

الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة أبي بكر بلقايد - تلمسان

Université Aboubakr Belkaïd – Tlemcen –

Faculté de TECHNOLOGIE



THESE

Présentée pour l'obtention du **grade de DOCTORAT 3^{ème} Cycle**

En : Génie biomédical

Spécialité : Informatique biomédical

Par : Khouani Amin

Sujet

Vers une reconnaissance intelligente des données biomédicales par apprentissage en profondeur

Soutenue publiquement, le 14 /09 /2021 , devant le jury composé de :

M. Mahammed MESSADI	Professeur	Univ. Tlemcen	Président
M. Med Amine CHIKH	Professeur	Univ. Tlemcen	Directeur de thèse
M. Mostafa EL HABIB DAHO	MCA	Univ. Tlemcen	Co- Directeur de thèse
M. Abdellatif RAHMOUN	Professeur	Univ. Sidi BelAbbes	Examineur
M. Med El Amine LAZOUNI	MCA	Univ. Tlemcen	Examineur
M. Sidi Ahmed MAHMOUDI	Professeur	Univ. Mons	Invité
M. Brahim BENZINEB	Professeur	CHU. Tlemcen	Invité

Remerciement

Il me sera difficile de remercier toutes les personnes qui ont su m'apporter leur soutien durant ces années. Je commencerai évidemment par remercier mon directeur de thèse, le Professeur Mohamed Amine CHIKH, pour la confiance qu'il m'a attribué tout en m'encourageant à aller de l'avant, en me laissant autonome, avec des échanges toujours aussi riches et divers.

Je tiens à exprimer ma gratitude à mon co-directeur de thèse Monsieur Mostafa EL HABIB DAHO, tout d'abord, de m'avoir introduit à ce domaine et de m'avoir guider et orienter tout au long de mes travaux de thèse, je vous exprime mes sincères remerciements.

J'adresse mes sincères remerciements à Monsieur Mohammed MESSADI, professeur de l'université de Tlemcen, qui m'a fait l'honneur de présider le jury de thèse.

J'exprime ma profonde reconnaissance à M. Abdellatif RAHMOUN, Professeur à l'école supérieur d'informatique ESI de Sidi BelAbbes, M. Mohammed El Amine LAZOUNI Maître de conférence à l'université de Tlemcen, pour l'intérêt qu'ils ont bien voulu porter à ce travail de thèse en acceptant de l'examiner et d'en être rapporteurs.

Je tiens à remercier, M. Sidi Ahmed MAHMOUDI, Professeur à l'université de Mons, M. Brahim Benzineb Maître de Conférence à l'hôpital universitaire de Tlemcen, de m'avoir accueilli et de m'avoir bien orienté dans mes travaux de thèse, tout en m'en-

courageant, je vous exprime ma gratitude.

J'adresse mes remerciements les plus sincères à tous les enseignants du département de génie biomédical et spécialement ceux qui ont contribué à mon cursus, aux membres du laboratoire Génie Biomédicale et spécialement les membres de l'équipe CREDOM.

A ma famille, je vous remercie tous, du plus grand au plus petit. Tout d'abord, j'adresse mes plus affectueux remerciements à mes parents, Tahar et Fatiha depuis ma naissance à ce jour vous êtes toujours à mes côtés, vos précieuses valeurs et votre amour propre m'ont toujours guidés et me guideraient pour toujours. A ma femme Ihsen et ma fille Tesnim, votre soutien et votre présence m'ont encouragé à continuer, je vous remercie du fond de mon cœur. Et à l'occasion un grand merci pour mon frère Youcef et ma sœur Siham, vous êtes toujours présents pour me soutenir. Un clin d'œil à mes tantes, mes oncles et toute la grande famille.

Résumé

Dans l'analyse microscopique en hématologie, la morphologie et la proportion des cellules dans les lames peuvent avoir une signification médicale dans le diagnostic de plusieurs maladies cancéreuses telles que le myélome, les leucémies etc. Cependant, la reconnaissance manuelle reste une tâche fastidieuse qui nécessite beaucoup de concentration et d'expérience pour différencier et compter les différents éléments cellulaires. Par conséquent, un système de reconnaissance automatique est nécessaire afin d'aider le praticien et rendre la lecture des lames rapide et précise. Les systèmes automatiques standard sont basés principalement sur la segmentation et la classification des globules blancs.

Dans cette thèse, nous proposons deux contributions principales, dans la première, nous nous sommes intéressés à la détection et la segmentation des globules blancs. Nous avons formé un modèle profond pour qu'il soit capable de détecter et de segmenter automatiquement tous les noyaux et les cytoplasmes. Nous avons appliqué plusieurs techniques de régularisations et quelques améliorations sur les prédictions du modèle en ajoutant des corrections et des prédictions combinées, ainsi qu'un nouvel algorithme qui utilise une coopération entre les sorties du modèle et les informations spatiales de l'image initiale pour une meilleure qualité de segmentation. Les résultats de ce premier travail sont très prometteurs et apportent une amélioration signi-

ficative dans la précision et le temps de calcul.

La deuxième contribution concerne la classification des sous-types des globules blancs. Nous avons proposé une nouvelle méthode avec cinq architectures profondes pour l'extraction des caractéristiques à partir des images. Les résultats obtenus ont montré l'efficacité de l'approche proposée qui utilise une prédiction combinée entre les architectures par rapport à chaque modèle individuellement.

Mots-clés : Image microscopique, Apprentissage profond, Segmentation d'image, Classification

Table des matières

Remerciement	1
Résumé	3
1 Introduction générale	12
1.1 Contexte et Motivation	12
1.2 Contribution	14
1.3 Organisation de la thèse	16
2 L'approche standard de diagnostic en Hématologie	19
2.1 Introduction	19
2.2 préparation des frottis	21
2.3 Coloration MGG (May-Grünwald Geimsa)	21
2.4 Composition du sang et ses multiples fonctions	24
2.4.1 Les globules rouges	24
2.4.2 Les globules blancs	26
2.4.3 Les plaquettes	30
2.5 Système d'aide au diagnostic	30
2.5.1 Acquisition des images cytologiques	30
2.5.2 Segmentation automatique des globules blancs	31
2.5.3 Classification des sous-types des leuco- cytes dans les images microscopiques	32
2.6 Conclusion	32

4.4.1	Environnement de développement	84
4.4.2	Base de données des frottis sanguins et médullaires	86
4.4.3	Préparation de jeux de données	87
4.4.4	Pré-traitement des données	87
4.4.5	Modèle d'apprentissage	90
4.4.6	Post-traitement	92
4.5	Résultats et discussion	94
4.6	Conclusion	100
5	Méthode réalisée pour la classification des sous-types des leucocytes	102
5.1	Introduction	102
5.2	État de l'art sur la classification des cellules	104
5.3	Approche proposée pour la classification	106
5.4	Base de données des leucocytes	106
5.5	Modèle d'apprentissage et paramétrage	108
5.5.1	Méthodes d'optimisation	108
5.5.2	Modèle d'extraction des caractéristiques	109
5.5.3	Régularisation	112
5.6	Résultats et discussion	114
5.7	Conclusion	117
6	Conclusion générale	119
	Bibliographie	123

Table des figures

2.1	Système d'aide au diagnostic pour la reconnaissance des images microscopiques en utilisant les méthodes d'apprentissage artificiel	20
2.2	Préparation du frottis sanguin (1)	22
2.3	Globule rouge	25
2.4	Observation des globules blancs avec un microscope optique (colorés au MGG). (a) basophiles (b) éosinophile (c) neutrophile (d) Lymphocytes (e) Monocytes.	28
3.1	Représentation graphique de la relation entre l'intelligence artificiel, l'apprentissage automatique (Machine Learning) et apprentissage profond (Deep Learning) (2)	35
3.2	Neurone biologique	41
3.3	Architecture du perceptron	42
3.4	Les fonctions logiques ET, OU et XOR. On voit que l'ensemble de données est séparé par une seule ligne dans le 'ET' et le 'OU'. Dans le 'XOR' il faut deux lignes pour séparer les données. Elles ne sont pas linéairement séparables.	44
3.5	Courbe de la fonction de Heaviside	45
3.6	Les tracés des fonctions d'activation les plus populaires dans la littérature.	46

3.7 Réseau de neurones avec L couche, les poids du neurone i de la couche 'L' issue du j^{ime} neurone de la $(l - 1)^{ime}$ couche précédente sont montrés en rouge, bleu et vert.	48
3.8 Processus de convolution extraite de (3). Le noyau ici est de taille 3x3, l'entrée en bleu est de taille 5x5 avec un pas de 1 et un remplissage de 1.	54
3.9 Principe de fonctionnement de ReLU	56
3.10 Exemple de pooling avec les différentes stratégies (maximum, moyen et somme). La taille des fenêtres ainsi que le pas utilisé dans cet exemple sont de 2x2.	57
3.11 Architecture d'un réseau de neurones convolutif complète.	59
3.12 Principe de fonctionnement de l'algorithme de descente de gradient pour trouver un minimum d'une fonction f par sa dérivée (2)	61
3.13 Architecture du réseau LeNet5 (4)	66
3.14 Architecture du réseau de neurone convolutif Alex-Net (5)	67
3.15 Module inception extraite de (6). Il est composé de quatre chemins avec des convolutions qui ont des noyaux de taille différentes.	68
3.16 Architecture complète du réseau GoogLeNet (6)	68
3.17 Architecture VGG16 extraite de (7)	69
3.18 Principe d'une connexion résiduelle extraite de (8). L'entrée X est ajoutée au résultat $F(x)$	70

4.1	Algorithmes de segmentations basées sur DL pour les images 2D depuis 2014 jusqu'au 2020 (9)	76
4.2	Schéma bloc de l'approche proposée pour la segmentation des instances dans les images cytologiques (10)	83
4.3	Exemple d'annotation : images originales et annotations correspondantes.	86
4.4	Exemple de transformation sur la couleur. a) Image originale, b) Image avec décalages du canal, c) Image au niveau de gris, d) Image contrastée, e) Image avec Luminosité, f) Image saturé	90
4.5	a) Image originale et son masque correspondant. b), c), d), e), f) ,g) des cas de transformations en parallèle	90
4.6	Résultats de la détection, la segmentation, et la classification automatique par le modèle original et l'approche proposé (10)	97
5.1	Modèle proposé pour la classification des cellules	107
5.2	Distribution des images de la base de données avant et après l'augmentation	108
5.3	Résumé du modèle proposé	111
5.4	Courbe de précision d'apprentissage et de validation de notre modèle CNN proposé (CytoNet).	114
5.5	Matrice de confusion de différent modèles	116

Liste des tableaux

4.1	Comparaison entre le Mask RCNN et notre modèle.	94
4.2	Comparaison des performances de la segmenta- tion entre l'approche proposée et l'état de l'art sur les même données	95
5.1	Précision, Rappel et F1-score sur les données de test	117
5.2	Comparaison avec d'autres méthodes sur la même base de données	118

1 Introduction générale

1.1 Contexte et Motivation

La cytologie est une discipline scientifique qui permet de comprendre la structure et les fonctionnalités des divers éléments cellulaires normales ou pathologiques (cytopathologie). Elle occupe une place très importante dans le diagnostic médical de plusieurs maladies, et notamment des cancers. Le processus de diagnostic des cancers commence généralement par une visite médicale chez un médecin généraliste qui dirige le patient vers un spécialiste. Lorsque ce dernier remarque des défaillances sur des cellules après une analyse sanguine, il prescrit une variété de tests pour analyser les cellules du sang et de la moelle osseuse. Tout d'abord, une étude quantitative se fait à l'aide de l'hémogramme, suivi par un examen qualitatif complémentaire via un frottis sanguin. S'il découvre une anomalie quantitative ou qualitative, il oriente le patient pour la réalisation d'un frottis médullaire (myélogramme).

Dans une telle analyse, un prélèvement d'échantillons doit être réalisé pour préparer les frottis sanguins ou médullaires dans une lame de verre et l'examiner au microscope optique. La lecture et l'analyse de ces lames se fait manuellement par les anatomopathologistes pour identifier les maladies par une étude qualitative et quantitative des cellules sanguines.

Ces tâches manuelles sont très longues, fastidieuses et demandent

beaucoup d'expérience et de concentrations en plus qu'elles sont sujets aux erreurs humaines. Pour ces raisons, l'automatisation de cette phase de diagnostic à l'aide d'un système de reconnaissance des cellules peut aider énormément le praticien dans la lecture des lames pour effectuer un diagnostic rapide et fiable. Dans ce projet de thèse, nous proposons une solution à travers un système informatique basé sur des techniques de l'intelligence artificielle (IA) et de traitement d'images pour assister le praticien et simplifier la tâche de reconnaissance des cellules ainsi que leurs catégorisations en sous-types. Ce travail nécessite, une très bonne compréhension du domaine avec une connaissance a priori sur les techniques de machine learning (ML) et le traitement d'images.

Plusieurs travaux dans la littérature ont été présentés dans ce domaine. La plupart d'entre eux utilisent des méthodes classiques qui reposent sur une étape de segmentation et de caractérisation pour faire la classification. Ces méthodes ont mené des recherches intensives avec des résultats remarquables. Cependant, elles sont limitées par leurs capacités de représentation et leur performance dépend principalement sur l'étape de caractérisation. En outre, ces méthodes séparent la partie de segmentation et d'extraction de caractéristique de la phase de classification, ce qui est un désavantage en matière de coût.

En revanche, les chercheurs en intelligence artificielle ont commencé ces dernières années à utiliser l'apprentissage profond, aussi appelé deep learning (DL), dans les tâches de reconnaissances d'images. Depuis leurs succès en 2012 dans le challenge de ImageNet, les réseaux de neurones profonds ont montré une

grande efficacité dans le domaine biomédical. Leur principe est inspiré par l'organisation du cortex visuel humain, Ils utilisent un réseau de neurones convolutifs ([11]) pour extraire les différentes caractéristiques de l'image d'une manière automatique à plusieurs niveaux d'abstraction par un processus d'apprentissage. Les travaux présentés dans cette thèse sont le fruit d'une collaboration (projet PRFU A13N01UN130120180005, 2018-2021) entre des membres du laboratoire de Génie Biomédical de l'université de Tlemcen et des médecins du service d'hématologie du CHU de Tlemcen.

1.2 Contribution

Dans ce projet de thèse, nous avons proposé deux contributions principales qui se basent sur les techniques d'apprentissage profond pour l'aide au diagnostic médical dans le domaine de la cytologie.

La première contribution concerne la détection et la segmentation des globules blancs en tant que noyaux et cytoplasmes dans une image microscopique. Nous nous intéressons beaucoup plus aux cytoplasmes et aux noyaux car ils sont cliniquement plus importants que les globules rouges. L'objectif principal est d'entraîner un modèle de réseau de neurones profond sur notre ensemble de données cytologiques pour qu'il soit capable de détecter correctement les globules blancs et extraire ses différents composants à savoir le cytoplasme et les noyaux ([10]).

Les modèles neuronaux profonds ont besoin d'une grande masse

de données pour effectuer leurs apprentissages. Malheureusement, nous n'avons pas toujours un ensemble de données conséquent pour un apprentissage optimal de ces modèles. Pour traiter ce problème d'insuffisance des données, nous avons utilisé une technique appelée augmentation de données ou Data Augmentation. Cette méthode consiste à créer de nouvelles images en appliquant plusieurs transformations sur la forme et la couleur sur l'ensemble de données originales.

Nous avons testé plusieurs architectures profondes afin de choisir le meilleur réseau avec ses hyperparamètres optimales pour notre base de données. Nous avons également appliqué plusieurs techniques de régularisation pour une meilleure généralisation et pour éviter le problème de sur-apprentissage.

L'idée principale dans cette contribution réside dans la combinaison coopérative entre les sorties du modèle. nous avons proposé un nouvel algorithme qui combine les meilleurs résultats obtenus avec les informations spatiales de l'image d'entrée pour améliorer la qualité de segmentation. Les résultats obtenus par cette première approche sont très prometteurs et montrent une amélioration remarquable dans la précision et surtout dans le temps de calcul en comparant avec les autres méthodes de l'état de l'art.

La deuxième contribution de ce travail de thèse concerne la classification des sous-types des globules blancs dans les images microscopiques. Une nouvelle méthode basée sur les techniques d'apprentissage profond a été proposée pour identifier quatre types de leucocytes, à savoir : neutrophiles, éosinophiles monocytes, et lymphocytes.

Nous avons utilisé quatre architectures profondes connues sous les noms de VGG (12), ResNet(8), DensNet (13), et Inception-ResNet (14) ainsi qu'un nouveau modèle que nous avons développé pour apprendre les caractéristiques à partir des images cytologiques.

Nous avons appliqué plusieurs techniques de régularisation tels que la régularisation L2, l'augmentation des données, l'arrêt précoce, dropout et le transfert d'apprentissage afin de minimiser le temps d'apprentissage et de contrôler le problème de sur-apprentissage. Les résultats obtenus montrent l'efficacité de l'approche proposée qui utilise une prédiction combinée des différents modèles.

1.3 Organisation de la thèse

Ce manuscrit est organisé en quatre chapitres en plus de cette partie d'introduction générale et la conclusion générale. Dans le chapitre 2 nous présentons l'approche standard dans le diagnostic médical en hématologie. Nous commençons par les différentes étapes de préparation des frottis sanguins et médullaires pour l'analyse visuelle. On détaillera ensuite les composants cellulaires et ses multiples fonctions dans le corps humain. Dans ce même chapitre, nous introduisons une brève définition sur les différentes techniques utilisées dans le système d'automatisation.

Le chapitre 3 permet de se familiariser techniquement avec les concepts fondamentaux de l'intelligence artificielle et de l'apprentissage profond qui sont importants pour appréhender la suite de la thèse. Les préliminaires présentés dans la section 3.2 per-

mettent de comprendre l'apprentissage automatique et ses mécanismes avec ses trois types, supervisés, semis-supervisés et non supervisés. Nous présentons ensuite dans la section [3.3](#) les compositions classiques d'apprentissage profond, à savoir les perceptrons et les réseaux de neurones multicouches, ainsi que les techniques utilisées pour leur apprentissage. Ensuite, nous introduisons dans la section [3.4](#) les réseaux de neurones convolutifs, qui constitue les piliers de l'apprentissage profond. Nous présenterons les différents blocs et leurs paramètres ainsi que les techniques d'optimisation et de régularisation nécessaires pour construire une architecture complète. Nous terminerons ce chapitre par une revue sur quelques architectures populaires dans l'état de l'art.

Dans le chapitre [4](#), nous nous placerons dans le contexte applicatif où nous présenterons notre première contribution qui s'intéresse à la reconnaissance automatique des globules blancs dans les images microscopiques.

Tout d'abord, la section [4.2](#) présente une généralité sur les méthodes de segmentation et les différents types existents dans la littérature, nous entrerons plus en détail dans les méthodes de segmentation qui utilisent l'apprentissage profond avec leurs trois subdivisions (segmentation d'instance, segmentation sémantique, et segmentation panoptique), ainsi que leurs critères d'évaluation.

Ensuite dans la section [4.3](#) nous passerons en revue les méthodes de segmentation des images microscopiques existentes dans la littérature. La section [4.4](#) détaille notre approche proposée pour la détection et la segmentation avec les outils de programmation et la base de données utilisées dans nos expérimentations. Les

résultats et les comparaisons de cette approche ainsi que leurs discussions sont présentés à la fin de ce chapitre.

Dans le chapitre 5, nous présentons notre deuxième contribution qui se base sur les techniques d'apprentissage profond pour la classification des sous-types des globules blancs dans les images microscopiques, à savoir neutrophiles, éosinophiles, lymphocytes, et monocytes. Nous allons présenter une nouvelle méthode qui utilise une prédiction combinée entre plusieurs architectures profondes. À la fin de ce chapitre, nous présentons les résultats obtenus avec leurs discussions.

En conclusion, le chapitre 6 récapitule les problématiques étudiées et comment nos travaux permettent d'y répondre à travers les contributions apportées. Ensuite il présente les perspectives qui permettraient d'améliorer les performances pour la mise en œuvre dans un système d'aide au diagnostic médical.

2 L'approche standard de diagnostic en Hématologie

2.1 Introduction

Dans un processus du diagnostic en hématologie, l'analyse des frottis sanguins ou médullaires est une étape très importante qui permet au médecin de reconnaître les cellules anormales et de calculer les pourcentages des leucocytes. Cette étape a un intérêt primordial dans le diagnostic de plusieurs maladies cancéreuses comme la leucémie, le myélome, etc.

Cependant, l'analyse visuelle reste toujours une tâche fastidieuse, longue et sujette aux erreurs, en plus qu'elle demande beaucoup d'expérience et de concentration. Pour pallier ces tâches difficiles, nous proposons un système automatique pour la reconnaissance des cellules dans les images cytologiques afin d'aider les médecins dans le processus de diagnostic médical (figure 2.1).

