMEYER, « SMOTE : Synthetic Minority Over-sampling Technique », *Journal of Artificial Intelligence Research*, t. 16, p. 321-357, jan. 2002. DOI : 10.1613/jair.953.
- [32] « Unsupervised representation learning with deep convolutional generative adversarial networks. », 2015.

- [33] A. M. REZA, « Realization of the Contrast Limited Adaptive Histogram Equalization (CLAHE) for Real-Time Image Enhancement », *The Journal of VLSI Signal Processing-Systems for Signal, Image, and Video Technology*, t. 38, 1, p. 35-44, 2004. DOI : 10.1023/b:vlsi.0000028532.53893.82.
- [34] K. HE, X. ZHANG, S. REN et J. SUN, « Deep Residual Learning for Image Recognition », *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. DOI : 10.1109/cvpr.2016.90.
- [35] C. SZEGEDY, S. IOFFE, V. VANHOUCKE et A. ALEMI, « Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning », *AAAI Conference on Artificial Intelligence*, fév. 2016.
- [36] C. IOFFE Sergey et SZEGEDY, « Batch normalization : Accelerating deep network training by reducing internal covariate shift. », 2015.
- [37] F. LOSHCHELOV Ilya et Hutter, « Decoupled weight decay regularization », 2017.
- [38] T. SPENCER, J. A. OLSON, K. C. MCHARDY, P. F. SHARP et J. V. FORRESTER, « An Image-Processing Strategy for the Segmentation and Quantification of Microaneurysms in Fluorescein Angiograms of the Ocular Fundus », *Computers and Biomedical Research*, t. 29, 4, p. 284-302, 1996. DOI : 10.1006/cbmr.1996.0021.
- [39] O. FAUST, R. A. U., E. Y. K. NG, K.-H. NG et J. S. SURI, « Algorithms for the Automated Detection of Diabetic Retinopathy Using Digital Fundus Images : A Review », *Journal of Medical Systems*, t. 36, 1, p. 145-157, juin 2010. DOI : 10.1007/s10916-010-9454-7.
- [40] U. R. ACHARYA, C. M. LIM, E. Y. K. NG, C. CHEE et T. TAMURA, « Computer-based detection of diabetes retinopathy stages using digital fundus images », *Proceedings of the Institution of Mechanical Engineers, Part H : Journal of Engineering in Medicine*, t. 223, 5, p. 545-553, 2009. DOI : 10.1243/09544119jeim486.
- [41] F. ROSENBLATT, « The perceptron : A probabilistic model for information storage and organization in the brain. », *Psychological Review*, t. 65, 6, p. 386-408, 1958. DOI : 10.1037/h0042519.

Acronymes

AMIR Anomalies Microvasculaires IntraRétiniennes. iii, 4, 6, 7

API Application Programming Interface. 80

BDD Base De Donées. 84

CLAHE Contrast Limited Adaptive Histogram Equalization. 87

CNN Convolutional Neural Network. iv, 7, 19–21, 23, 41

DL Deep Learning. 9, 12

EyePACS Eye Picture Archive Communication System. 62

GAN Generative adversarial networks. 63

GPU Graphics Processing Unit. 41

HSL Hue Saturation Lightness. 34

HTTP Hypertext Transfer Protocol. 83, 84

IA Intelligence Artificielle. 9, 10

IRM Imagerie par Résonance Magnétique. 61, 62

JPEG Joint Photographic Experts Group. 79

ML Machine Learning. 81

NPDR Non-Proliferative Diabetic Retinopathy. 27

ORM Object Relational Mapping. 84

RD Rétinopathie Diabétique. i, 3, 105

REST Representational state transfer. 85

RGB Red Green Blue. 34

SVM Support Vector Machine. 39, 40

URL Uniform Resource Locator. 84

Glossaire

Angiographie une technique d'imagerie médicale portant sur les vaisseaux sanguins qui ne sont pas visibles sur des radiographies standards. Elle impose l'injection d'un produit de contraste lors d'une imagerie par rayons X. L'angiographie est un examen invasif. iv, 4, 28, 29

Angiopathie Une angiopathie est toute affection concernant les vaisseaux sanguins : artères, veines et capillaires, voire vaisseaux lymphatiques. L'angiopathie la plus connue et la plus répandue est l'angiopathie diabétique, une complication fréquente du diabète chronique. 3

Macula La macula, ou tache jaune, est la zone de la rétine caractérisée par une concentration maximale de cônes. Située au fond de l'œil, dans l'axe de la pupille, la macula a un diamètre d'environ 5,5 mm. 3, 7

rétine La rétine est une fine membrane tapissant le globe oculaire. Par analogie avec un appareil photo, elle est en quelque sorte la « pellicule » de l'œil, chargée de capter les rayons lumineux pour les transmettre au système nerveux central. 3, 6, 7, 37, 40

Rétinographe Le Rétinographe est l'appareil qui sert à photographier le fond d'œil, soit tout ce qui est visible en arrière de l'iris et du cristallin. On peut y observer la rétine, la papille optique, la macula, l'ensemble portant le nom de pôle postérieur. 7, 105