Un système d'aide au diagnostic est constitué de plusieurs étapes, la première est de préparer et de colorer les frottis sanguins ou médullaires, ensuite vient l'étape d'acquisition des images en utilisant une caméra liée à un microscope optique. Avant de commencer le traitement, une connaissance a priori du domaine de la cytologie qui doit être combinée avec le domaine de traitement d'images et les techniques de machine learning pour pouvoir

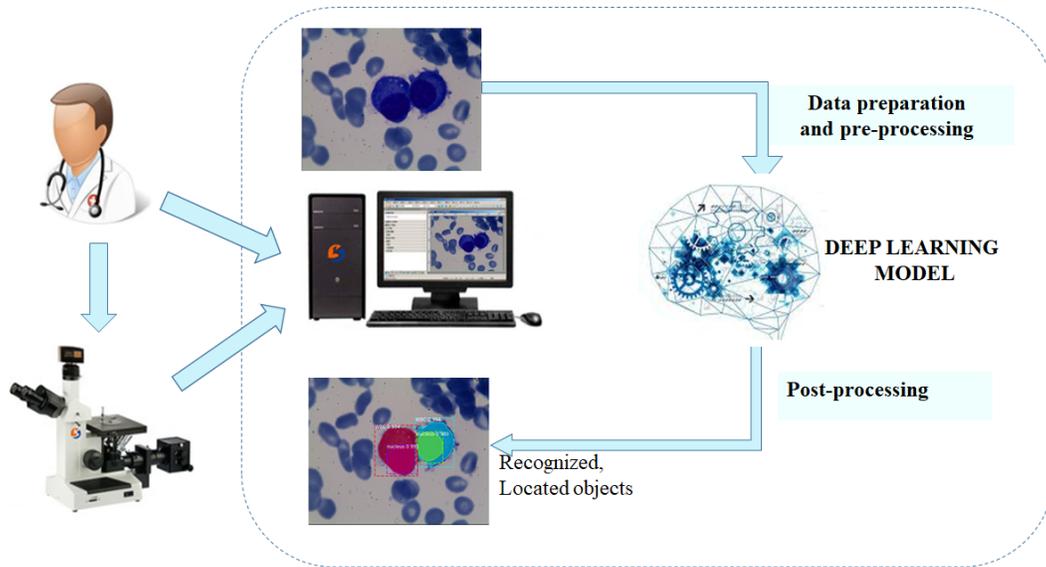


Figure 2.1 – Système d'aide au diagnostic pour la reconnaissance des images microscopiques en utilisant les méthodes d'apprentissage artificiel

donner une solution pratique et efficace. Dans la littérature, La segmentation des globules blancs et la classification de leurs sous-types dans les images cytologiques sont les principaux traitements dans la plupart des systèmes d'aide au diagnostic.

Dans ce qui suit, nous commençons par les différentes étapes pour préparer les frottis sanguins et médullaires pour l'analyse visuelle, par la suite nous détaillons la composition du sang et leurs morphologies ainsi que ses multiples fonctionnalités. Finalement, nous introduisons les différentes techniques standards utilisées dans le système d'automatisation.

2.2 préparation des frottis

La préparation des frottis sanguins ou médullaires est une étape importante pour analyser la morphologie des cellules et de les compter. L'idée principale est d'étaler une goutte de l'échantillon que nous voulons étudier (sang ou moelle osseuse) sur une lame de verre, pour obtenir une seule couche de cellules.

pour préparer un échantillon on doit suivre les étapes suivantes :

1. Placer une petite goutte de sang ou de moelle osseuse sur une extrémité d'une lame de verre (voir la figure [2.2](#)).
2. Placer le bord de la lamelle a côté de la goutte, puis mise en contact avec elle en formant un angle de 45° (figure [2.2](#)).
3. Pousser rapidement et régulièrement la goutte vers l'autre extrémité de la lame en gardant le même angle et sans trop presser, puis enlever doucement la lamelle et sécher rapidement le frottis à l'air libre loin des poussières.

Avant de passer à l'analyse des frottis via le microscope optique, une coloration au MGG (May-Grünwald Geimsa) doit être réalisée pour pouvoir distinguer les différents sous type de cellules.

2.3 Coloration MGG (May-Grünwald Geimsa)

La coloration MGG ou appelée également coloration de Papanheim, est une méthode utilisée en hématologie lors de pré-

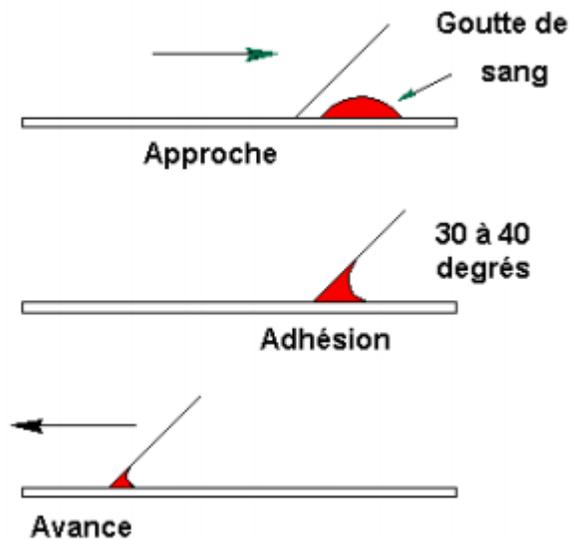


Figure 2.2 – Préparation du frottis sanguin (1)

parations des frottis pour distinguer les cellules du sang. Leur principe repose sur la combinaison de deux colorants :

- Le May-Grunwald, une solution neutre de l'éosine (colorant acide) et du bleu de méthylène (colorant basique).
- Le Giemsa, une solution neutre contenant également de l'éosine, et l'azur de méthylène.

Les deux colorants sont initialement inactifs dans une solution alcoolique, les sels précipitent (éosinate de méthylène et azur de méthylène) et se fixent sélectivement sur les constituants cellulaires après l'addition d'eau selon les règles suivantes :

- Les colorant basiques fixent électivement sur les constituants cellulaires acides, elles sont qualifiées de basophiles

(ADN, ARN).

- Les colorant acides fixent sélectivement sur Les constituants cellulaires basiques, elles sont qualifiées d'éosinophiles ou d'acidophiles (dans l'hémoglobine les hématies contiennent des protéines basiques, et dans Les granulations des éosinophiles).
- Les constituants cellulaires qui fixent dans Les deux types de colorant sont des neutrophiles.

Au terme de processus de coloration Les éléments cellulaires seront colorés de manière la suivante :

- Les granulations des polynucléaires basophiles sont bleu-noir (figure 2.4 (a)).
- Les granulations des polynucléaires éosinophiles sont rouge-orangé (figure 2.4 (b)).
- Les granulations des polynucléaires neutrophiles sont violets-lilas (figure 2.4 (c)).
- Les granulations des lymphocytes sont pourpres (figure 2.4 (d)).
- Les noyaux sont en bleu à violets-noir.
- Les hématies sont en beiges-rosé.

Dans ce qui suit, nous allons décrire la composition du sang et ses multiples fonctions.

2.4 Composition du sang et ses multiples fonctions

Le corps d'un adulte humain moyen comporte plus de 5 litres de sang. Ce dernier circule dans les artères et les veines pour délivrer l'oxygène et les matières organiques vers les cellules vivantes et récupérer leurs déchets. Il est responsable également sur le transport des cellules immunitaires (globules blancs) permettant au corps de lutter contre les infections. Et dans le cas d'un endommagement dans les vaisseaux sanguins, les plaquettes interviennent et s'accumulent pour former un bouchon afin d'éviter la perte de sang (coagulation).

Le sang donc est un liquide composé de plasma qui forme un pourcentage de 55% du sang, dans lequel circulent trois types de cellules : les globules rouges (44%), les globules blancs (ou leucocytes) et les plaquettes qui forment tous les deux 1% du sang. Toutes ces cellules commencent leur vie dans la moelle osseuse sous forme de cellules souches.

2.4.1 Les globules rouges

Avec une production de 2 à 3 millions de cellules par seconde, les globules rouges sont les cellules les plus courantes dans la circulation sanguine. Dans chaque millimètre cube du sang il existe plus de 5 millions de globules rouges responsables du transport de l'oxygène des poumons vers les cellules vivantes et récupérer

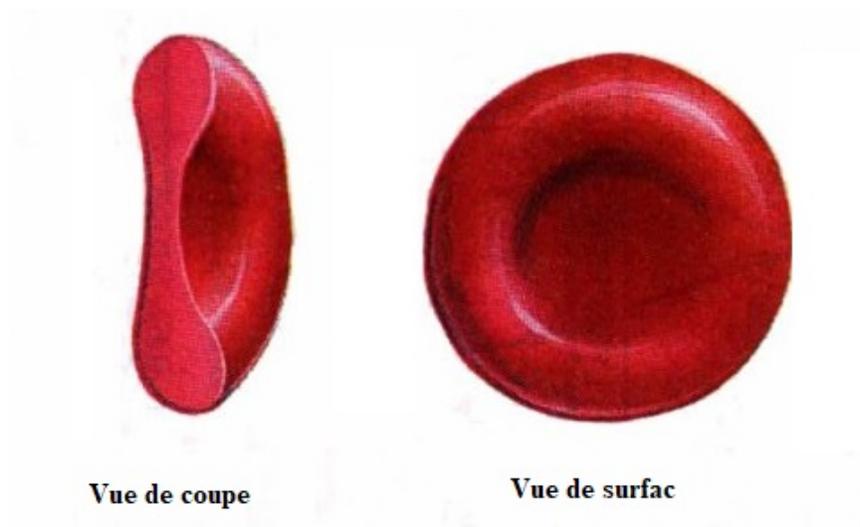


Figure 2.3 – Globule rouge

Le gaz carbonique lors du retour pour l'évacuer dans l'expiration. Leurs petits diamètres de 6 μm lui permettent de faufler dans les plus petits vaisseaux sanguins. La durée de vie moyenne de ces cellules est de 120 jours, après la fin de leur vie les macrophages interviennent pour les retirer de la circulation sanguine vers la rate et le foie.

Les globules rouges ou appelés aussi érythrocytes ou hématies ont une forme biconcave aplatie au centre sans noyau (voir figure 2.3) qui lui donne une surface plus grande pour transporter plus d'oxygène. Les molécules d'oxygène sont liées à une protéine appelée hémoglobine celle qui donne la couleur rouge aux sangs.

2.4.2 Les globules blancs

Les globules blancs ou leucocytes, sont les cellules responsables de la réponse immunitaire. Elles se déplacent dans le sang lorsqu'elles reçoivent un signal indiquant l'existence d'une attaque par un corps extérieur ou même si un corps interne est endommagé. Dans ce cas les globules blancs quittent leur circulation dans les vaisseaux sanguins vers la source du signal pour exercer leur fonction qui aide le corps humain dans le processus de guérison. Leur durée de vie est très courte, seulement de 2 à 3 jours en moyenne, leur densité est de 4000 à 10000 /mm³. Les leucocytes apparaissent dans le corps humain sous différents types selon les caractéristiques structurelles. Certaines ont un gros noyau rond et d'autres ont des noyaux multilobés, tandis que d'autres cellules connues sous le nom de granulocytes contiennent dans leur cytoplasme des sachets de granules. Les leucocytes donc se divisent selon la présence et l'absence de granules dans leurs cytoplasmes ainsi que la forme de leurs noyaux.

Parmi les cellules polynucléaires qui ont des granules dans leurs cytoplasmes avec un noyau multilobé, on peut distinguer les polynucléaires neutrophiles, les polynucléaires éosinophiles et les polynucléaires basophiles. Les cellules mononucléaires sont agranulaires avec un noyau massif, on peut distinguer parmi eux deux types totalement différents, les lymphocytes et les monocytes.

Le calcul de pourcentage de chaque type de cellules permet

de diagnostiquer et de surveiller plusieurs maladies telles que les infections, les tumeurs etc. Chez un adulte humain normal, on compte de 55 à 60 % de polynucléaires neutrophiles, de 1 à 3 % de polynucléaires éosinophiles, de 0 à 1 % de polynucléaires basophiles, de 25 à 33% de lymphocytes, et de 3 à 7 % de monocytes.

Polynucléaires neutrophiles

Les polynucléaires neutrophiles ou appelés également polymorphonucléaires (Figure 2.4) sont les plus nombreuses cellules parmi les leucocytes, ils sont caractérisés par leurs noyaux multilobés (de 2 à 5 lobes) avec une forme irrégulière. Leurs diamètres est de 12 μm . Le cytoplasme contient des granules minuscules rosés et qui sont à peine visibles par un microscope optique. Les neutrophiles interviennent dans le cas d'une infection bactérienne en dehors du sang. Elles phagocytent les bactéries très activement à l'endroit des réactions inflammatoires.

Polynucléaires éosinophiles

Les polynucléaires éosinophiles sont des leucocytes peu nombreux dans le sang (1 à 3%) avec une taille de 12 μm (Figure 2.4). Ils sont caractérisés par un noyau bilobé avec un cytoplasme rempli de granules acidophiles. Leur rôle est de détruire les complexes antigènes-anticorps.

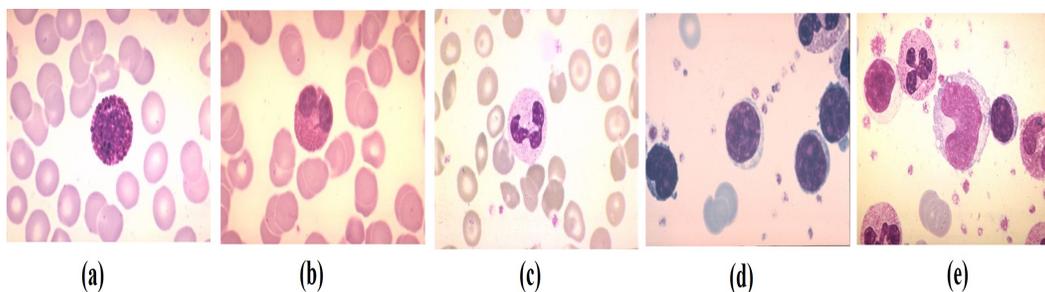


Figure 2.4 – Observation des globules blancs avec un microscope optique (colorés au MGG). (a) basophiles (b) éosinophile (c) neutrophile (d) Lymphocytes (e) Monocytes.

Polynucléaires basophiles

Les cellules polynucléaires basophiles sont les leucocytes les plus rares dans le sang (moins de 1%) avec un diamètre de 12 μm . Leurs noyaux comportent plusieurs lobes irréguliers et le cytoplasme contient de nombreuses granulations basophiles qui masquent le noyau (Figure 2.4). Les basophiles libèrent des produits comme les histamines et la sérotonine au cours des réactions allergiques.

Lymphocytes

Les lymphocytes sont des cellules mononucléaires avec une forme ronde. Ils contiennent un seul gros noyau circulaire très dense situé dans le centre et qui occupe presque tout l'espace de la cellule (Figure 2.4). Leurs diamètres se varient entre 7 et 12 μm , et selon lequel ils se divisent en trois classes : les petits lymphocytes, les moyens lymphocytes et les grands lymphocytes. Ils représentent de 20 à 30 % des leucocytes circulent dans le

sang, comme ils peuvent se trouver dans certains tissus typiquement dans les vaisseaux lymphatiques et les organes lymphoïdes.

Les lymphocytes sont les cellules principales dans la défense immunitaire contre les éléments considérés comme étrangers. Quand il y a une attaque extérieure, les lymphocytes reconnaissent les molécules étrangères et déclenchent une réponse immunitaire selon différents types. Certaines cellules peuvent se transformer en plasmocytes (ou lymphocyte B) qui produisent des protéines spécifiques appelées anticorps pour se fixer sur l'élément qui a déclenché la réponse immunitaire (immunité humorale).

D'autres cellules lymphocytes peuvent transformer en lymphocytes T, ou appelées également cellules T auxiliaires, ces cellules sécrètent des produits chimiques qui entraînent une prolifération cellulaire (immunité cellulaire). Un autre groupe de lymphocyte T appelé cellules T cytotoxiques attaque directement les cellules infectées par le virus.

Monocytes

Les monocytes sont des cellules monoculaires qui représentent 3 à 7 % des globules blancs. Leurs tailles moyennes de 17 μm est la plus grande parmi les cellules sanguines. Le cytoplasme est gris bleuté, un peu granuleux avec un noyau unique non polylobé central, en fer à cheval (Figure 2.4). Les monocytes fournissent une défense immédiate dans les premiers temps d'une réaction inflammatoire et dans l'immunité. Ils deviennent des macrophages

pour éliminer les substances étrangères.

2.4.3 Les plaquettes

Les plaquettes ou appelées aussi thrombocytes, sont des fragments cellulaires sans noyau de forme irrégulière proviennent de cellules souches de la moelle osseuses. Ces cellules souches se transforment en mégacaryocytes qui libèrent les plaquettes dans la circulation sanguine. Leur densité est de 200 000 à 300 000 par mm^3 de sang avec une durée de vie très courte, seulement 7 jours au moyen. Pendant cette période elles jouent un rôle important dans la coagulation. En cas d'une fissure dans les vaisseaux sanguins, les plaquettes interviennent en s'accrochant les unes aux autres pour former un bouchon qui permet de stopper les hémorragies.

2.5 Système d'aide au diagnostic

2.5.1 Acquisition des images cytologiques

Après la phase de préparation des frottis et de coloration, le processus d'aide au diagnostic se poursuit en plaçant la lame préparée sous un microscope optique, qui doit être relié avec une caméra et un ordinateur pour commencer l'acquisition et l'enregistrement des images.

Pour réaliser des captures de haute qualité, il faut d'abord

fixer un objectif de grossissement à x10 pour trouver une zone de lecture appropriée avec des cellules bien étalées non entassées. Puis augmenter le zoom jusqu'à x100 pour bien visualiser les cellules et réaliser des captures permettant de stocker les images dans le disque dur avant de les analyser.

2.5.2 Segmentation automatique des globules blancs

La segmentation est une étape très importante qui sert à identifier un groupe de pixels comme appartenant à un objet ou une région significative dans une image. Dans le domaine de cytologie, l'objectif est d'extraire les composants cellulaires afin d'acquérir des informations quantitatives et qualitatives nécessaires pour le diagnostic médical. De nombreux travaux existent dans la littérature qui ont été menés pour la segmentation des globules blancs dans les images microscopiques. La plupart d'entre eux sont basés sur des méthodes classiques dans lesquels une étape d'extraction des caractéristiques sur la couleur, textures et/ou forme est nécessaire pour réaliser la classification. Ces méthodes ont obtenu des résultats remarquables dans le domaine de la segmentation des images microscopiques, cependant, leurs inconvénients majeurs est le fait qu'ils séparent les parties du traitement à savoir la segmentation, l'extraction des caractéristiques et la classification, ce qui est considéré comme désavantage en matière de coût.

En revanche, il existe d'autres types de méthodes qui se basent sur les techniques d'apprentissage artificiel en particulier l'apprentissage profond pour la segmentation et la classification des

images microscopique sans la nécessité d'une étape de caractérisation. Ces méthodes utilisent une architecture profonde pour extraire les paramètres de l'image automatiquement à plusieurs niveaux d'abstraction.

Dans notre système d'aide au diagnostic, nous proposons une méthode intelligente basée sur le deep learning pour la reconnaissance automatique des leucocytes dans les images microscopiques.

2.5.3 Classification des sous-types des leucocytes dans les images microscopiques

Comme nous l'avons mentionné dans la section [2.4.2](#), les globules blancs peuvent être classés en cinq sous-types : éosinophiles, basophiles, neutrophiles, monocytes et lymphocytes. La deuxième partie de cette thèse se focalise sur la classification des sous-types des globules blancs dans les images microscopiques en utilisant les réseaux de neurones convolutifs qui sont des modèles d'apprentissage profond. Ces dernières années, les méthodes d'apprentissage profond ont montré une grande efficacité dans la classification des images surtout avec la disponibilité des données massives (Big data) et des ressources de calculs (carte graphiques - GPU-).

2.6 Conclusion

Dans le domaine de cytologie, l'analyse visuelle des frottis sanguins au microscope est une étape primordiale pour le diag-

nostic médical. Dans cette analyse le comptage des globules blancs et l'étude de leurs morphologies a une importance cruciale pour détecter le type de la maladie et son stade. Cependant, une telle analyse peut prendre beaucoup de temps et nécessite des efforts importants et une bonne expérience pour établir un diagnostic fiable et précis. Notre objectif dans ce travail est de proposer une solution pratique qui consiste à automatiser ces tâches pour aider le médecin à effectuer un diagnostic rapide et précis à l'aide d'un système d'aide au diagnostic. Deux approches qui se basent sur les techniques de machine learning et l'apprentissage profond ont été proposées pour la détection et la segmentation des globules blancs et pour la classification de leurs sous-types.

Dans le chapitre 3 nous présentons les différentes techniques de machine learning et d'apprentissage profond utilisées pour nos modèles.

3 Principes de Deep Learning

3.1 Introduction

Ces dernières années, l'apprentissage profond est de plus en plus utilisé dans de nombreux domaines tels que la reconnaissance d'images, de la parole, et le traitement de langage naturel. L'apprentissage profond (Deep Learning en anglais) fait partie du domaine de l'apprentissage automatique, qui est à son tour un sous-domaine de l'intelligence artificielle, la figure [3.1](#) montre la représentation graphique de cette relation. Le domaine de l'intelligence artificielle représente l'ensemble des algorithmes et des techniques permettant aux machines d'effectuer un raisonnement automatique, et l'apprentissage automatique se spécialise principalement dans la reconnaissance automatique des formes et l'apprentissage en utilisant des données.

L'algorithme le plus populaire dans le domaine de l'apprentissage automatique est celui des réseaux de neurones artificiels (RNA) qui utilise des données pour l'apprentissage. Leur principe de fonctionnement est inspiré par le comportement du cerveau humain. De même, l'apprentissage profond fait partie de la famille des réseaux de neurones artificiels. Il existe depuis des dizaines d'années sous différentes appellations, mais le concept de Deep Learning émerge au début des années 2010 avec l'arrivée des machines de calcul plus puissantes et la disponibilité des grandes bases de données.

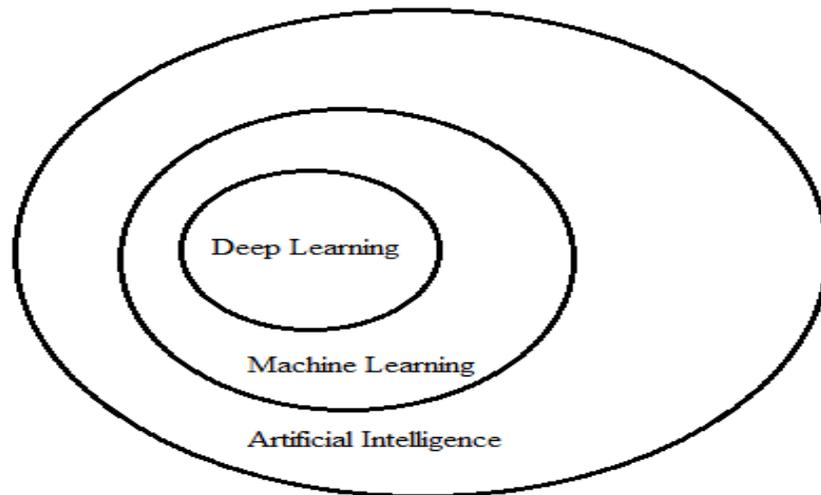


Figure 3.1 – Représentation graphique de la relation entre l'intelligence artificiel, l'apprentissage automatique (Machine Learning) et apprentissage profond (Deep Learning) (2)

Dans ce chapitre, nous verrons les concepts fondamentaux de l'apprentissage profond, nous commençant par une présentation des différentes notions liées à l'apprentissage automatique, avec ses trois formes, supervisés, semi supervisés et non supervisés. Nous verrons par la suite les structures classiques d'apprentissage profond tel que les perceptrons et les réseaux de neurones artificiels, ainsi que les techniques utilisées pour leurs apprentissages. Enfin, nous entrerons dans les réseaux de neurones convolutifs qui constituent la colonne vertébrale de l'apprentissage profond. Nous présenterons les différents blocs et leurs paramètres ainsi que les méthodes d'optimisation et de régularisations, et nous passerons une revue sur quelques modèles populaires dans la littérature pour conclure ce chapitre.

3.2 Apprentissage Automatique

L'apprentissage automatique (Machine Learning) est une branche de l'intelligence artificielle qui permet à des machines d'apprendre sans avoir été explicitement programmées à cet effet (15). Les algorithmes d'apprentissage automatique interviennent dans les situations très complexes qui ne peuvent pas être résolues par des algorithmes basés sur des règles comme la reconnaissance d'image ou la classification des données. Concrètement, les méthodes d'apprentissage automatique consistent à imiter la façon d'apprentissage des êtres humains, ce processus peut être réalisé en utilisant des techniques inspirées par le comportement humain comme les réseaux de neurones artificiels qui ont essayé d'aller plus loin et imiter le cerveau humain.

L'apprentissage automatique permet de trouver automatiquement des modèles et effectuer des prédictions à partir d'un ensemble de données. Il existe trois types d'algorithmes d'apprentissage automatique qui sont répartis selon la disponibilité des étiquettes dans les bases de données : l'apprentissage supervisé, l'apprentissage non supervisé, et l'apprentissage semi-supervisé. Dans cette thèse, nous nous concentrons principalement sur le premier type qui est l'apprentissage supervisé dans le contexte de l'apprentissage profond. Toutefois, nous définirons les trois types principaux dans la section qui suit.

3.2.1 Les types d'apprentissage automatique

Apprentissage supervisé

Dans l'apprentissage supervisé, l'ensemble de données utilisé pour l'apprentissage est déjà étiqueté par un expert de domaine. Par conséquent, l'algorithme d'apprentissage automatique connaît ce qu'il doit chercher dans la base de données pour produire à la fin un modèle capable de prédire des étiquettes correctes sur des données non étiquetées (données de test). Donc, le processus d'apprentissage automatique supervisé passe par deux phases, la première est la phase d'apprentissage, elle permet d'entraîner le modèle en utilisant une partie des données étiquetées, ensuite la phase de test qui consiste à prédire les classes des nouvelles données à travers le modèle formé dans la première phase.

Il existe deux sous-types dans l'apprentissage supervisés selon le problème à résoudre : Les algorithmes de classification qui donnent des prédictions discrètes (non numériques), et les algorithmes de régression qui donnent des prédictions numériques (continues).

Apprentissage non supervisé

Contrairement à l'apprentissage supervisé, l'ensemble de données utilisées dans l'apprentissage non supervisé ne sont pas étiquetées. Les algorithmes d'apprentissage non supervisé parcourent l'ensemble des données sans aucune indication et essaient de découvrir le partitionnement des données selon leurs

paramètres disponibles pour les grouper en classes homogènes. On peut distinguer trois sous-types dans les modèles d'apprentissage non supervisés : Les algorithmes de Clustering qui se basent sur le calcul de similarité selon une fonction de distance entre deux exemples pour trouver des groupes d'exemples similaires. Le deuxième type est les algorithmes d'association qui permettent de trouver des liens entre les exemples. Le troisième type est les algorithmes de réduction dimensionnelle qui sont utilisés pour extraire ou choisir des caractéristiques dans la base de données.

Apprentissage semi-supervisé

Dans l'apprentissage semi-supervisé, l'ensemble de données contient peu d'exemples étiquetés et beaucoup d'exemples non-étiquetés. Les algorithmes d'apprentissage semi-supervisés utilisent la partie de la base de données étiquetées pour entraîner le modèle qui sera ensuite utilisé pour labéliser le reste de la base non-étiquetée. Ce processus peut s'itérer plusieurs fois afin de donner des prévisions plus précises. Généralement, ce type d'apprentissage est utile dans le domaine de reconnaissance d'image où il est difficile d'annoter chaque image individuellement dans une grande base de données, il suffit juste d'annoter une partie de la base de données et utiliser l'apprentissage semi-supervisé pour étiqueter le reste.

Actuellement, les deux derniers types d'apprentissage (non supervisé et semi-supervisé) sont des sujets de recherche très actifs dans le contexte de l'apprentissage profond. Néanmoins, il n'y a pas une direction claire sur les méthodes à utiliser jusqu'à maintenant. Pour cela, nous nous concentrons principalement dans la

suite de cette thèse sur le type d'apprentissage supervisé dans le cadre de l'apprentissage profond.

3.3 Les réseaux de neurones artificiels

Les réseaux de neurones artificiels sont les premières bases de l'apprentissage profond, leur principe de fonctionnement est inspiré par le comportement du cerveau humain. L'idée principale est de trouver automatiquement des relations entre une ou plusieurs entrées et une sortie connue dans un ensemble de données afin de construire des machines intelligentes. Dans ce qui suit, nous présenterons les aspects fondamentaux des réseaux de neurones artificiels et leurs relations avec les neurones biologiques. Nous présenterons également le modèle de perceptron et l'algorithme de rétro-propagation ainsi que leur utilisation dans l'apprentissage des réseaux de neurones multicouches. Cette partie permet de bien comprendre les réseaux de neurones afin de pouvoir passer aux réseaux de neurones profonds plus avancés.

3.3.1 Introduction aux réseaux de neurones artificiels

Dans notre vie quotidienne, nous rencontrons de nombreuses situations qui nécessitent de l'intelligence que ce soit au niveau des tâches liées à la reconnaissance ou à la prise de certaines décisions ou d'autres. En effet, les tâches que notre propre cerveau humain accomplit inconsciemment, sont très complexes et il est difficile de les reproduire dans un système qui tente à imiter

ces processus naturels. Par conséquent, la compréhension de ce qui se passe à l'intérieur de notre propre cerveau nous permet de formaliser cette intelligence et la transmettre à des programmes ou des machines qui performant des tâches similaires.

En effet, notre système d'apprentissage naturel contient un ensemble de neurones biologiques (cellules nerveuses) connectés entre eux et forme un réseau neuronal lié à notre système nerveux. Chaque neurone biologique (voir figure 3.2) est constitué d'un corps cellulaire avec un noyau et des entrées appelées dendrites. Les autres neurones génèrent un signal électrique et l'envoient vers les synapses de ces dendrites (leurs entrées). Si ce signal électrique est assez stimulant, les dendrites ont une action de le recevoir et le transmettre à travers l'axone. De même manière, chaque neurone artificiel reçoit des signaux en entrée, et effectue un calcul simple : si la somme pondérée de ces entrées atteint un certain seuil, le neurone est activé. Le seuil est décidé à l'aide d'une fonction qui s'appelle fonction d'activation et leur résultat va se transmettre aux autres neurones qui suivent. Certains neurones ont de grands poids qui amplifient le signal, cela signifie que la sortie de ces neurones est très importante dans le modèle final. Les autres qui ont des poids faibles qui diminuent le signal signifié que la sortie est moins importante dans le modèle final de classification. Donc un réseau de neurones artificiel est un système qui constitue d'un ensemble de neurones liés entre eux avec des poids de connexion modifiables par un processus d'apprentissage.

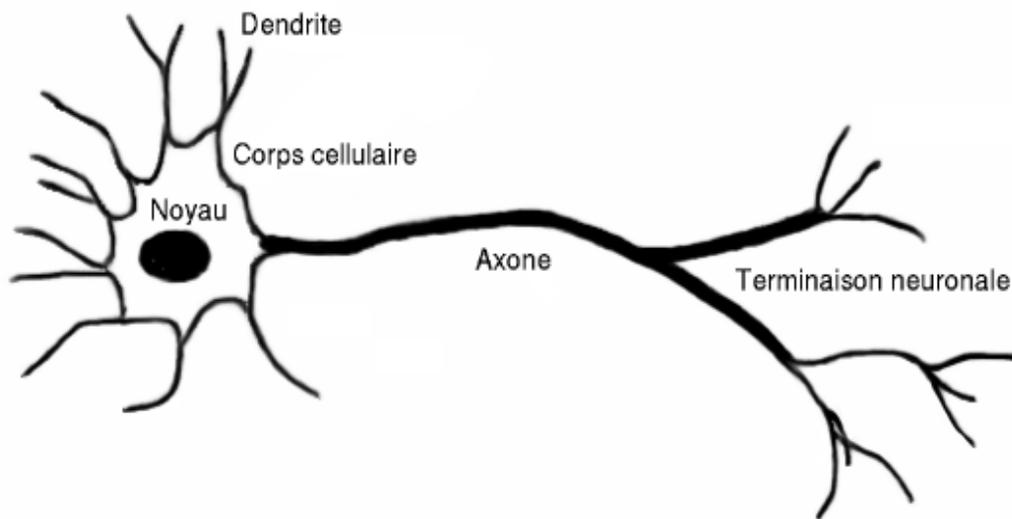


Figure 3.2 – Neurone biologique

3.3.2 Perceptrons

En 1958 Rosenblatt et al. (16) ont présenté le premier perceptron capable d'apprendre à partir d'un ensemble de données étiquetées (c-t-d. apprentissage supervisé). Le perceptron est un neurone formel avec une règle d'apprentissage qui permet de trouver une correspondance entre les données d'entrée qui sont linéairement séparées et une sortie binaire. Leur architecture contient N nœuds d'entrées fixées selon le nombre d'attributs dans les données étiquetées, suivi d'une seule couche dans le réseau (voir la figure 3.3).

Les X_i sont les entrées et les w_i sont les poids du modèle. Il existe un nœud unique dans la sortie du réseau qui prend la somme pondérée des entrées et utilise une fonction d'activation pour décider l'étiquette de sortie binaire (0 : pour la classe 1, ou

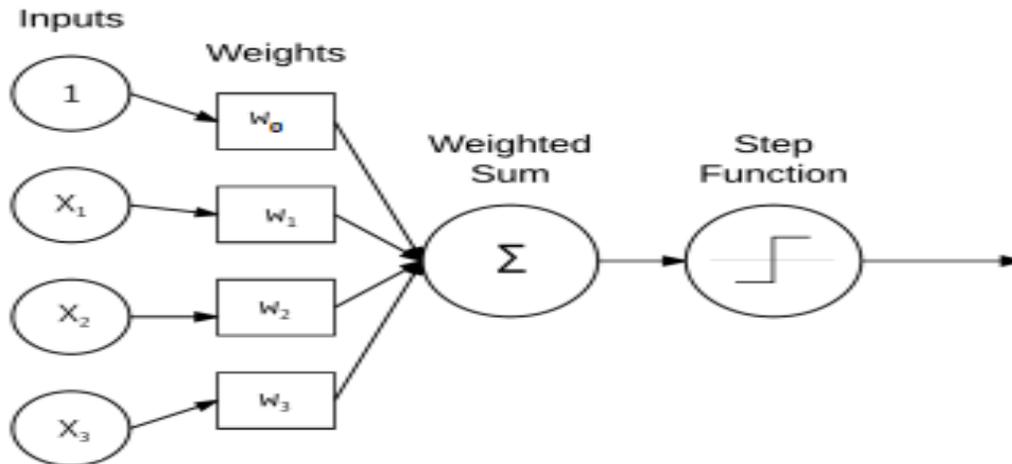


Figure 3.3 – Architecture du perceptron

1 : pour la classe 2).

La fonction d'activation utilisée dans le perceptron est la fonction d'Heaviside ou appelé aussi 'step function' (voir l'équation [3.1](#)).

$$\forall x \in \mathbb{R}, H(x) = 0 \text{ si } x < 0, \text{ et } H(x) = 1 \text{ si } x \geq 0 \quad (3.1)$$

Donc la fonction de perceptron est : $f(x) = H(x^T w)$.

Pour l'apprentissage d'un perceptron, il faut trouver l'ensemble de poids w qui classifie correctement chaque exemple dans l'ensemble de base d'apprentissage. Tout d'abord, le réseau doit s'alimenter itérativement par l'ensemble de données d'apprentissage à plusieurs époques. Nous disons une époque lorsque toutes les données d'apprentissage sont passées sur le réseau. Dans la première époque le réseau prenait le produit de vecteur d'entrée x_i avec les poids initiaux w_i et passait par la fonction d'activation pour donner soit 0 si la somme pondérée est inférieure à 0 ou 1 si non. Ensuite, mettre à jour le vecteur des poids w itérativement pour obtenir une sortie Y_j similaire au sortie réel d_j . La mise à

jour des poids se fait par la formule du delta suivant (équation 3.2) :

$$w_i(t + 1) = w_i(t) + \alpha(d_j - y_j)x_{i,j} \quad (3.2)$$

La valeur de $(d_j - y_j)$ calcul la distance entre la sortie réelle et la sortie du réseau pour déterminer si la classification est correcte. Un résultat nul ça veut dire que la classification est correcte à 100%. Si le résultat est différent de 0, la différence sera soit positive soit négative ce qui nous donne la direction de la mise à jour des poids afin de rapprocher à la classification correcte.

La valeur α est le pas d'apprentissage, il est important de bien choisir la valeur de α , une valeur très élevée permet de converger rapidement, mais elle peut dépasser un optimum global facilement. Par contre une petite valeur de α assure que ne dépasse pas un minimum global. Cependant, elle peut prendre un grand temps pour que l'apprentissage converge.

Le processus d'apprentissage se continuera jusqu'à ce que l'un des critères d'arrêt s'atteigne. Soit une valeur prédéfinie d'erreur qui montre que le total ou la majorité des données sont classées correctement ou un nombre d'époque déterminé.

Le problème de ce modèle est qu'il est applicable que si les données d'apprentissage sont linéairement séparées. La figure 3.4 illustre les tracés des fonctions ET, OU et XOR, nous pouvons clairement remarquer que les classes bleus et vertes sont séparés par une seule ligne dans les fonctions logiques 'ET' et 'OU'. Par contre, il faut deux lignes pour séparer les deux classes dans le problème de XOR. En 1969, Minsky(17) montre qu'il faut ajouter des couches pour résoudre ce problème, c'est-à-dire un perceptron multicouche et non pas un seul neurone. Un autre

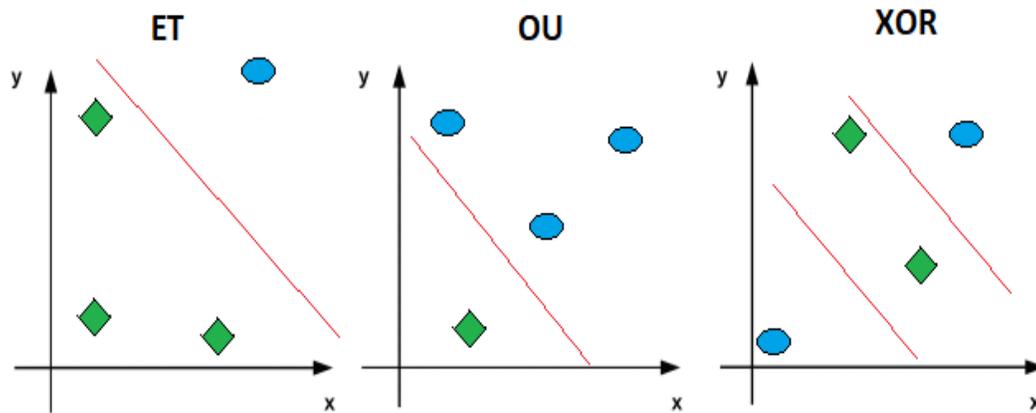


Figure 3.4 – Les fonctions logiques ET, OU et XOR. On voit que l'ensemble de données est séparé par une seule ligne dans le 'ET' et le 'OU'. Dans le 'XOR' il faut deux lignes pour séparer les données. Elles ne sont pas linéairement séparables.

problème du perceptron en relation avec la non-continuité de la fonction d'activation (Heaviside) montrée dans la figure [3.5](#) Pour entraîner un modèle de perceptron multicouche par l'algorithme de rétro-propagation (voir dans la section I.3.3), il faut que la fonction d'activation soit dérivable. La dérivabilité des fonctions d'activations permet d'utiliser un apprentissage qui se base sur la descente du gradient ([18](#)). Dans la section suivante nous présentons les trois fonctions d'activation les plus populaires dans la littérature.

3.3.3 Fonctions d'activation

La notion de fonction d'activation est inspirée par le fait qu'un signal électrique d'un neurone biologique atteint un seuil d'activation, et excite un message nerveux pour se propager au long de l'axone. Les fonctions d'activation utilisées dans les réseaux de

Fonction de Heaviside

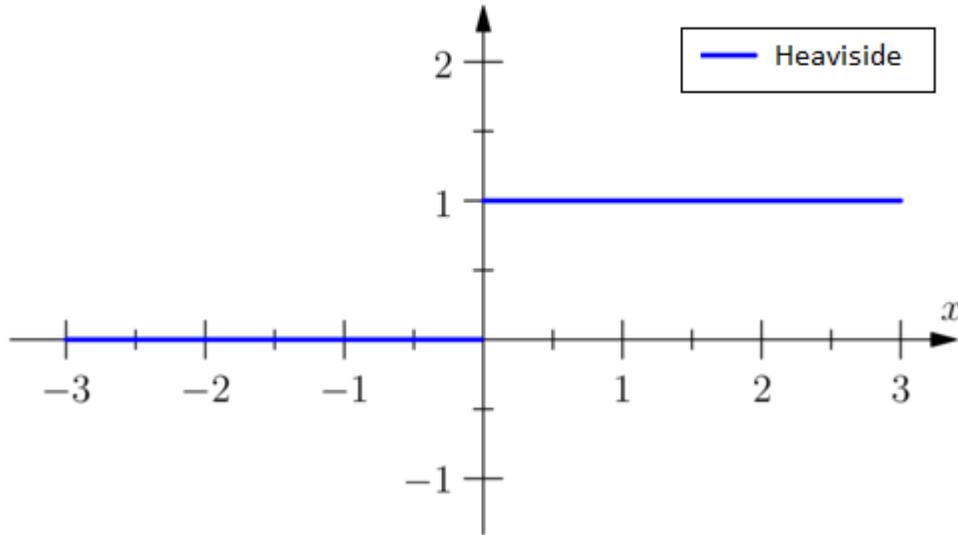


Figure 3.5 – Courbe de La fonction de Heaviside

neurones artificiels et les réseaux de neurones profonds, sont des fonctions non linéaires ce qui permet d'effectuer une classification sur des données non linéairement séparables. La figure [3.6](#) illustre les courbes des trois principales fonctions d'activation les plus utilisées dans la littérature (sigmoïde, tanh, et ReLU).

Fonction sigmoïde : comme montré dans la figure [3.6](#) la fonction sigmoïde a une forme de S, leur formule s'écrit comme suivant (équation [3.3](#)) :

$$\text{sigmoïde}(x) = \frac{1}{1 + e^{-x}} \quad (3.3)$$

Si la valeur de x est un nombre positif très grand : $f(x)$ tend vers 1. Par contre, si x est grand nombre négatif $f(x)$ tend vers 0. Donc la fonction sigmoïde traite le problème de non-dérivabilité de la fonction d'Heaviside. Elle permet de convertir toute valeur de x à une probabilité entre 0 et 1, elle est utilisée dans l'apprentissage

Fonction d'activation

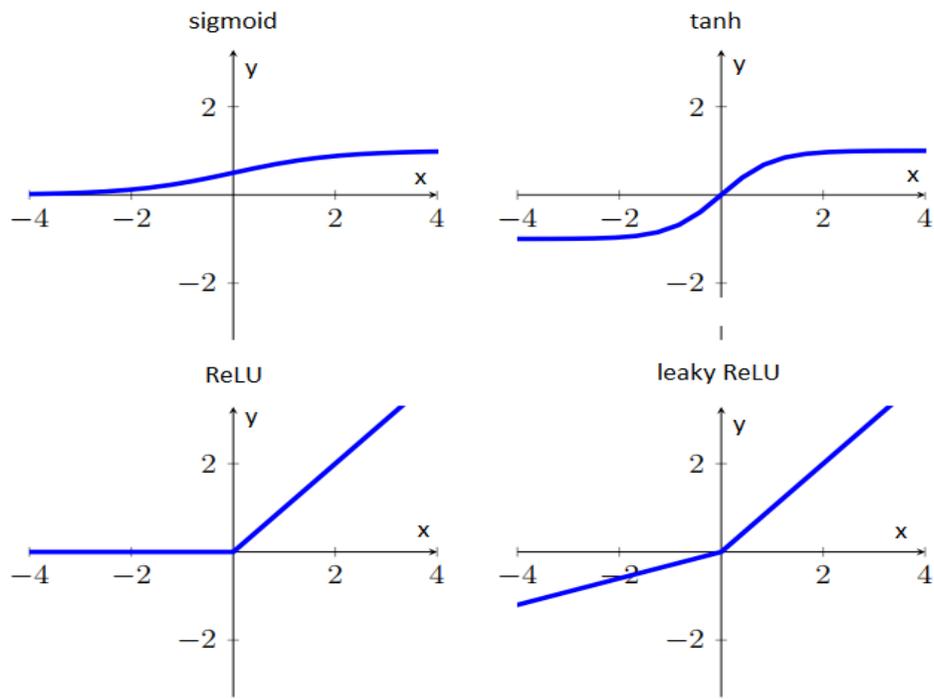


Figure 3.6 – Les tracés des fonctions d'activation les plus populaires dans la littérature.

automatique pour la régression logistique. Cette fonction traite le problème de non-dérivabilité de la fonction d'Heaviside.

Fonction tanh : La fonction d'activation tangente hyperbolique ou tanh se rapproche à la fonction sigmoïde mais avec une image dans l'intervalle $[-1 ; 1]$ (équation 3.4) :

$$\tanh(x) = 2 \operatorname{sigmoid}(2x) - 1 \quad (3.4)$$

Fonction ReLU : La fonction ReLU (Rectified Linear Unit) ou unité de rectification linéaire est présentée comme suivant (équation 3.5) :

$$\operatorname{ReLU}(x) = 0 \text{ si } x < 0, \text{ et } x \text{ si } x \geq 0 \quad (3.5)$$

Aujourd'hui la fonction ReLU devient la fonction par défaut utilisée dans les réseaux de neurones profonds. Leur avantage réside dans la simplicité de dérivation ce qui rend l'apprentissage en utilisant cette fonction très rapide, et pour toutes valeurs négatives la fonction ReLU n'active pas le neurone, cela rend les neurones non concernés inactifs. Il existe une autre version améliorée de ReLU qui s'appelle leaky ReLU, elle est présentée comme suivant (équation 3.6) :

$$\operatorname{ReLU}_{\text{leaky}}(x) = 0.01 x \text{ si } x < 0, \text{ et } x \text{ si } x \geq 0 \quad (3.6)$$

3.3.4 Rétropropagation et réseaux multicouches

La rétropropagation est un algorithme qui permet de former un réseau de neurones artificiel à partir d'un ensemble de données annoté. Jusqu'à maintenant, la rétropropagation reste le plus important algorithme dans l'histoire d'apprentissage depuis

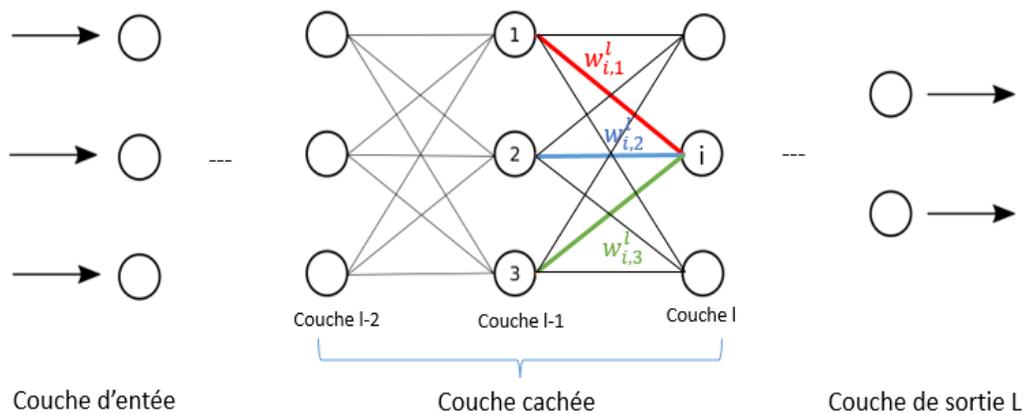


Figure 3.7 – Réseau de neurones avec L couche, Les poids du neurone i de la couche 'L' issue du j^{ime} neurone de la $(l - 1)^{ime}$ couche précédente sont montrés en rouge, bleu et vert.

Leur découverte en 1986 par Rumelhart et al (19). Pour bien comprendre le principe de fonctionnement de cet algorithme, on prend l'exemple d'un réseau de neurones multicouches (aussi appelés perceptrons multicouches PMC). Dans ce réseau, la propagation de l'information se déplace dans une seule direction vers l'avant, pour ce qui suit nous utilisons les notations au-dessous pour décrire l'algorithme de rétropropagation. La figure 3.7 montre un exemple d'un réseau de neurones multicouches.

- $w_{i,j}^l$: est le poids qui relie le neurone i de la couche L au neurone j de la couche $L-1$ précédente.
- L : est le nombre total des couches de réseau.
- σ : est la fonction d'activation.
- b_i^l : est le biais de neurone i de la couche L .

La sortie d'un neurone i de la couche L se calcule en utilisant les informations d'entrée de ce neurone, qui sont les sorties des neurones de la couche précédente ($L-1$), ainsi que les poids de

connexions synaptiques. Donc pour propager vers l'avant dans un PMC, on doit réaliser les calculs suivants :

$$z_i^l = \sum_{j=1}^n w_{ij}^l a_j^{l-1} + b_i^l \quad (3.7)$$

$$a_i^l = \sigma(z_i^l) \quad (3.8)$$

Avec :

- z_i^l : est la valeur d'agrégation de neurone i de la couche L. Cette valeur est calculée avant de passer dans la fonction d'activation dans l'équation [3.8](#).
- a_i^l : est la sortie de neurone i de la couche L (valeur d'activation).

Pour simplifier la notation on va décrire ces équations sous forme matricielle. ça devient donc :

- z^l : la matrice des poids qui relie la couche (L-1) et L : $z^l = z_1^l, z_2^l \dots \dots, z_n^l$
- a^l : est le vecteur des activations des neurones de la couche L : $a^l = a_1^l, a_2^l \dots \dots, a_n^l$
- b^l : est le vecteur des biais des neurones de la couche L : $b^l = b_1^l, b_2^l \dots \dots, b_n^l$

On obtient alors l'équation [3.9](#), une version vectorisée des équations précédentes :

$$a^l = \sigma(w^l * a^{l-1} + b^l) \quad (3.9)$$

L'algorithme de rétropropagation permet de calculer l'erreur ϵ^{sortie} en comparant la sortie réelle avec la sortie du réseau. Ensuite,

propager cette erreur vers l'arrière. Donc pour un coût C on cherche à déterminer dans les équations (3.10) et (3.11) les dérivées partielles du coût pour chaque poids et biais.

$$\frac{\partial C}{\partial w_{ij}^l} = a_j^{l-1} \sigma_i^l \quad (3.10)$$

$$\frac{\partial C}{\partial b_i^l} = \sigma_i^l \quad (3.11)$$

En savant la dérivée partielle de l'erreur par rapport aux poids et biais du réseau, on peut optimiser ces paramètres en utilisant l'algorithme de descente de gradient expliqué dans la section 3.4.4 jusqu'à ce que $C(w, b)$ converge.

Les perceptrons multicouches se composent de plusieurs couches qui sont entièrement connectées (c-t-d chaque neurone est connecté à chaque neurone de la couche suivante), avec une fonction de coût (ou fonction objective) à optimiser. Leur mécanisme d'apprentissage permet de résoudre plusieurs problèmes dans la littérature. Cependant, les PMC ne peuvent pas apprendre que sur des données unidimensionnelles. Les réseaux de neurones convolutifs viennent pour traiter ce problème, ils permettent de représenter les modèles à dimensions supérieures (2D et 3D).

Dans la suite de ce chapitre, nous détaillerons les principes fondamentaux des réseaux de neurones convolutifs, leurs architectures ainsi que les méthodes d'optimisation et de régularisation. Nous terminerons ce chapitre par une revue sur les modèles les plus connus dans la littérature.

3.4 Réseaux de neurones convolutifs

Un réseau de neurones convolutifs (CNN ou ConvNet), est un algorithme d'apprentissage profond conçu pour travailler avec des données en deux dimensions, comme ils peuvent être appliqués sur des données tridimensionnelles et unidimensionnelles. Les CNNs sont très similaires aux réseaux de neurones classiques présentés dans la partie précédente, avec plusieurs couches constituées de neurones qui ont des poids et des biais modifiables par un processus d'apprentissage.

La particularité des CNNs est qu'ils n'ont pas besoin d'un pré-traitement pour injecter les données en entrant dans le réseau comme les algorithmes de classification traditionnels, les CNNs ont la capacité d'apprendre les caractéristiques à partir des données brutes (image par exemple). En outre, les neurones des différentes couches ne sont pas entièrement connectés dans les CNNs. La couche la plus importante est la couche de convolution (qui donne son nom au réseau), chaque couche de convolution contient plusieurs noyaux de 3 dimensions (hauteur, largeur, et profondeur, d'où vient le mot apprentissage en profondeur et non à la profondeur liée au nombre total de couches dans un réseau) qui servent à calculer des cartes de caractéristiques.

Dans la dernière couche entièrement connectée d'un CNN, il existe toujours une fonction de perte (Softmax par exemple) pour classer les caractéristiques extraites dans les couches précédentes avec des classes et leurs scores. Alors, l'architecture des CNNs permet d'encoder certaines caractéristiques des entrées, ce qui rend la classification plus efficace et facile à mettre en œuvre,

tout en réduisant considérablement le nombre total des paramètres de réseau.

3.4.1 Architecture des CNNs

Si on veut appliquer des informations brutes comme les pixels d'une image sur l'entrée d'un réseau de neurones multicouches, le nombre des pixels sera ingérable par cette structure entièrement connectée, surtout avec des images de grande taille. Par exemple une image de taille $512 \times 512 \times 3$ (512 de largeur, 512 d'hauteur, 3 pour les trois canaux de couleur RGB) contient 786 432 neurones ($512 \times 512 \times 3$) dans la couche d'entrée seulement. En plus de ce nombre important des neurones et des poids, ce réseau est souvent exposé au problème de sur-apprentissage, cette connectivité totale peut être considérée comme un gaspillage de temps et de mémoire.

Contrairement aux réseaux de neurones classiques, les réseaux de neurones convolutifs considèrent que l'entrée est une image de trois dimensions (largeur, hauteur, profondeur), et adaptent leur architecture d'une manière appropriée. Dans un CNN, les neurones d'une couche ne sont pas entièrement connectés avec la couche qui la précède, et à la couche de sortie finale toutes les caractéristiques calculées sont réduites en un seul vecteur de $1 \times 1 \times N$ (avec N est le nombre de classes) pour déduire le vecteur de score final qui contient les classes avec leurs scores (voir figure 3.11).

Comme nous l'avons présenté précédemment, une architecture CNN est une suite des couches, chaque couche transforme une entrée de trois dimensions en volume de sortie grâce à une fonction

différentiable. Les principales couches utilisées pour construire des architectures CNN complètes sont : couche de convolution (CONV), couche d'activation (ACT ou ReLu), couche de sous-échantillonnage ou pooling en anglais (POOL), et couche entièrement connectée (FC).

Pour construire une architecture CNN, on empile une suite de ces couches d'une manière spécifique, généralement un simple diagramme textuel est utilisé pour décrire une architecture. Dans l'exemple suivant, nous construirons un réseau de neurones profond avec deux couches de convolution, chaque CONV est suivi par une fonction d'activation ReLU, puis une couche entièrement connectée, et à la fin une couche d'activation Softmax pour donner les classes et leurs probabilités.

INPUT IMAGE => CONV => ReLu => CONV => ReLu => FC => SOFTMAX => OUTPUT

Seules les deux couches 'CONV et FC' ont des paramètres (ou poids) modifiables par un processus d'apprentissage. Dans ce qui suit, nous entrerons plus en détail dans ses différentes couches et leurs hyperparamètres.

3.4.2 Couches des CNNs

Dans cette partie, nous décrivons les différentes couches utilisées pour créer des réseaux de neurones convolutifs.

Couche de convolution

Dans un réseau de neurones profond, la couche de convolution ou CONV est la couche la plus importante, elle est constituée d'un ensemble de filtres de petite taille (matrice des poids) qui

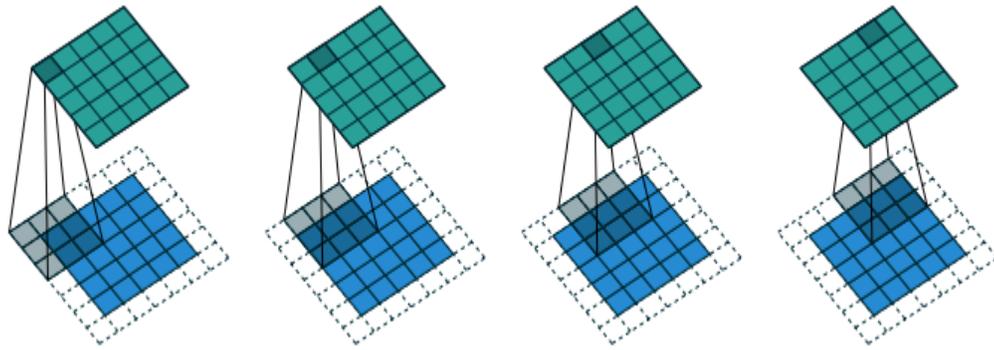


Figure 3.8 – Processus de convolution extraite de (3). Le noyau ici est de taille 3x3, l'entrée en bleu est de taille 5x5 avec un pas de 1 et un remplissage de 1.

glisse sur la largeur et la hauteur d'une entrée en calculant les produits scalaires dans toutes les positions, puis appliquer une transformation non linéaire sur la somme des produits calculés. La valeur obtenue sera enregistrée dans une localité au sein de la carte de caractéristique résultante (voir la figure 3.11). Chaque filtre produira une carte de caractéristique bidimensionnelle séparée, pour les empiler selon la profondeur de l'entrée et injecter dans la couche qui suit. Les valeurs de filtre (appelé aussi noyau ou fenêtre de convolution) sont les paramètres à optimiser par le processus d'apprentissage.

Lors de la conception d'un CNN, on fixe le nombre total des filtres pour chaque couche, leurs tailles, et le pas de convolution. Un pas de convolution correspond au nombre de pixels à décaler par les noyaux après chaque calcul. Il existe un autre hyperparamètre à contrôler lors de la conception d'un réseau, qui concerne la taille de remplissage par zéro autour des bordures des entrées, ce dernier hyperparamètre permet de conserver les dimensions des

entrées après la convolution avec les filtres. Puisque la taille de fenêtre de convolution est toujours impaire et on s'intéresse à un pixel central dans les calculs des produits de convolution, il est impossible de calculer les pixels qui se trouvent dans les bords d'une entrée, ce qui nous oblige soit à négliger ces pixels, soit à ajouter une marge pour que les filtres puissent traiter ces pixels. Donc les paramètres qu'on doit fixer en dehors de l'apprentissage ce sont des hyperparamètres. Nous pouvons calculer la taille de la carte des caractéristiques de sortie (O) en fonction de la taille d'entrée (E), la taille du noyau (N), le pas de convolution en anglais stride (S), et la taille de remplissage des zéros ou zéro-padding (P) par la formule de l'équation (3.12) suivante :

$$O = (E - N + 2 * P) / S + 1 \quad (3.12)$$

Par exemple une entrée de taille 12x12, et un filtre de taille 3x3 avec un pas de 1 et 0 padding, donne une sortie de 10x10, et avec un remplissage de 1 on peut préserver la taille de l'entrée et dans ce cas, la taille de sortie sera 12x12.

Nous pouvons voir aussi que si un pas de convolution est supérieur à 1, la taille de sortie sera réduite, ce qui provoque une réduction de résolution de la carte des caractéristiques. Généralement, les concepteurs des réseaux utilisent la couche de pooling présenté dans la section 3.4.2, pour ce processus réduction, pour cela le pas de convolution est souvent fixé à 1.

Couche d'activation ReLU

Après chaque couche de convolution dans un réseau de neurones profond, il existe une fonction d'activation non linéaire telle

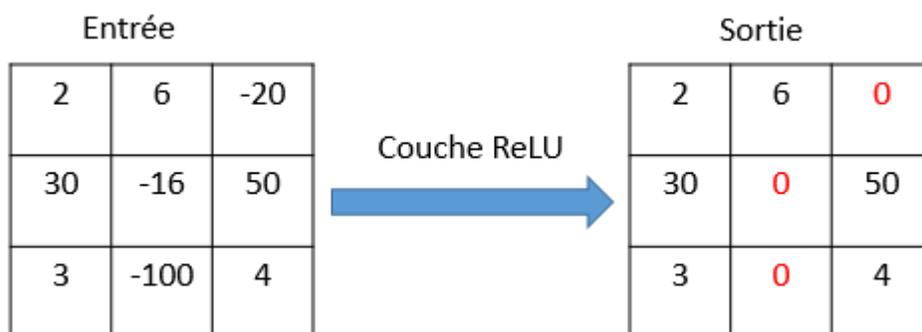


Figure 3.9 – Principe de fonctionnement de ReLU

que présentée dans la section [3.3.3](#). Dans la littérature, la fonction d'activation ReLU (Rectified Linear Unit) reste la plus utilisée à cause de son efficacité par rapport aux autres. L'objectif de ReLU est d'apporter la non-linéarité dans les CNNs en appliquant une fonction d'activation f , sur la carte de caractéristique de sortie, et donner en sortie une carte de la même dimension.

Avec : $f(x) = \max(0, x)$

La figure [3.9](#) montre le principe de fonctionnement de fonction d'activation ReLU.

Couche de pooling

Pour réduire le nombre des paramètres dans les cartes de caractéristiques, soit nous utilisons un pas supérieur à 1 dans la couche de convolution comme nous avons déjà parlé dans la section [3.4.2](#), soit on insère des couches de pooling entre les couches CONV. L'objectif principal de la couche de pooling (en français sous-échantillonnage) est de réduire la largeur et la hauteur des cartes de caractéristique en conservant les informations importantes afin de baisser les calculs et contrôler le sur-apprentissage

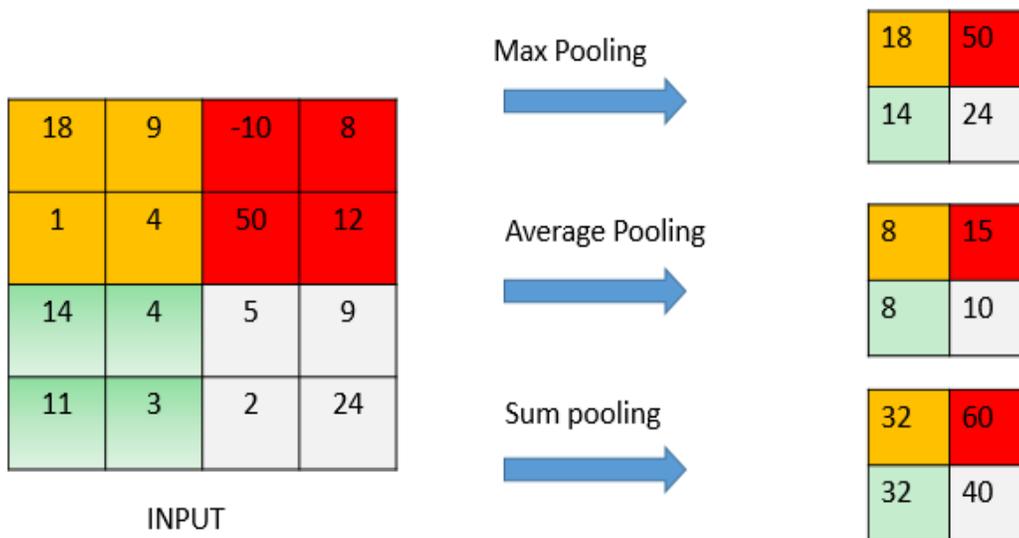


Figure 3.10 – Exemple de pooling avec les différentes stratégies (maximum, moyen et somme). La taille des fenêtres ainsi que le pas utilisé dans cet exemple sont de 2x2.

dans un CNN.

Le principe de fonctionnement du pooling se déroule de la même manière que la couche de convolution, des fenêtres de taille fixe $P \times P$ glisse sur les cartes de caractéristiques d'entrée indépendamment avec un pas P en utilisant soit une fonction max, moyenne, ou somme. Généralement la taille utilisait des fenêtres de pooling ainsi que les pas sont de 2x2, ce qui réduit les dimensions des entrées à la moitié. La figure 3.10 montre un exemple d'application de pooling avec les différentes stratégies max, moyen, et somme.

3.4.3 Couche entièrement connectée

Dans la fin des CNN, la matrice de caractéristiques sera transformée en un vecteur et l'injecter dans une couche entièrement connectée, c'est exactement comme un réseau de neurones multicouches qu'on a présenté dans la section [3.3.4](#). Tous les neurones de la couche FC sont connectés avec les activations de la couche qui précède. Généralement les concepteurs des CNN utilisent une ou deux couches entièrement connectées avec un classifieur SOFTMAX pour calculer les probabilités finales pour chaque classe prédite.

Dans l'exemple illustré dans la figure [3.11](#) nous avons construit une architecture CNN avec deux couches de convolution, après chaque CONV une couche d'activation ReLU et une couche de POOL. A la fin du réseau deux couches FC sont appliquées avant un classifieur SOFTMAX.

INPUT => CONV => ReLU => POOL => CONV=> ReLU
=> POOL => FC=>FC=> SOFTMAX=> OUTPUT

3.4.4 Méthodes d'optimisation

Nous avons vu dans les sections précédentes les principes fondamentaux d'apprentissage artificiel, et comment il peut trouver une correspondance entre les données d'entrée et les étiquettes de sortie. Cette correspondance est faite à travers une fonction qui utilise la matrice des poids W et le vecteur de biais b , et renvoie une prédiction pour chaque entrée. Nous avons également discuté sur l'architecture des CNNs et les principaux blocs qui construisent un réseau de neurones convolutif. Sur la base

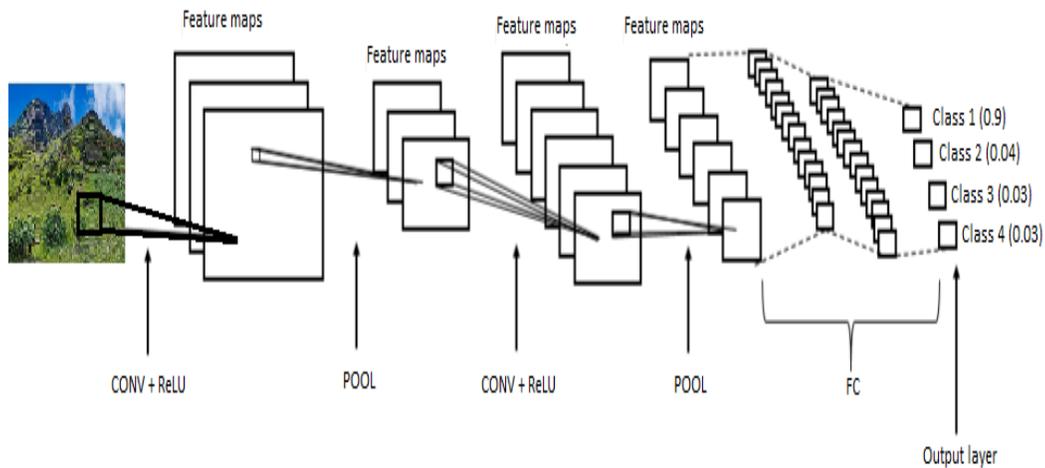


Figure 3.11 – Architecture d'un réseau de neurones convolutif complète.

de ces éléments, nous pouvons passer maintenant aux méthodes d'optimisation permettant aux modèles d'apprendre à partir des données.

L'objectif d'un apprentissage automatique est de trouver un ensemble de paramètres (poids W , et biais b), de sorte que les données d'entrée soient bien classées. Mais le problème qui se pose, quelle est la meilleure stratégie à suivre pour obtenir une combinaison de W et b qui donnent une bonne précision de classification.

Dans ce qui suit, nous allons présenter l'algorithme le plus utilisé dans l'apprentissage profond, à savoir la descente de gradient ainsi que ses variantes qui se reposent sur le même principe. Nous passerons ensuite aux améliorations possibles qui peuvent être apportées aux modèles profonds, ce qu'on appelle également les méthodes de régularisation.

Descente de gradient

L'algorithme de descente du gradient est un algorithme itératif qui permet de minimiser la fonction de coût $C(w,b)$ successivement en modifiant les paramètres w et b du modèle. Son objective est de trouver la combinaison qui s'approche à un minimum global, en calculant le gradient $\nabla_{(w,b)}C(w,b)$ et en mettant à jour les poids w et biais b dans la direction de la pente de la fonction C . La figure [3.12](#) illustre la stratégie de déplacement d'une fonction vers un minimum en utilisant sa dérivée. Un hyperparamètre appelé pas d'apprentissage, doit être déterminé au début pour fixer la longueur de pas à faire dans la direction du gradient.

Dans chaque itération de l'algorithme d'optimisation, les paramètres sont modifiés selon l'importance des neurones. Les paramètres des neurones qui ont participé à reproduire une erreur finale importante sont modifiés significativement par rapport aux autres neurones qui ne contribuent pas clairement dans l'erreur finale. Donc, pour mettre à jour le modèle on doit soit diminuer, soit augmenter les paramètres du modèle.

Variantes de l'algorithme de descente de gradient

L'algorithme de descente de gradient à trois variantes connues. La première appelée descente de gradient par batch, qui consiste à calculer la fonction du coût et son gradient sur toute la base d'apprentissage. Leur avantage est que le gradient calculé, donne une direction exacte de la pente pour minimiser l'erreur globale. Cependant, pour chaque pas il faut évaluer toutes la base d'ap-

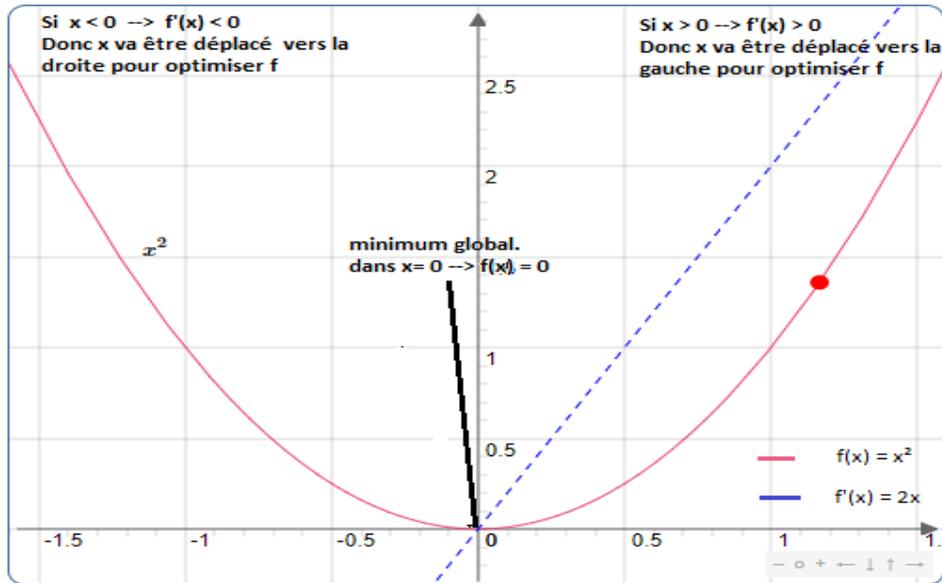


Figure 3.12 – Principe de fonctionnement de l’algorithme de descente de gradient pour trouver un minimum d’une fonction f par sa dérivée (2)

prentissage pour calculer l’erreur de chaque exemple, ce qui peut être très lourd à exécuter sur une grande base de données.

Une deuxième variante appelée descente de gradient stochastique, cette méthode permet de calculer les erreurs des exemples un par un en les choisissant aléatoirement, ensuite mettre à jour les paramètres après chaque évaluation. Cette stratégie permet d’ajuster les paramètres du réseau plus rapidement et donc une convergence plus rapide. Cependant, la modification des paramètres pour chaque exemple peut conduire à un minimum local. En outre, si la base d’apprentissage contient des exemples compliqués ou faux exemple (annotation incorrecte par exemple), la direction du gradient peut être effectué dans la mauvaise direction, mais ça sera corrigé par les exemples corrects qui suivent.

Ce problème a été traité dans la troisième variante de descente de gradient, qui situe entre les deux précédentes, cette variante appelée gradient par mini-batch. Elle consiste à calculer les erreurs sur des petits lots d'une taille prédéfinie par l'utilisateur, puis calculer l'erreur moyenne et enfin modifier les paramètres. L'évaluation par mini-batch permet de trouver un gradient plus précis sans être obligé de passer sur tout l'ensemble de données pour ajuster les paramètres.

Il existe d'autres améliorations sur l'algorithme de descente de gradient permettant une convergence plus rapide à un minimum local de la fonction du coût. Parmi lesquels on peut mentionner Momentum (20), NAG (Nestrov accelerated gradient) (21), Adadelta (22), RMSprop (23), et Adam (Adaptive Moment Estimation) (24).

3.4.5 Régularisation

Comme nous l'avons présenté dans les sections précédentes, la tâche d'optimisation permet de réduire l'erreur calculée à partir d'un ensemble de données d'apprentissage (erreur d'apprentissage) en ajustant les paramètres du modèle jusqu'à ce qu'on trouve la combinaison optimale. Dans l'apprentissage artificiel (Machine Learning), l'objectif central est de trouver un modèle capable de distinguer non seulement les données d'apprentissage, mais aussi les nouvelles données. Lorsque l'erreur de test est significativement plus élevée que l'erreur d'apprentissage, cela veut dire que le modèle souffre d'un sur-apprentissage, par contre si l'erreur de test est élevée, notre modèle fait face à un sous-apprentissage. Il existe de nombreuses stratégies pour contrôler l'apprentissage

des modèles pour éviter le problème de sur-apprentissage et réduire l'erreur de test, ces stratégies appelées les méthodes de régularisation. Il existe plusieurs façons de contrôler les réseaux de neurones profonds, comme la modification directe sur l'architecture en ajoutant des couches spécifiques permettant de réduire le nombre de paramètres, comme par exemple la couche Dropout qui vise à réduire la complexité du modèle en augmentant la capacités de généralisation. La couche Dropout permet de déconnecter aléatoirement les entrées venant de la couche précédente avec une probabilité fixée par le concepteur du réseau. Généralement, la couche Dropout est placée entre les dernières couches entièrement connectées dans une architecture profonde avec une probabilité moyenne de 0.5.

Il existe d'autres techniques de régularisation qui sont appliquées pendant le processus d'apprentissage tel que l'augmentation des données et l'arrêt précoce (Early Stopping).

Dans la littérature il y a un autre type de régularisation qui touche la fonction de coût. C'est un terme de régularisation qui est ajouté à la fonction du coût initial afin de limiter l'entraînement du modèle sur les données d'apprentissage. Ce terme permet de pénaliser les paramètres qui donnent une importance trop élevée à un détail dans les données d'entrée. En général, il existe deux méthodes qui sont largement utilisées dans ce type de régularisation :

- La régularisation L2 : dans cette première méthode, le terme de pénalité ajouté à la fonction du coût est une valeur au carré des poids synaptiques w , la fonction du coût devient

comme suite :

$$C(x) = C(x) + \lambda \sum_{j=1}^p w_j^2 \quad (3.13)$$

- La régularisation L1 : dans cette méthode le terme de pénalité ajouté à la fonction du coût est la valeur absolue des poids synaptique w au lieu de valeur au carré comme dans la régularisation L2. La fonction du coût C devient donc :

$$C(x) = C(x) + \lambda \sum_{j=1}^p |w_j| \quad (3.14)$$

Avec : λ est le coefficient de régularisation dans les deux méthodes, il doit être fixé par l'utilisateur comme un hyperparamètre avant de lancer l'apprentissage du modèle. Il est responsable de l'importance de régularisation. S'il est nul, la fonction de coût devient à sa forme de base initiale, donc la complexité du modèle restera la même comme si on n'a pas appliqué la régularisation. S'il est très élevé, la modification significative des poids synaptiques peut conduire à un sous apprentissage. Donc le choix de ce coefficient est très important dans ce type de régularisation.

3.4.6 Modèles existants dans l'état de l'art

Nous avons présenté dans les sections précédentes les éléments de base des réseaux de neurones convolutifs, leurs apprentissages, ainsi que les stratégies permettant d'améliorer leurs performances. Dans ce qui suit, nous passerons en revue sur les modèles de l'état de l'art les plus populaires, qui ont contribué significativement dans le développement du domaine de reconnaissance d'image. Ces modèles prennent leurs succès dans le

cadre d'une compétition annuelle nommée ILSVRC (ImageNet Large Scale Visual Recognition Challenge') (2), qui utilise la base de données ImageNet (25) pour évaluer les différentes architectures. La base de données ImageNet contient plus de 14 millions d'images annotées sur 1000 catégories d'objets tirées de la base lexicale WordNet.

LeNet5

En 1990, Yann LeCun et al. (4) ont développé le premier réseau de neurones convolutif appelé LeNet5, qui était utilisé pour la reconnaissance des chiffres manuscrits. L'architecture LeNet (figure 3.13) prend en entrée une image en niveau de gris de taille 32x32 et donne une sortie de taille 10 contenant les catégories des chiffres de 0 à 9.

Le réseau LeNet5 est composé de deux couches de convolution avec des noyaux de taille 5x5, chaque couche de convolution est suivie par une couche de pooling, et à la fin du réseau, deux couches entièrement connectées suivies d'une couche de sortie pour la classification. La raison de la petite taille de l'architecture est due aux limitations des capacités de calcul à cette époque.

AlexNet

Après 22 ans de la création du réseau LeNet5, les réseaux de neurones convolutifs commencent à populariser grâce à l'architecture AlexNet créée par Alex Krizhevsky et al (5). Le modèle AlexNet a été soumis en 2012 aux compétitions ILSVRC et a permis d'atteindre une précision de 84.6% (leur proche concurrent

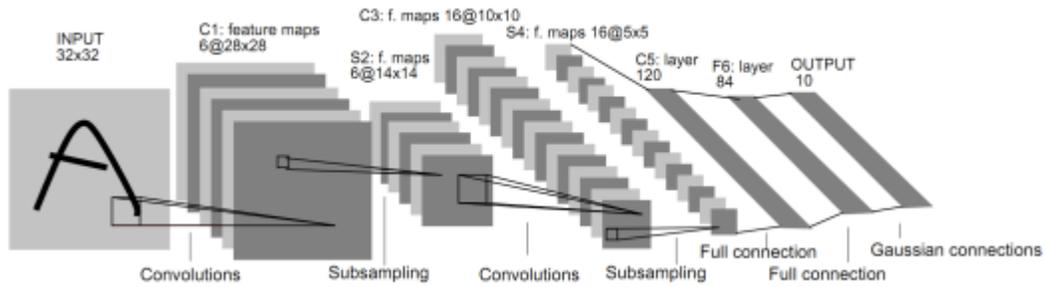


Figure 3.13 – Architecture du réseau LeNet5 (4)

atteint une précision de 73.8%). L'architecture AlexNet est inspirée par le modèle LeNet avec plusieurs couches de convolution empilées les unes sur les autres au lieu d'une seule couche, elle utilise aussi des couches plus récentes comme la couche d'activation ReLU et la technique dropout.

Comme montré dans la figure 3.14, l'architecture AlexNet est composée de cinq couches de convolution, chaque couche est suivie par une couche max-pooling et une fonction d'activation ReLU. A la fin du réseau, deux couches entièrement connectées avec l'opérateur de dropout qui permet d'éviter le problème de sur-apprentissage.

L'avantage important du modèle AlexNet est qu'il utilise deux chemins indépendants avant la dernière couche entièrement connectée, cela permet d'optimiser le temps de calcul en formant le réseau sur deux GPU en parallèle sans avoir transféré les informations entre eux jusqu'à la dernière couche.

GoogLeNet

Le réseau convolutif GoogLeNet était le gagnant de la compétition ILSVRC-2014 dans la tâche de classification avec une

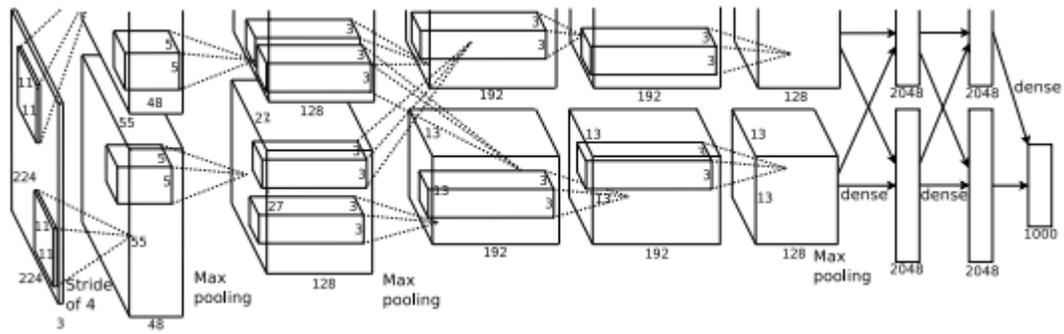


Figure 3.14 – Architecture du réseau de neurone convolutif Alex-Net (5)

précision de 93.3%. Il a été proposé par Szegedy et al. (6) de Google, ils ont introduit dans leur architecture un module appelé 'inception modul' qui permet de réduire le nombre de paramètres dans le réseau (4 millions au lieu de 60 millions d'AlexNet). Ce module est composé de quatre chemins en parallèle comme montré dans la figure 3.15, les résultats obtenus par ces quatre chemins sont concaténés par la suite et l'injecter dans le module qui suit. La figure 3.16 montre l'architecture complète du réseau GoogLeNet, il existe d'autres versions de cette architecture, la plus récente est celle de inception-V4.

VGGNet

Le gagnant de la compétition ILSVRC 2014 dans la tâche de localisation était le réseau convolutif de l'équipe Visual Geometry Group (VGG) de l'université d'Oxford connu sous le nom de VGGNet (12). Les chercheurs de l'équipe VGG ont étudié le réseau AlexNet et ont montré que la profondeur du réseau joue un rôle essentiel pour donner des bons résultats. Leur model ga-

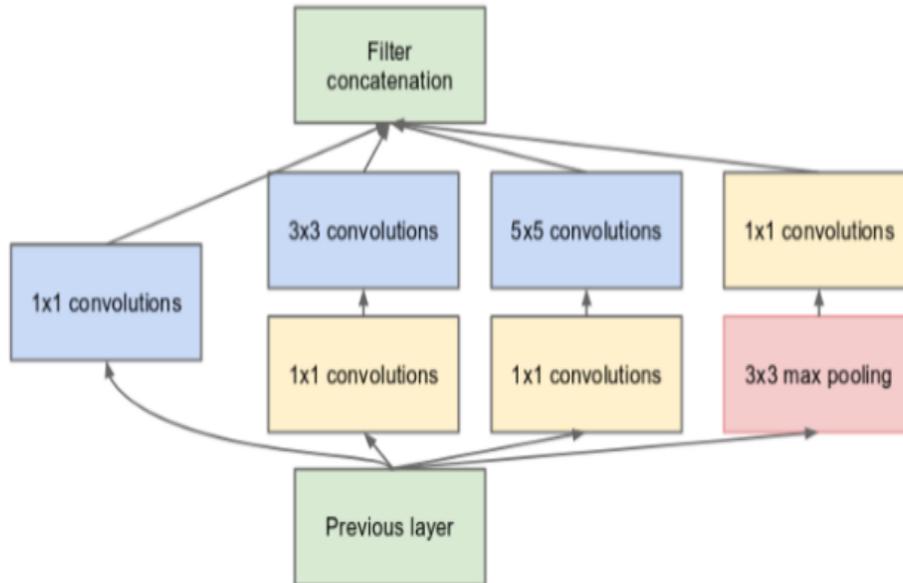


Figure 3.15 – Module inception extraite de (6). Il est composé de quatre chemin avec des convolutions qui ont des noyaux de taille différentes.

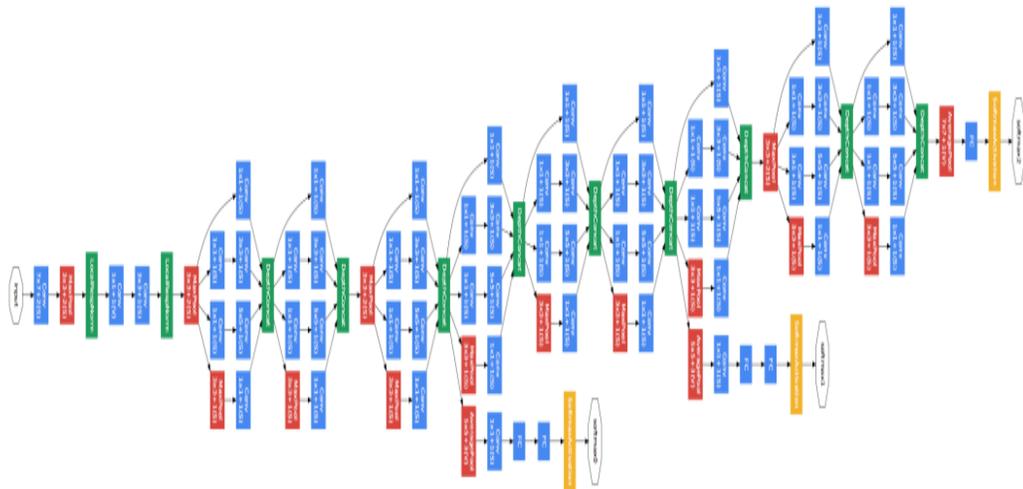


Figure 3.16 – Architecture complète du réseau GoogLeNet (6)

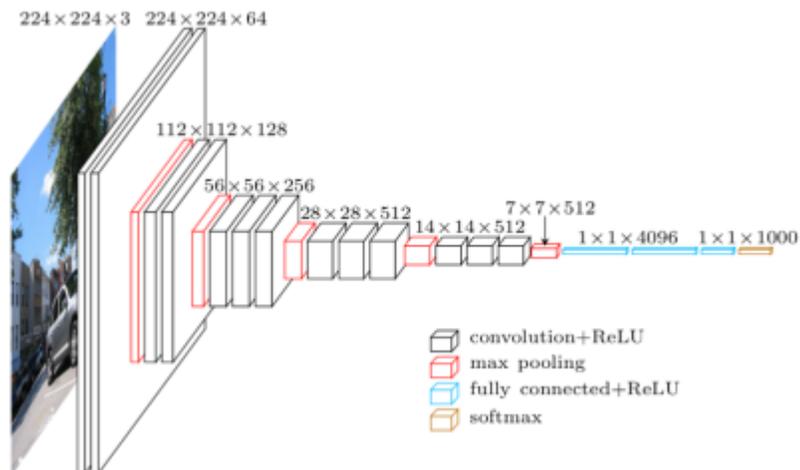


Figure 3.17 – Architecture VGG16 extraite de (7)

gnant (avec une précision de 92.7 %) appelé VGG16 se compose de 16 couches de convolution avec des tailles des noyaux de 3x3 et une pooling de 2x2 (Figure 3.17). Cette architecture possède d'une grande capacité de discrimination, mais elle est plus coûteuse à évaluer puisqu'elle utilise beaucoup plus de mémoire et de paramètres (140 millions).

ResNet

Le réseau convolutif ResNet (Residual Network) introduit par Kaiming He et al.(8). Il a été le gagnant de la compétition ILSVRC-2015. Ils utilisent dans leur architecture des connexions résiduelles qui consiste à additionner la sortie d'une couche avec leur entrée (voir la figure 3.18). Ces couches résiduelles permettent d'éviter le problème de disparition de gradient liée à la profondeur des réseaux, c-à-d que lorsque la profondeur augmente dans les réseaux résiduels les performances sont meilleures que les réseaux de neurones convolutifs simples qui n'utilisent pas les

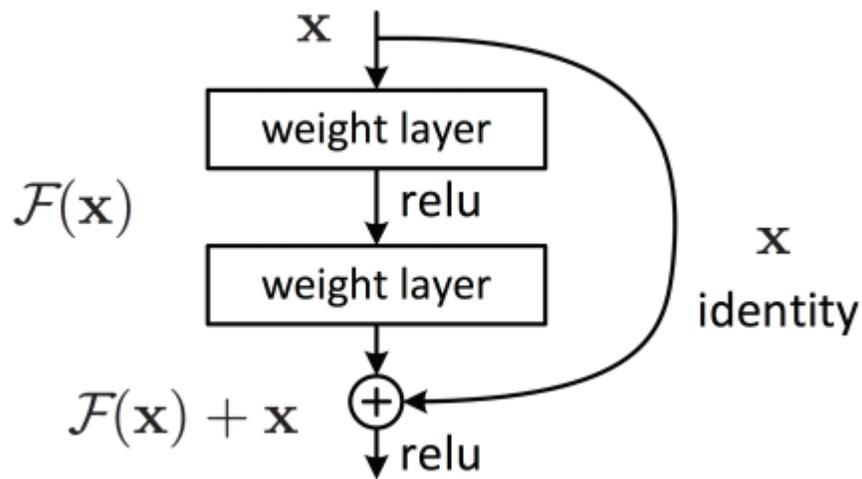


Figure 3.18 – Principe d’une connexion résiduelle extraite de (8). L’entrée X est ajoutée au résultat $F(x)$

connexions résiduelles.

Cette technique permet d’augmenter la capacité de discrimination des réseaux de neurones convolutifs en utilisant des réseaux plus profonds. Dans l’article de He et al. (8) Les auteurs ont formé des modèles profonds allant de 50 couches jusqu’à 152 couches. Actuellement, les ResNet sont devenus les modèles de l’état de l’art utilisé par défaut dans la pratique.

3.5 Conclusion

Dans ce chapitre nous avons présenté les concepts fondamentaux des réseaux de neurones convolutifs (CNN). Nous avons commencé par discuter les trois principaux types d’apprentissage automatique, l’apprentissage supervisé, non supervisé et semi-supervisé. Dans cette thèse nous utilisons principalement l’ap-

apprentissage supervisé où les étiquettes de classe sont disponibles. Ensuite nous avons passé en revue les principes des réseaux de neurones artificiels traditionnels. Plus, nous avons mis l'accent sur les réseaux de neurones et leurs relations avec les neurones biologiques. Dans cette partie, nous avons présenté le principe d'algorithme de perceptron et sa limitation lorsqu'il s'agit des données non linéaires. Pour résoudre le problème de la non-linéarité dans l'algorithme de perceptron, nous devons utiliser à la fois, une fonction d'activation non linéaire et un réseau de neurones multicouches au lieu d'un seul neurone. La formation des réseaux multicouches en utilisant l'algorithme de rétropropagation et les fonctions d'activations non linéaires permet de travailler avec des bases de données plus complexes qui contiennent des exemples qui ne sont pas linéairement séparables.

Après que nous avons introduit les bases d'apprentissage automatique et les réseaux de neurones artificiels et leurs apprentissages, nous avons présenté dans la section qui suit les concepts fondamentaux des réseaux de neurones convolutifs. Nous avons commencé cette partie par une présentation des différents éléments qui construisent les réseaux de neurones convolutifs ainsi que l'algorithme utilisé pour ajuster les paramètres d'apprentissage. Nous avons ensuite discuté sur la régularisation qui peut être définie comme une technique qui permet d'améliorer la précision de test. Finalement, nous avons conclu le chapitre par une revue sur les architectures CNN les plus populaires, ces architectures peuvent être utilisées pour développer un modèle spécialisé dans notre tâche.

4 Méthode réalisée pour la segmentation des globules blancs

4.1 Introduction

Dans l'analyse cytopathologique, la morphologie des globules blancs et/ou le comptage d'instance dans les frottis sanguins peuvent avoir une signification médicale pour le diagnostic de plusieurs maladies cancéreuses (14) (26). Cependant, la détection et la segmentation manuelle de ces structures peuvent être des tâches coûteuses, fastidieuses et prennent beaucoup de temps, surtout avec la variabilité des structures et le chevauchement entre les objets de l'image. Par conséquent, les techniques de recherche et de segmentation automatique peuvent aider les hématologues dans leur pratiques cliniques en améliorant l'efficacité et la fiabilité, et ainsi accélérer le traitement des patients.

Dans ce qui suit, nous présentons d'abord les différents types de segmentation existents dans la littérature en détaillant la segmentation par les modèles d'apprentissage profond. Ensuite, nous citons quelques méthodes de l'état de l'art du domaine de la segmentation des images microscopiques. Enfin nous présentons l'approche proposée pour la reconnaissance automatique des images cytologiques en utilisant l'apprentissage profond.

4.2 Généralités sur La segmentation

La segmentation est une étape très importante dans le domaine de vision par ordinateur, elle peut être définie comme un processus d'identification d'un groupe de pixels comme appartenant à un objet ou une région significative dans une image. Dans le domaine médical, la segmentation consiste à extraire des structures anatomiques, tumeurs, cellules, etc. afin d'acquérir des informations qualitatives et quantitatives nécessaires pour le diagnostic médicales. Les méthodes de segmentation dans la littérature sont diverses, et peuvent se regrouper en deux grandes familles (méthodes classiques et méthodes basées sur l'apprentissage profond). Précédemment, la segmentation d'image est basée sur des techniques traditionnelles comme le seuillage (27), k-means clustering (28), croissance des régions (29), Les bassins versants (watersheds) (30), jusqu'à des techniques plus avancées tels que l'algorithme de contour actif (31), Les coupes de graphes (32) (33), et les champs aléatoires de Markov (34). Ces dernières années, la segmentation d'images basées sur les modèles d'apprentissage profond (DL) a donné des améliorations remarquables sur les performances, ce qui a entraîné une naissance d'une nouvelle génération des modèles de segmentation. Dans cette partie, nous entrerons plus en détail dans cette nouvelle génération et les subdivisions qu'ils contiennent, en mentionnant quelques méthodes connues depuis quelques années, ensuite on détaillera les méthodes d'évaluation de segmentation.

4.2.1 Méthodes de segmentation basées sur le Deep Learning

Dans le processus de segmentation par apprentissage profond, il existe trois subdivisions :

Segmentation d'instance

Pour bien comprendre la segmentation d'instance, il est important d'éclaircir la notion de détection d'objet en premier. Dans la tâche de détection, l'algorithme doit prédire avec précision les objets existant dans l'image et les cadres de délimitations correspondantes pour entourer chaque objet détecté par une boîte englobante. Ces derniers sont accompagnés d'un score de confiance qui montre la probabilité qu'un objet appartient à une classe. Alors pour chaque objet la prédiction finale sera une combinaison entre les deux sorties suivantes :

— La position de la boîte englobante :

$$Prdiction = [Xmin', 'Ymin', 'Xmax', 'Ymax']$$

— La classe correspondante avec leurs scores :

$$Classes = ['classe 1', 'classe 2', 'classe N']$$

$$Score = ['score1', 'score2', 'score N']$$

Les valeurs de positions X et Y ci-dessus représentent les coordonnées des points d'un rectangle qui délimite l'objet. On peut remarquer que l'algorithme de détection exécute deux tâches différentes. La sortie de classe et le score exécuté à partir d'une tâche de classification, tandis que les boîtes englobantes viennent d'un processus de régression.

En s'appuyant sur la tâche de détection d'objets, les algorithmes

de segmentation d'instance ajouter un masque binaire à la sortie pour savoir s'il appartient à l'instance prédite ou non. La segmentation d'instance consiste donc à détecter chaque objet individuellement en lui donnant un masque et une étiquette selon leur catégorie.

Segmentation sémantique

La segmentation sémantique consiste à classer chaque pixel de l'image individuellement, et lui donner une étiquette selon leur catégorie (par exemple : voiture, avion, humain, train), il est donc plus difficile que la classification des images connues, qui prédit une seule étiquette pour chaque image. Contrairement à la segmentation d'instance, la segmentation sémantique ne préoccupe pas à identifier les instances, tous les objets de la même classe existent dans l'image sont traités de la même manière comme une seule région, et il n'existe pas un score de confiance. Par exemple s'il y a trois voitures dans une image, la segmentation sémantique classe toutes ces voitures comme une seule instance.

Segmentation panoptique

La segmentation panoptique est une combinaison entre les deux types précédents, elle consiste à classer chaque pixel de l'image et lui donner une étiquette de classe en identifiant à quel numéro d'instance de cette classe il appartient. Les résultats donc constitués de deux valeurs associées pour chaque pixel de l'image :

Prédiction = ['Label', 'Numéro d'instance'].

La figure [4.1](#) montre quelques méthodes de segmentation po-

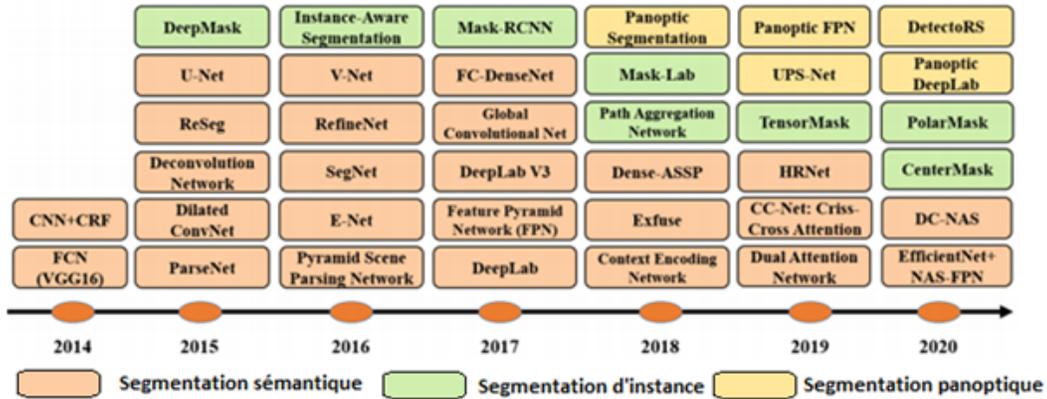


Figure 4.1 – Algorithmes de segmentations basées sur DL pour les images 2D depuis 2014 jusqu’au 2020 (9)

pulaires basées sur DL pour les trois subdivisions motionnées ci-dessus depuis 2014 jusqu’au temps actuel.

4.2.2 Évaluation de segmentation

L’évaluation automatique d’une méthode de segmentation reste toujours indispensable pour permettre d’assurer le bon choix d’un algorithme, de comparer ses performances avec les autres, et d’ajuster ses paramètres.

Dans la littérature les critères d’évaluation peuvent être regroupés en deux catégories selon la disponibilité d’une segmentation référence (vérité-terrain) réaliser par un expert du domaine.

La première est l’évaluation non supervisée, consiste à estimer des critères sans aucune connaissance a priori soit à partir des caractéristiques algorithmiques comme la complexité et la stabilité d’une méthode de segmentation, soit en se basant sur des statistiques calculées sur chaque primitive détectée (région, texture ou contour). La deuxième est l’évaluation supervisée, elle

consiste à évaluer les résultats en utilisant une segmentation référence obtenue manuellement grâce à un ou plusieurs experts du domaine.

Cependant, les performances d'une approche dépendent principalement de la justesse de la vérité-terrain. Jusqu'à présent, il n'existe pas une méthode automatique complète qui peut reproduire l'annotation des images à cause de la grande variabilité des structures anatomiques saine et malade dans les données cliniques. En effet, il existe actuellement une méthode semi-automatique de tracé interactive pour obtenir une segmentation désirée par les experts de domaine. Toutefois, cette solution souffre de plusieurs limites liées à l'expertise des humaines et la variabilité inter expert et intra expert. Pour cela, plusieurs méthodes ont essayé d'éliminer la variabilité entre les experts en définissant une segmentation de références moyenne entre les avis tracés par un groupe d'experts.

En ce qui concerne notre travail, nous avons utilisé les métriques qui mesurent le chevauchement entre la zone segmentée et celle de la vérité-terrain. Ces derniers fournissent des évaluations complètes sur la qualité de la segmentation de l'image. Toutes ces mesures sont obtenues à partir des quatre éléments de la matrice de confusion :

- VP : Vrai Positif : région positifs classés positifs.
- FP : Faux Positif : région négatifs classés positifs.
- FN : Faux Négatif : région positifs classés négatifs.
- VN : Vrai Négatif : région négatifs classés négatifs.

Nous présentons ci-dessous les critères d'évaluation utilisés :

Précision, Rappel, Accuracy et F-score :

- Précision : est le nombre de pixels correctement détecté divisé par le nombre total détecté qui contient les pixels mal classés.

$$Precision = \frac{VP}{VP + FP} \quad (4.1)$$

- Rappel : est le nombre de pixels correctement détecté divisé par le nombre total annoté positivement.

$$Rappel = \frac{VP}{VP + FN} \quad (4.2)$$

- Accuracy : est le pourcentage de pixels qui ont été bien classés.

$$Accuracy = \frac{VP + VN}{VP + VN + FP + FN} \quad (4.3)$$

- F-score : montre la moyenne harmonique de la précision et du rappel. Une valeur élevée de F-score démontre la pertinence de la technique.

$$F1 - score = 2 \cdot \frac{Precision \cdot Rappel}{Precision + Rappel} \quad (4.4)$$

Intersection sur l'union IoU :

L'intersection de l'union est appelée aussi l'indice de Jaccard (JAC), cette métrique mesure le nombre de pixels communs entre la zone prédit et la cible divisée par le nombre total de ces deux derniers.

$$IoU = \frac{prediction \cap cible}{prediction \cup cible} \quad (4.5)$$

Cette mesure est très proche de l'indice de chevauchement "Dice", qui est souvent utilisée comme une fonction de perte pendant la phase d'apprentissage.

$$Dice = 2 \cdot \frac{prediction \cap cible}{|prediction| \cap |cible|} = 2 \cdot \frac{VP}{2VP + FP + FN} = 2 \cdot \frac{JAC}{1 + JAC} \quad (4.6)$$

L'indice de Jaccard (JAC) et le Dice sont moins sensibles au nombre d'objets existents dans l'image, et plus souvent utilisés dans l'évaluation de segmentation d'instance qui produit un groupe de masques de segmentation pour chaque instance détectée dans l'image.

4.3 État de l'art sur la segmentation des images cytologiques

Il existe dans la littérature plusieurs méthodes et techniques classiques et modernes pour la segmentation des images cytologique. La plupart des méthodes classiques basées sur une phase de segmentation pour extraire les caractéristiques de l'image en utilisant différentes techniques puis réalisées une étape de classification. En revanche, les chercheurs en intelligence artificielle commencent ces dernières années à utiliser les réseaux de neurones profonds pour la segmentation automatique des images biomédicale.

Précédemment, la détection et la segmentation automatiques des images cytologiques ont posé des difficultés, principalement en raison de la grande variation entre les caractéristiques des cel-

lules. Alors que certaines méthodes comme (35) ont utilisé une approche optimisée de classification basée sur les pixels, elles ont divisé leur travail en deux étapes. La première consiste à utiliser un classificateur (arbre de décision ou forêt aléatoire) pour effectuer la classification de chaque pixel de l'image en fonction des caractéristiques des espaces de couleur. La deuxième étape est la segmentation qui commence par la détection des points d'intérêt, puis le regroupement des pixels d'une même classe par la méthode de croissance de région.

Suite à ce travail, Saidi et al (36) ont amélioré cet algorithme en ajoutant une étape de sélection des caractéristiques afin de minimiser le temps de calcul et de conserver les mêmes performances de la méthode précédente. Leur méthode de sélection basée sur les méthodes d'ensemble est appelée Ensemble Margin Instance Selection (EMIS).

Un autre travail a été réalisé par (37) pour la segmentation de quatre types de cellules sanguines dans les images cytologiques (noyau, cytoplasme, globule blanc et fond), les auteurs ont proposé une classification supervisée des pixels pour segmenter les cellules. Ils ont utilisé des méthodes de sélection de caractéristiques telles que ReliefF, LDA, RFE, mRMR, puis classifiaient ces caractéristiques par la méthode SVM.

Baghli et al (38) ont proposé une nouvelle méthode de segmentation des cellules sanguines basée sur l'algorithme de segmentation évidentiel (ESA), l'idée principale est d'extraire les composants d'une image cellulaire en utilisant la théorie des évidences. Ces travaux et d'autres comme (39) ont mené des recherches intensives et ont obtenu des améliorations remarquables dans

La segmentation automatique des images cytologiques. Cependant, les caractéristiques des cellules ne sont pas standardisées et diffèrent les unes des autres. Par conséquent, la plupart des approches proposées sont limitées par leur capacité de représentation et sensibles aux grandes variations de forme et de couleur. En outre, ces méthodes séparent la partie extraction des caractéristiques de la classification, ce qui est imparfait en matière de performance et de coût. Ces méthodes semi-automatiques utilisent aussi une classification basée sur les pixels, ce qui les rend très lentes. À cet effet, des méthodes d'apprentissage profond qui peuvent être applicables à de grands ensembles de données offrent une solution potentielle à ces problèmes.

En effet, les réseaux neuronaux profonds, dont la force est leur capacité à représenter des caractéristiques, ont montré une efficacité considérable dans la détection et la segmentation des images (40), (41), (42), (43), (44), (45) (46)). Ces méthodes viennent pour automatiser la partie caractérisation afin qu'elle soit invariante et sensible à tous les changements possibles, tels que la rotation, le déplacement et la déformation. Elles utilisent des réseaux de neurones convolutifs (CNN) pour extraire automatiquement les différentes caractéristiques d'une image à plusieurs niveaux d'abstraction, et leur architecture s'inspire de l'organisation du cortex visuel humain. Au cours de l'étape d'apprentissage, les couches de convolution ajustent automatiquement leur poids initial à l'aide d'un algorithme de rétro-propagation.

Des études récentes ont montré l'efficacité des CNN par rapport aux méthodes conventionnelles dans l'aide au diagnostic. Dans (47), les auteurs ont montré que les CNN donnaient de

meilleurs résultats (71.80, 84.23 %) que les forêts aléatoires (67,53 %, 78,74 %).

Dans le domaine de segmentation des images microscopiques, un nouveau réseau qui a été proposé par (48) qui vise la segmentation et la classification simultanées, ils ont utilisé une carte de distance horizontale et verticale pour séparer les objets groupés. Une autre méthode proposée par (49) qui intègre la détection d'objets dans la segmentation d'instance, les auteurs ont utilisé un réseau neuronal entièrement convolutif FCN pour la localisation, puis forment un autre CNN pour prédire le masque de chaque boîte détectée.

Suivant cette idée basée sur les boîtes, qui a montré son efficacité dans la segmentation selon (50), Les auteurs de (51) ont proposé une nouvelle méthode qui utilise une combinaison du détecteur basé sur les points-clés et la segmentation des cellules individuelles. Ils ont montré une supériorité de la segmentation cellulaire par rapport aux autres méthodes.

Ces méthodes donnent des résultats efficaces grâce à la compréhension globale des objets. Elles permettent de résoudre certains problèmes, comme celui des cellules qui se chevauchent. Cependant, ces méthodes nécessitent une grande masse d'images annotées et un bon choix de poids initiaux pour une meilleure généralisation, ce qui conduirait à une performance de segmentation des instances plus faibles en cas des données insuffisantes. Il convient donc de les améliorer en utilisant certains traitements pour une meilleure qualité de segmentation.

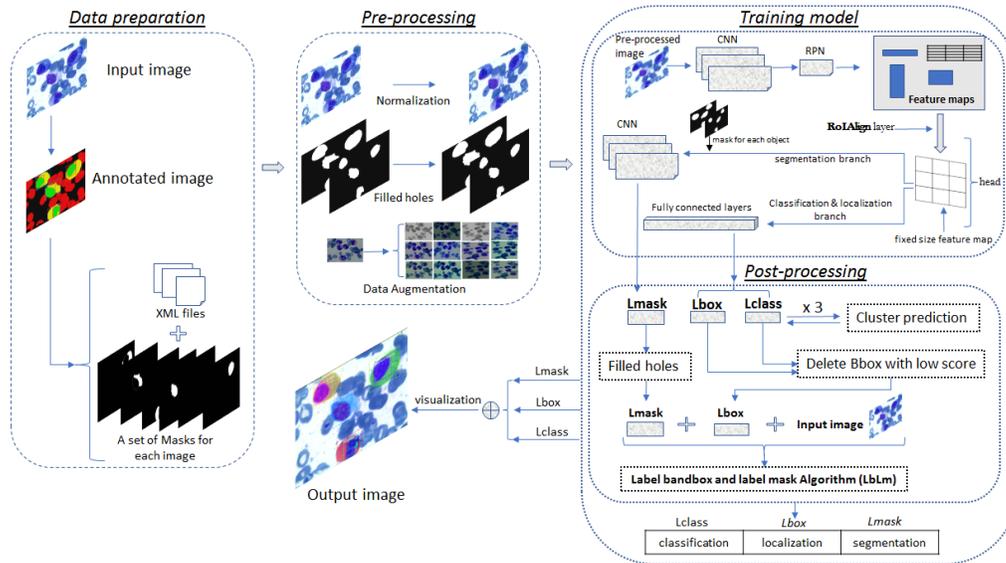


Figure 4.2 – Schéma bloc de l’approche proposée pour la segmentation des instances dans les images cytologiques (10).

4.4 Approche proposée

Dans ce travail de thèse nous proposons une approche efficace pour la segmentation des objets dans les images cytologiques. L’objectif principal est de former un réseau neuronal profond sur notre ensemble de données, afin qu’il reconnaisse les noyaux et le cytoplasme avec précision. Notre approche proposée comporte quatre étapes fondamentales : la préparation des données, le pré-traitement, la formation du modèle et le post-traitement comme indiquée dans la figure 4.2.

Les deux premières étapes contiennent des méthodes et techniques appliquées sur la base de données pour qu’elle soit utilisable par le modèle d’apprentissage, elles sont considérées comme des étapes très importantes et ont un impact direct sur le résultat final. Dans la troisième étape, nous proposons l’amélioration

de l'algorithme du Masque RCNN (52). Le processus d'amélioration a été réalisé en ajustant et en ajoutant des hyper-paramètres en fonction des besoins d'apprentissage. Différentes architectures ont été testées pour choisir la meilleure pour notre ensemble de données. Ensuite, les résultats des modèles ont été améliorés en combinant les prédictions et en appliquant des méthodes de traitement d'images sur les sorties prédites et aux masques. Enfin, un nouvel algorithme qui utilise une ordination entre les sorties a été proposé pour améliorer la qualité de la segmentation.

4.4.1 Environnement de développement

Lorsque nous voulons travailler avec les méthodes d'apprentissage profond, il faut d'abord commencer par préparer et comprendre les outils et les bibliothèques appropriés. Avec la variété des systèmes d'exploitation et les versions des langages de programmation ainsi que les bibliothèques d'apprentissage profond ce n'est pas toujours évident de trouver la combinaison qui s'exécute sans erreurs. La publication de nouvelles bibliothèques et leurs mises-à-jour poussent bien l'innovation grâce aux nouvelles fonctionnalités et les optimisations apportées. Cependant, avec chaque version qui apparaît la compatibilité devient plus compliqué et peut causer des problèmes d'incompatibilités pour les gens qui utilisent des versions antérieures. Donc la configuration d'un environnement de développement dépend souvent des bibliothèques choisies et des versions qui sont compatibles entre elles. Dans ce qui suit, nous présentons le langage de programmation utilisé, ainsi que les principales bibliothèques pour réaliser nos expérimentations.

Python

Dans toutes nos expérimentations nous avons utilisé le langage de programmation python. Python est un langage puissant et facile à apprendre, Il est considéré comme le meilleur langage pour travailler sur les algorithmes d'apprentissage profond. Sa syntaxe simple et compréhensible permet de se focaliser sur le contenu des algorithmes au lieu de passer plus de temps à corriger les erreurs syntaxiques.

Tensorflow et Keras

Tensorflow est une bibliothèque d'apprentissage en profondeur gratuite, qui a été développée par Google. L'utilisation direct de Tensorflow n'est pas toujours nécessaire vu l'existence de l'API Keras (ou tf.keras) qui simplifie et facilite la création et la formation des réseaux profonds.

CUDA et cuDNN

CUDA est une plate-forme de calcul parallèle et un modèle de programmation développés par NVIDIA qui permet d'écrire des programmes et de l'exécutés sur des cartes graphiques GPU. cuDNN est une bibliothèque construite avec CUDA pour les réseaux de neurones profond, elle contient des fonctionnalités accélérées par GPU pour les opérations standards dans les réseaux de neurones profondes telles que la convolution vers l'avant et la rétro-propagation. Dans nos expérimentations les calculs ont été effectués sur une unité de traitement graphique Nvidia GTX 1080 (Nvidia GPU GTX 1080) à travers ces bibliothèques.

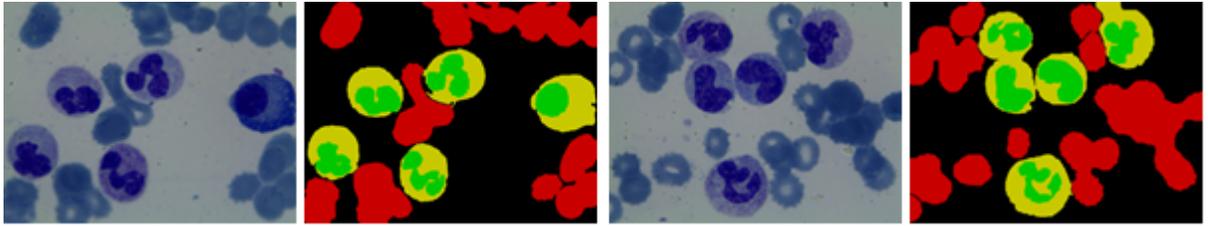


Figure 4.3 – Exemple d’annotation : images originales et annotations correspondantes.

4.4.2 Base de données des frottis sanguins et médullaires

La base de données que nous avons utilisée dans ces expérimentations contient environ 1500 images augmentées. Il existe réellement 87 images en couleurs contenant 155 WBC, proviennent des patients au sein du service d’hémobiologie de l’hôpital universitaire de Tlemcen « CHU Tlemcen » (53).

Le système d’acquisition se compose d’une caméra et microscope avec une haute résolution permet d’acquérir les images couleur RGB de 24 bits et enregistrer sur un disque dur en utilisant un format Bitmap. Les spécialistes de l’hôpital doivent identifier les différents types de cellule existants dans l’image afin de la préparer pour l’apprentissage du modèle et pour évaluer les résultats de la segmentation et de la détection. La figure 4.3 montre deux exemples d’images microscopiques avec leur annotation, effectuée par un expert du domaine, tel que les globules blancs émergent en jaune et les noyaux en vert.

4.4.3 Préparation de jeux de données

Notre travail consiste d'abord à convertir d'une manière semi-automatique toutes les annotations de chaque image réalisées par l'expert du domaine à des masques binaires en utilisant des outils de traitement d'images, ensuite standardisé les images à un format XML et les stocker dans un fichier séparé. Chaque fichier XML contient les caractéristiques de l'image comme le titre, la taille, le chemin, le nombre d'objets existant dans l'image, ces positions et classes correspondantes. Nous avons utilisé ces informations par la suite pour charger les images et leurs masques adéquats avec les classes concernées pour l'apprentissage du réseau.

4.4.4 Pré-traitement des données

Normalisation et remplissage des trous

L'un des plus grands challenge dans la segmentation est la grande variation des couleurs de l'image, causée d'un côté par le processus de coloration, et d'un autre par la qualité du capteur utilisé pendant l'acquisition. Pour améliorer la segmentation des cellules en tenant compte ces variations nous avons effectué une normalisation des couleurs, en soustrayant la moyenne et en divisant par l'écart-type. De cette façon l'image normalisée n'aura pas des problèmes de variation de luminosité et contraste.

D'un autre côté, les masques fournis par la première étape peuvent avoir des vides à l'intérieur, et cela peut produire des erreurs dans l'entraînement des poids du modèle CNN donc il est nécessaire de remplir ces trous en utilisant des opérations morphologiques

(dilatation suivie par érosion).

Augmentation des données

L'insuffisance des données fournies pour l'apprentissage profond constitue une limite pour la création d'un modèle précis et robuste. En effet, le réseau améliore sa précision lorsqu'il reçoit plus de données diverses et de bonne qualité. Dans ce travail nous avons appliqué des transformations aléatoires sur les images d'entrée pour augmenter la base de données. Ce processus permet à notre réseau de mieux s'adapter à toute nouvelle donnée et offre une meilleure généralisation même si la base de données utilisée est petite. Plusieurs types de transformations sur la forme et la couleur ont été effectués pendant la phase d'apprentissage, chaque transformation avait une probabilité de 0 à 1 et il est possible qu'une image puisse subir plus qu'une transformation en même temps. Nous avons utilisé les transformations suivantes pour notre méthode :

- *Retournement (Flip)* : tourner l'image autour de l'axe vertical comme un miroir.
- *Rotation* : tourne l'image autour de centre en utilisant la matrix de transformation T .

$$T = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Avec θ varie entre $[-30,30]$

- *Zoom* : effectue un zoom vers le centre en utilisant la matrice T [figure4].

$$T = \begin{bmatrix} Zx & 0 & 0 \\ 0 & Zy & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Avec Zx, Zy choisi entre [0.8,1]

- *Cisaille (shear)* : effectue la transformation en utilisant la matrix T.

$$T = \begin{bmatrix} 1 & R\sin(\theta) & 0 \\ 0 & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Avec : R pour déterminer la direction de rotation et θ compris entre [-30,30]

- *Transformations sur la couleur* :

Des transformations sur la couleur ont été effectuées en jouant sur le contraste, la luminosité, la saturation, le décalage du canal et la transformation au niveau de gris avec une probabilité de 0 à 0.5. La figure [4.4](#) montre des exemples de transformations sur la couleur.

Transformations en parallèle : Toutes ces transformations ont été appliquées en parallèle avec une probabilité de 0 à 1. Cela permet d'augmenter la possibilité de ne pas avoir la même image deux fois, même si on utilise plus d'itération dans la phase d'apprentissage. La figure [4.5](#) montre des exemples d'une image avec

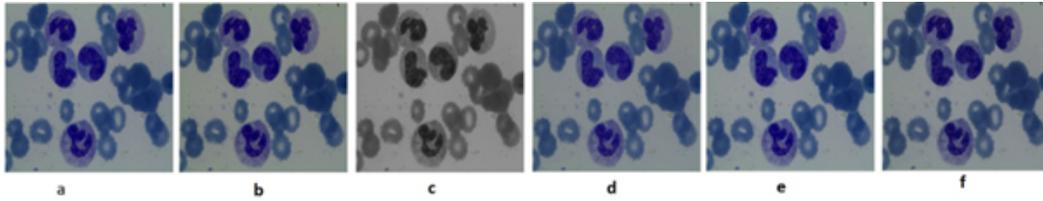


Figure 4.4 – Exemple de transformation sur la couleur. a) Image originale, b) Image avec décalages du canal, c) Image au niveau de gris, d) Image contrastée, e) Image avec Luminosité, f) Image saturé

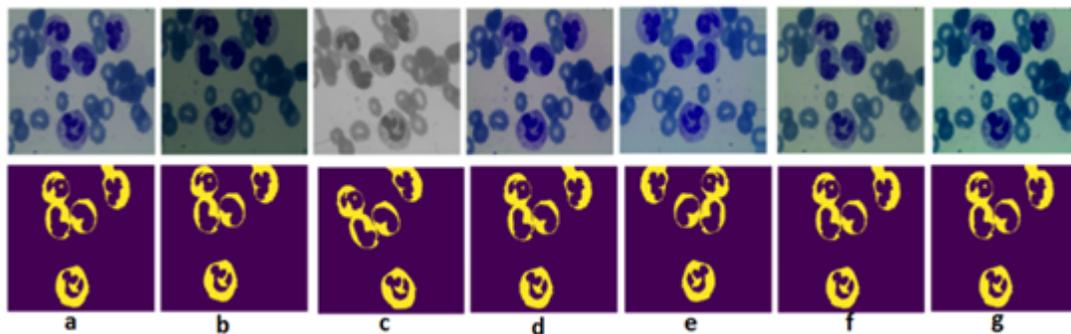


Figure 4.5 – a) Image originale et son masque correspondant. b), c), d), e), f) ,g) des cas de transformations en parallèle

des cas de transformation effectués en parallèle sur la forme et la couleur.

4.4.5 Modèle d'apprentissage

Le modèle proposé dans ce travail se compose de deux étapes, la première appelée RPN (Régional Proposal Network) (52) consiste à proposer une boîte englobant pour chaque objet candidat. La deuxième étape extrait les caractéristiques en utilisant RoIAlign de chaque boîte candidate et effectue une détection et une classification pour chaque instance. Parallèlement de ça, l'algorithme

produit un masque binaire pour chaque région d'intérêt RoI. Le RoIAlign joue un rôle crucial dans la prédiction précise du masque, contrairement au RoIPool (54), la couche RoIAlign propose un alignement correct entre la région d'intérêt RoI et les caractéristiques extraites ce qui conduit à une amélioration significative de précision.

Nous avons utilisé le ResNet50 comme architecture avec des poids initiaux pré entraînés. Plusieurs travaux dans la littérature tel que (55) (56) (57) (44) ont montré que la réutilisation des modèles (ou transfert d'apprentissage) peut être meilleur que l'initialisation aléatoire des poids et peut accélérer l'apprentissage même quand le modèle de base est entraîné sur des problèmes différents. De nombreux modèles formés sur des bases de données connues comme ImageNet ou COCO sont disponibles et peuvent être utilisés pour un apprentissage en utilisant peu de données.

Dans ce travail, l'adaptation des hyperparamètres du modèle joue un rôle crucial dans l'amélioration de la précision de détection et de segmentation, et aide le système à mieux généraliser et évite le problème de sur-apprentissage. Nous avons utilisé une taille réduite dans les régions proposées de la méthode RPN, car la plupart des noyaux et cytoplasmes sont petits et peuvent être trouvés dans n'importe quelle position dans l'image. Nous avons également augmenté leur nombre pendant l'apprentissage et nous avons ajouté d'autres hyperparamètres qui renforcent la capacité à détecter les limites des petits objets.

Pendant l'apprentissage du réseau de neurones convolutif, il peut y arriver un moment où le modèle commencera à sur-apprendre

sur les données d'apprentissage, ce qui entraînera une augmentation dans l'erreur de validation. Ainsi, ce modèle sera incapable de faire de bonnes prédictions sur les nouvelles données (mauvaise généralisation). Nous proposons donc d'utiliser la méthode d'arrêt précoce (Early Stopping). Cette technique de régularisation permet d'arrêter l'apprentissage au moment où la performance sur la base de validation commence à se dégrader.

Comme indiqué dans la figure [4.2](#) la sortie du modèle d'apprentissage est constitué de trois prédictions en parallèle, la première pour la classe (Label Class ou Lclass), la deuxième pour la détection (Label Band box ou Lbbox), et la dernière sortie pour la segmentation (Lmask), elle est constituée d'un masque binaire pour chaque objet détecté. Ces trois sorties seront utilisées par la suite dans le post-traitement (la dernière phase).

4.4.6 Post-traitement

Après la formation du réseau, la majorité des objets dans l'image sont détectées et classées correctement, le gros problème restant est la mauvaise qualité de la segmentation produite par le modèle. Pour cela, nous avons proposé un post traitement sur la sortie du modèle. La première idée consiste à effectuer une combinaison entre la prédiction de l'image source (original) et ces cas de transformation en prenant seulement les points du masque avec un fort chevauchement. Ensuite utiliser une dilata-tion suivie par une érosion pour remplir les petits trous dans les masques prédits. Finalement, nous supprimons les objets détectés qui ont un faible score (moins de 90 % par exemple).

Avec toutes ces améliorations sur la sortie du modèle il est pos-

Algorithm 1 LBLM

INPUT: Original image with a set of predicted bonding box B_i and their masks M_i , Threshold Θ

OUTPUT: A set of masks MASK

```
1: for Each bonding box  $B_i$  predicted by model do
2:   if Result of mask  $M_i$  exist then
3:      $P_c$  = Center of current mask  $M_i$ ;
4:     MASK $_i$  =  $M_i$ 
5:   else
6:      $P_c$  = Center of current bonding box  $B_i$ ;
7:     MASK $_i$  =  $P_c$ 
8:   end if
9:   for Each neighbor pixel  $P_n$  of MASK $_i$  in bonding box
     $B_i$  do
10:    if  $|P_n - P_c| < \Theta$  then
11:      Add current pixel to MASK $_i$ ;
12:    end if
13:  end for
14: end for
```

sible que certains objets n'atteignent pas une bonne qualité de segmentation, cela nous a conduit à proposer une nouvelle méthode appelée LbLm (Label masque and Label Bandbox) pour améliorer la qualité de segmentation. L'idée principale est d'utiliser une combinaison entre les deux sorties du modèle prédit : masque (Label Mask) et la boîte englobante (Label Band box), avec les informations spatiales de l'image d'entrée. À partir du centre d'une boîte englobante candidate ou du masque existant (P_c), nous calculons la variation de couleur entre le pixel central P_c et ses pixels voisins P_n à l'intérieur de la boîte englobante. Si la variation est inférieure à un seuil fixe θ , nous ajoutons ce pixel au nouveau masque. Notre méthode LbLm est décrite en détail dans Algorithme 1 (10).

4.5 Résultats et discussion

Dans nos expérimentations, nous avons sélectionné aléatoirement 70 % de la base pour faire l'apprentissage et laisser le reste pour le test. Nous avons réalisé plusieurs expériences en utilisant différentes architectures (Resnet50, Resnet101, VGG16, VGG19, et Inception V3), les meilleurs résultats ont été obtenus avec l'architecture ResNet50 et l'optimiseur Adam.

Le tableau [4.1](#) montre une comparaison des performances entre le Mask RCNN original et notre approche, il résume les performances de classification, détection et de segmentation de l'ensemble des images d'apprentissage et de test par rapport aux images vérité-terrain pour les deux instances (cytoplasme et noyau).

Table 4.1 – Comparaison entre le Mask RCNN et notre modèle.

		Mask RCNN	Notre approche
Apprentissage	Classification	0.9544	0.9915
	Détection	0.9237	0.9826
	Segmentation	0.8945	0.9630
Test	Classification	0.9219	0.9804
	Détection	0.8603	0.9781
	Segmentation	0.7966	0.9573

Le temps d'exécution dans la phase d'apprentissage et de test varie selon le nombre des régions d'intérêt à traiter par image, il est compris entre 500 et 900 ms/image pour une image de taille 512x512 pixels.

Pour une meilleure évaluation de notre approche proposée, nous l'avons comparée aux méthodes conventionnelles mentionnées

dans (53), (35), (38), (58), (48), (51). La comparaison a été faite sur la même base de données pour la segmentation du noyau et du cytoplasme. Pour la configuration des méthodes basées sur l'apprentissage profond, nous avons utilisé les paramètres par défaut qui ont été mis en œuvre par les auteurs des articles. La comparaison détaillée basée sur la précision et le f-score pour calculer le chevauchement entre la zone segmentée et celle de vérité-terrain est exposée dans le tableau 4.2.

Table 4.2 – Comparaison des performances de la segmentation entre l'approche proposée et l'état de l'art sur les mêmes données

Méthodes	Précision	F-score	Temps/image
Benazzouz et al. (37)	59.000	0.7400	138 s
Baghli et al. (38)	87.360	0.9325	5 s
Settouti et al. (35)	98.135	0.9257	30-60 s
Chen et al. (58)	86.920	0.8316	2-3 s
Graham et al. (48)	87.141	0.8488	2 s
Yi et al. (51)	85.210	0.8925	< 1 s
Notre approche	95.730	0.9706	500-900 ms

Les résultats du Tableau 4.1 montrent l'influence des paramètres utilisés sur les performances du modèle surtout dans la phase de test. Les améliorations dans la phase de post traitement ont clairement amélioré la qualité de la prédiction. Les méthodes d'augmentation ont permis d'éviter le problème de sur-apprentissage et ont fourni au modèle des capacités de généralisation.

Pour plus de visibilité, nous avons choisi aléatoirement des images à partir de notre base de test afin de réaliser une comparaison

visuelle et discuter les performances et la qualité de la segmentation entre l'expert (vérité-terrain), l'algorithme Mask RCNN original, et notre approche proposée (Figure 4.6). Dans la méthode proposée, nous avons utilisé des ancres RPN de petite taille ce qui conduit à une amélioration dans la capacité de détecter les objets de petite taille.

L'avantage de l'approche proposée, est sa puissance de séparer correctement entre les régions qui ont les mêmes propriétés, comme il est démontré dans l'image 1 (figure 4.6). Nous pouvons le voir aussi dans l'image 3 (figure 4.6) que les cellules collées sont bien séparées. Le modèle original ne permettait pas une identification correcte et précise des noyaux et du cytoplasme contrairement au modèle adapté.

Le poste traitement peut fournir aux images initiales des bords nets avec de bons résultats de segmentation, l'idée est d'utiliser un petit nombre de points de départ prédit par le modèle, puis utiliser les informations spatiales pour agrandir la région jusqu'à atteindre un critère d'arrêt. Nous pouvons choisir plusieurs critères en même temps.

Ces résultats nous permettent de valider notre approche proposée et de confirmer la rigueur et la robustesse des méthodes d'apprentissage profond pour la détection des noyaux et du cytoplasme.

En utilisant notre approche sur les données cytologiques, nous avons remarqué certains inconvénients de cet algorithme. L'efficacité de l'algorithme de post traitement dépend de la position initiale de la graine (Seed) prédite par le modèle, ainsi que le

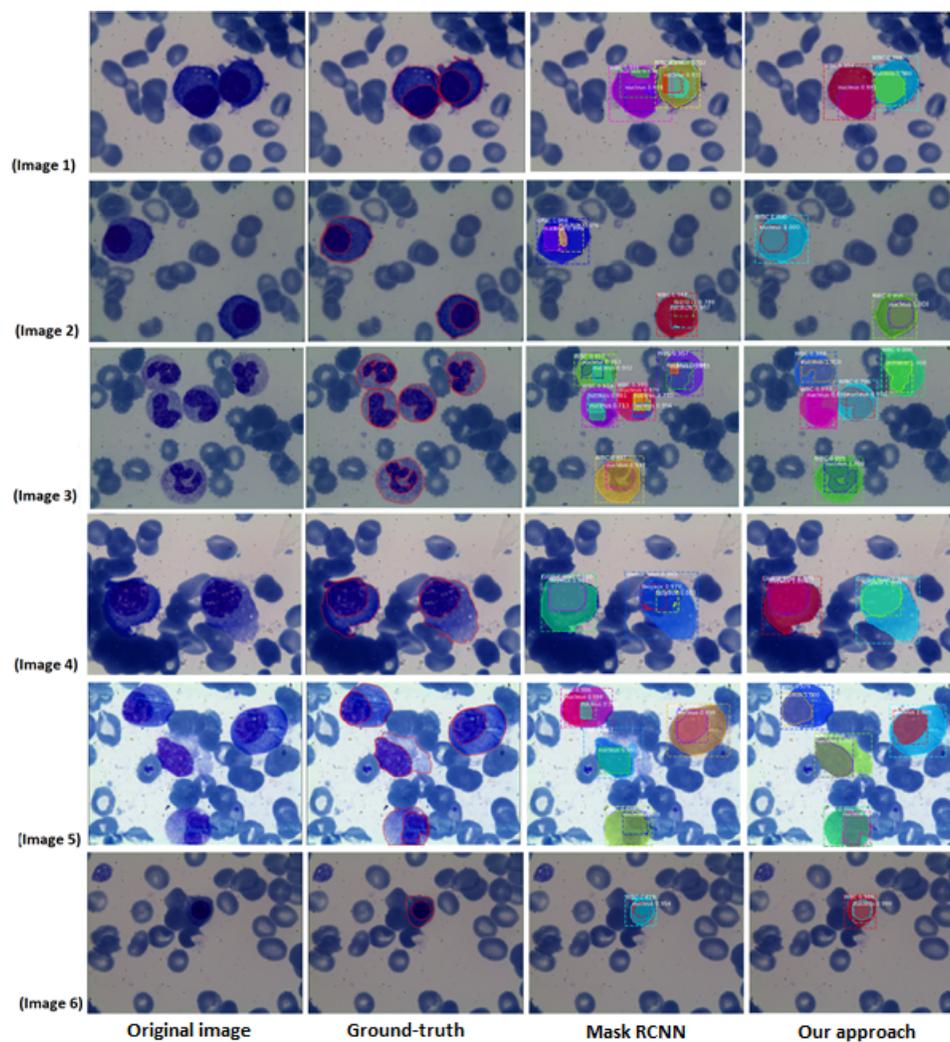


Figure 4.6 – Résultats de la détection, la segmentation, et La classification automatique par Le modèle original et L'approche proposé (10)

choix des critères d'arrêts influe sur la qualité de segmentation, si la valeur du premier critère (seuil) est très petite, l'objet serait sous-segmenté, par contre, si la valeur du seuil est très grande la segmentation de l'image serait dépendue du deuxième critère qui est les limites de la boîte prédite par le modèle. Nous avons facilement résolu certains de ces problèmes en supprimant les pertes qui ont un faible score (moins de 90% par exemple) et par l'utilisation d'un seuil plus au moins grand pour assurer que la segmentation touche les limites de l'objet. En fin, il est à noter que le problème le plus sérieux est que le deuxième critère d'arrêt dépend de la prédiction du modèle, et ce dernier dépend du type, de la nature, et de la quantité de données disponibles. L'une des solutions proposées dans les travaux futurs pour résoudre ce problème, est d'enrichir la base de données en clarifiant le type et la nature de chaque cellule collectée. En fait, il serait très difficile de collecter toutes les cellules existantes en tenant compte de l'agrégation du même nombre de chaque type, ce qui provoquerait un déséquilibre dans la base, et donc ça peut induire à une mauvaise généralisation. Nous suggérons donc dans les travaux futurs d'introduire les méthodes d'équilibrage pour résoudre certains problèmes.

Nous avons comparé nos résultats avec des méthodes de l'état de l'art sur la même base de données [4.2](#). Notre méthode donnait des résultats plus rapides et un F-score meilleur comparé aux autres travaux classiques qui utilisent la caractérisation, ce qui démontre l'efficacité et la rapidité d'apprentissage en profondeur dans la détection des noyaux et du cytoplasme, tandis que le travail de [\(35\)](#) a fournit des résultats plus précis. Cela signifie que

notre modèle a une grande capacité dans la détection de tous les objets pertinents, il est donc plus efficace que les modèles traditionnels, et qui peuvent être très utiles pour les problèmes de diagnostic des maladies sanguines.

L'autre comparaison a été faite avec des méthodes basées sur l'apprentissage profond (48) (58) (51). La première observation du tableau des résultats est que les travaux de Yi et al (51) donnent de meilleurs résultats que (48) (58). Cela peut s'expliquer par le fait que cette méthode intègre la détection d'objets dans la segmentation, elle consiste d'abord à localiser les cellules et ensuite à effectuer la segmentation de chaque objet délimité par les boîtes englobantes, qui permet de séparer les cellules selon les caractéristiques globales mieux que les individuelles de chaque pixel, c'est aussi le cas pour notre approche. La deuxième observation de cette étude est que notre modèle est plus performant que les modèles utilisés dans (48) (58) et (51). Cela peut s'expliquer par le fait que nous avons utilisé le post-traitement (méthode LBLM) qui améliore la qualité de segmentation.

Ce travail présente des résultats importants obtenus grâce à des méthodes d'apprentissage profond. L'efficacité de l'approche proposée pour ce type de base de données, donne une bonne indication qu'elle est applicable et aide les médecins dans l'analyse des images cytologiques. Les appareils qui effectuent l'analyse qualitative des cellules sanguines (automates d'hématologie) disponibles sur le marché sont très coûteux, et sur la base de nos discussions avec les médecins de l'hôpital (CHU de Tlemcen), nous pouvons dire que l'équipement des microscopes disponibles au

service d'hématologie avec notre modèle leur permettra d'avoir des résultats précis dans un temps réduit. L'avantage majeur de notre approche est la grande rapidité de réponse par rapport à d'autres méthodes, nous parlons ici sur quelques millisecondes pour obtenir une réponse fiable et précise, ce qui est parfait pour un expert.

4.6 Conclusion

Dans ce chapitre nous avons présenté notre méthode intelligente qui se base sur l'apprentissage en profondeur pour la reconnaissance automatique des régions noyau et cytoplasme dans les images cytologiques afin d'aider les experts dans le diagnostic médical. L'objectif est de détecter automatiquement chaque objet dans l'image et le classer en tant que noyaux ou cytoplasme, tout en formant un masque binaire pour réaliser la segmentation. Différentes techniques de régularisation, transfert d'apprentissage et d'augmentation des données ont été utilisés afin d'aider le modèle à mieux généraliser et éviter le problème du sur-apprentissage tout en minimisant le temps d'apprentissage. La principale contribution est l'utilisation d'une combinaison coopérative entre les sorties du modèle dans le but d'augmenter la qualité de segmentation. L'architecture du modèle a été correctement adaptée pour détecter avec succès toutes les cellules de l'image d'entrée. Pour améliorer les résultats, nous avons proposé un nouvel algorithme dans lequel une coopération entre les sorties du modèle et les informations spatiales de l'image d'entrée a été utilisée. Les résultats obtenus ont été très prometteurs

et encourageants, et montrent une amélioration importante de la précision et du temps de calcul par rapport aux modèles précédents.

Ce travail nous a permis d'ouvrir une piste de recherche dans le domaine de reconnaissance automatique des données microscopiques par les méthodes d'apprentissage profond. Nous travaillons actuellement dans le cadre d'un projet PRFU sous deux directions. Tout d'abord, collecter de nouvelles images cytologiques et les étiqueter (en profitant de l'expertise des médecins hématologues) pour définir clairement les cas pathologiques pour la maladie de la leucémie. Deuxièmement, mettre en œuvre une approche qui se base sur l'apprentissage profond dans un système d'aide au diagnostic pour aider les médecins à compter les différents types de cellule, et de détecter précisément les cas pathologique.

5 Méthode réalisée pour la classification des sous-types des Leucocytes

5.1 Introduction

L'analyse des images microscopiques sanguines est très importante pour le diagnostic et le traitement de nombreuses maladies. Les globules blancs (WBC) sont des cellules sanguines qui font partie du système immunitaire du corps humain et qui jouent un rôle essentiel dans la lutte contre les bactéries, les virus, et les microbes. Leurs nombres chez un adulte humain normal variaient de 4000 à 10 000 globules blancs. Par conséquent, si le nombre des globules blancs se situe en dehors de l'intervalle de référence normal (excès ou faiblesse), il peut nous conduire à déterminer plusieurs maladies telles que la leucémie, les maladies liées à l'infection, l'insuffisance de la moelle osseuse, etc.

Les globules blancs sont classés en cinq sous-types : Neutrophile, Éosinophiles, Basophile, Lymphocytes et Monocytes. Lorsque le nombre des sous-types est en dehors de l'intervalle de référence normale, il peut nous donner une indication sur les maladies dont le corps souffre (59), (60). Actuellement, dans le service d'hématologie au CHU de Tlemcen, les médecins ana-

lysent manuellement les frottis sanguins. Ce calcul manuel des types de globules blancs dans les images microscopiques est une tâche fastidieuse pour un spécialiste. Pour effectuer un diagnostic de la maladie, il est important de connaître l'emplacement des cellules dans le microscope, puis réaliser un comptage manuel de chaque sous-type individuellement sur une population de 200 éléments au moyenne. Il s'agit donc d'un processus long et sujet aux erreurs ainsi qu'il nécessite une grande expérience dans le domaine de la cytologie.

Actuellement, plusieurs recherches ont été menées pour la classification automatique des globules blancs à l'aide d'un système automatique qui permet de gagner plus de temps et de réduire le taux d'erreurs humaines pour effectuer un diagnostic médical fiable. Ces dernières années, les réseaux de neurones profonds ont montré leurs grandes efficacités dans plusieurs domaines tels que l'analyse d'image, de son, de la parole et le diagnostic médical. Depuis leurs succès dans la compétition ImageNet en 2012, la majorité des chercheurs ont commencé à travailler avec les architectures CNN et les techniques d'apprentissage profond pour la classification des données.

Dans ce travail de thèse, une nouvelle méthode basée sur les techniques d'apprentissage profond a été proposée pour la classification des sous-types dans les images microscopiques. Dans ce qui suit, nous présentons tout d'abord un état de l'art sur les méthodes de classification des cellules, ensuite, nous détaillons l'approche proposée et les techniques utilisés ainsi que nos résul-

tats et une discussion.

5.2 État de l'art sur la classification des cellules

Plusieurs méthodes dans la littérature ont été menées pour la classification des images microscopiques selon le sous-type des globules blancs. La méthode proposée dans ce travail de thèse a été comparée avec d'autres études dans la littérature sur le même ensemble de données.

En 2018, les auteurs de (61) ont utilisé un ensemble d'algorithmes d'apprentissage semi-supervisé pour la classification des sous-types de cellules sanguines. L'efficacité de l'algorithme présenté dans cette étude a été démontrée par une série d'expériences indiquant la possibilité de développer des modèles de prédiction fiables et robustes par l'adaptation des techniques d'ensembles dans le cadre d'apprentissage semi-supervisé. Ils ont obtenu une précision de 93.29% avec la technique d'apprentissage semi-supervisé SSL et le classifieur KNN.

Bani-Hani et al. (62) ont utilisé l'algorithme génétique (GA) pour optimiser les hyperparamètres du modèle CNN, ils ont obtenu une précision de classification de 91% sur les données de test.

Dans un autre travail sur la même base de données, les auteurs de (63) ont utilisé une combinaison entre les modèles d'appren-

tissage profonds (CNN) et les réseaux de neurones récurrents (RNN) pour classer les types des cellules. Ils ont testé plusieurs architectures profondes avec des poids pré-entraînées sur la base de données ImageNet. Le modèle proposé dans cette étude a atteint une précision de 90.79% sur la base de test.

En 2019, Banik et al. (64) ont proposé une architecture CNN pour la classification des images de la même base de données. Ils ont proposé un modèle avec cinq couches de convolution, trois couches de max-pooling et un réseau entièrement connecté de trois couches. Ils ont fusionné les sorties de deux couches de convolution en utilisant l'opération max-pooling pour les injecter en entrée de la dernière couche entièrement connectée. La précision moyenne obtenue est de 90.79% sur la base de test.

Dans le travail de (65) les auteurs ont utilisé une combinaison des caractéristiques extraites à partir de plusieurs architectures profondes à savoir AlexNet, VGG, GoogleNet, et ResNet. Ensuite, ils ont appliqué la méthode de la Redondance Minimale Pertinence Maximale pour sélectionner les caractéristiques efficaces avant de réaliser la classification en utilisant le classifieur ELM (Extreme Learning Machine). Ils ont obtenu une précision de 96.03% sur les données de test.

5.3 Approche proposée pour la classification

Dans ce travail de thèse, nous proposons une nouvelle méthode d'apprentissage profond pour la classification des cellules dans les images microscopiques. Nous utilisons cinq architectures CNN profondes pour l'extraction des caractéristiques. Ces architectures CNN sont pré-entraînées sur l'ensemble de données ImageNet pour des images génériques de la nature et qui peuvent être utilisées pour extraire les caractéristiques d'images biomédicales sans avoir à entraîner le modèle à partir de zéro en utilisant le transfert d'apprentissage (66). Pendant la phase d'apprentissage, nous appliquons différentes méthodes de régularisation pour éviter le problème de sur-apprentissage, à savoir la régularisation L2, l'augmentation des données, dropout et l'arrêt précoce. Après la phase d'apprentissage des caractéristiques, nous combinons les prédictions des cinq architectures CNN par un vote majoritaire pour former une seule prédiction. La figure 5.1 montre la conception du modèle proposé dans ce travail.

5.4 Base de données des leucocytes

Dans nos expérimentations nous utilisons la base de données BCCD qui a été collectée par (67). Elle se compose de quatre types de cellules : éosinophile, lymphocyte, monocyte, et neutrophile séparés sur deux ensembles d'images. Le premier contient 367 images originales (sans augmentation) avec une taille de

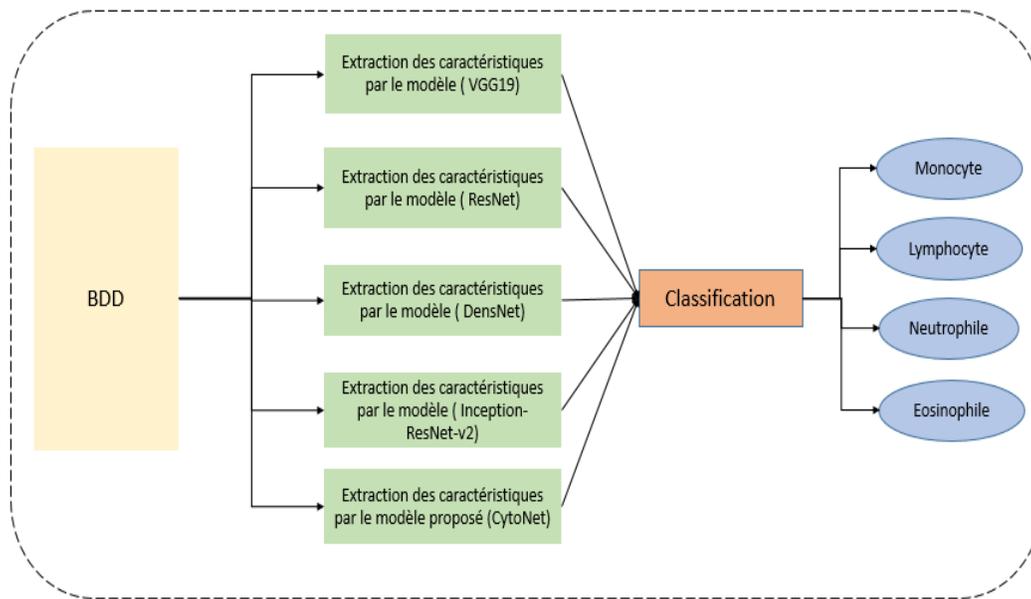


Figure 5.1 – Modèle proposé pour la classification des cellules

640x480. Ces images ont été acquises en utilisant un microscope avec un zoom de 100x liée à une caméra couleur CCD analogique (68). Le deuxième ensemble de données est composé de 12 444 images augmentées, chaque image a une taille de 320x240 pixels sauvegardés sous format JPEG.

L'augmentation a été réalisée par des transformations aléatoires comme la rotation, le décalage horizontal et vertical, le retournement horizontal etc. La figure 5.2 montre la distribution des images avant et après l'augmentation. L'ensemble de données est divisé sur quatre sous-ensembles répartis selon les types de WBC : éosinophile, neutrophile, lymphocyte, et monocyte.

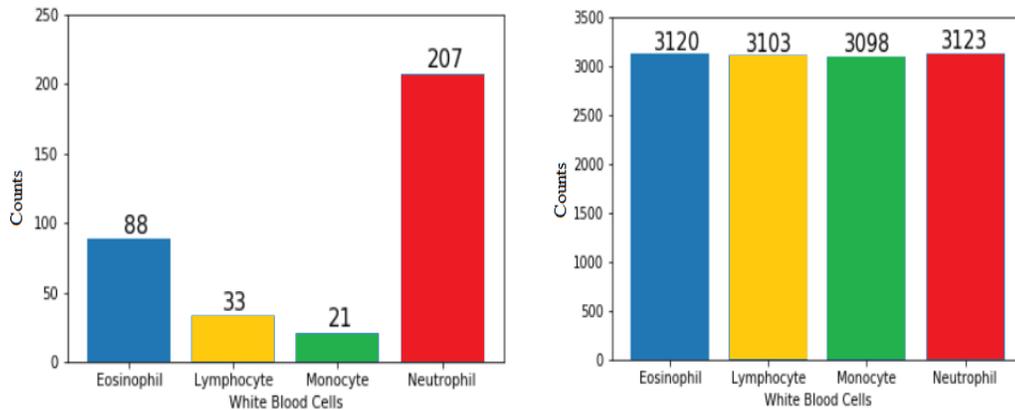


Figure 5.2 – Distribution des images de la base de données avant et après l'augmentation

5.5 Modèle d'apprentissage et paramétrage

5.5.1 Méthodes d'optimisation

Comme nous avons décrit dans la section [3.4.4](#), l'objectif des méthodes d'optimisation est de mettre à jour les poids du réseau à chaque itération jusqu'à atteindre une erreur d'apprentissage minimal. Les stratégies de modification des poids se diffèrent d'un algorithme à un autre, et la mise à jour des poids s'effectue par plusieurs méthodes.

Dans la méthode de descente de gradient stochastique SGD ([69](#)) ([70](#)), la modification est effectuée itérativement pour chaque exemple dans la base d'apprentissage. Pour cette raison, l'erreur d'apprentissage converge lentement vers le minimum. Par contre, dans la méthode d'optimiseur Adam (Adaptative Moment Esti-

mation) Le pas d'apprentissage se calcule automatiquement ce qui implique une convergence plus rapide mais avec plus d'oscillations de l'erreur qu'une descente de gradient. Donc, il est plus pratique d'utiliser un compromis en commençant les premières époques d'apprentissage par l'optimiseur Adam pour se rapprocher plus de minimum, puis appliquer l'optimiseur SGD qui peut trouver des meilleures solutions si la durée d'apprentissage est beaucoup plus long (71).

5.5.2 Modèle d'extraction des caractéristiques

Dans ce travail, nous avons utilisé plusieurs architectures profondes pour l'extraction des caractéristiques à partir des images d'entrées.

Le modèle VGG-19 (12) a une structure profonde de 19 couches, les images utilisées par cette architecture sont tout d'abord redimensionnées à une taille de 224x224 pixels avant de les injecter dans le modèle VGG-19. Pour les autres paramètres du réseau VGG-19, les valeurs par défaut sont conservées.

ResNet ou résiduel network (8) est le modèle qui a obtenu les meilleures performances lors de la compétition ImageNet en 2015. Le fait que ce modèle est doté des connexions résiduelles, présenté dans la section 3.4.6, lui permet de construire des architectures plus profondes allant de 50 jusqu'à 200 couches. Dans nos expérimentations, nous utilisons une profondeur de 50 couches, et les tailles des images sont redimensionnées en 224x224 pixels par l'architecture ResNet.

L'architecture Inception (6), appelée également GoogLeNet, permet d'extraire les caractéristiques de l'image à plusieurs niveaux (voir la section 3.4.6). Il existe plusieurs versions de cette architecture publiées par Google ; dans nos expérimentations nous utilisons la version Inception-V3.

Le modèle Inception-ResNet-v2 (14) combine la structure Inception et le principe des connexions résiduelle dans l'architecture ResNet. La combinaison entre ces deux techniques permet de réduire le temps d'apprentissage et d'éviter le problème de dégradation liée aux structures profondes.

Pour notre architecture CNN proposée que nous avons appelée CytoNet, nous construirons un réseau avec 4 couches de convolution (Conv2D), chaque couche est suivie par une couche de mise en commun (max-pooling) et enfin trois couches entièrement connecte FC. La figure 5.3 résume le modèle CNN proposé. Le nombre de filtre pour chaque couche est de 32, 64, 128 et 256 respectivement pour les quatre couches de modèle avec une taille de 3x3, un padding de 0, et un stride de 2x2. Les tailles des filtres ainsi que leurs nombres sont choisis empiriquement en prenant en compte le temps de calcul dans les deux phases d'apprentissage et de test, ainsi que les performances obtenues. Pour les dernières couches entièrement connectées, nous utilisons 64 neurones dans la première couche cachée, 128 neurones dans la deuxième, et 64 neurones pour la troisième. Dans la couche de sortie nous utilisons quatre neurones selon le nombre des classes

à prédire.

```
Model: "sequential"
-----
Layer (type)                Output Shape                Param #
-----
conv2d (Conv2D)             (None, 318, 238, 32)      896
-----
max_pooling2d (MaxPooling2D) (None, 159, 119, 32)      0
-----
dropout (Dropout)           (None, 159, 119, 32)      0
-----
conv2d_1 (Conv2D)           (None, 157, 117, 64)     18496
-----
max_pooling2d_1 (MaxPooling2 (None, 78, 58, 64)        0
-----
dropout_1 (Dropout)         (None, 78, 58, 64)        0
-----
conv2d_2 (Conv2D)           (None, 76, 56, 128)     73856
-----
max_pooling2d_2 (MaxPooling2 (None, 38, 28, 128)       0
-----
dropout_2 (Dropout)         (None, 38, 28, 128)       0
-----
conv2d_3 (Conv2D)           (None, 36, 26, 256)    295168
-----
max_pooling2d_3 (MaxPooling2 (None, 18, 13, 256)       0
-----
dropout_3 (Dropout)         (None, 18, 13, 256)       0
-----
flatten (Flatten)           (None, 59904)             0
-----
dense (Dense)                (None, 64)                3833920
-----
dense_1 (Dense)              (None, 128)               8320
-----
dense_2 (Dense)              (None, 64)                8256
-----
dense_3 (Dense)              (None, 4)                 260
-----
Total params: 4,239,172
Trainable params: 4,239,172
Non-trainable params: 0
```

Figure 5.3 – Résumé du modèle proposé

Après chaque couche de mise en commun nous utilisons une couche Dropout avec une probabilité d'abandon de 0.4, pour éliminer un nombre aléatoire des neurones pendant la phase d'apprentissage. La fonction d'activation ReLu est appliquée sur toutes les couches de convolution et les couches entièrement connectées. Pour la couche de sortie nous appliquons la fonction d'activation softmax.

5.5.3 Régularisation

Pour bien contrôler la capacité de notre modèle à mieux généraliser et éviter le problème de sur-apprentissage, nous avons appliqué les méthodes de régularisation. Il existe différentes stratégies de régularisation en Deep Learning (voir la section [3.4.5](#)), dans nos expérimentations nous avons utilisé les techniques de régularisation L2, l'augmentation des données, Dropout et l'arrêt précoce.

Dans la régularisation L2, une pénalité est ajoutée à la fonction de coût pour qu'elle ne devienne pas proche de zéro, c'est-à-dire que chaque fois notre modèle essaie d'être parfait sur les données d'apprentissage, un terme de régularisation pénalise le sur-apprentissage sur ces données. La régularisation L2 est l'une des stratégies les plus utilisées dans l'apprentissage artificiel pour former des modèles performants et capables de bien généraliser. Elle donne de meilleurs résultats par rapport à la régularisation L1 dans la plupart des tests.

Les modèles d'apprentissage profond contiennent un très grand nombre de paramètres qui lui donne une grande capacité de représentation. Mais, dans le cas où le nombre de données d'apprentissage est petit, le modèle peut être confronté à un sur-apprentissage. Pour cette raison, nous avons utilisé l'augmentation des données pour produire des nouvelles données à partir des exemples existants en appliquant des transformations aléatoires comme la rotation, le zoom, le retournement, etc.

La technique de régularisation Dropout est l'une des méthodes les plus utilisées dans la construction des architectures de Deep Learning. L'application du dropout dans un réseau de neurones profond permet de supprimer aléatoirement des nœuds. Une valeur de probabilité de décrochage entre 0 et 1 doit être fixée pour indiquer la probabilité de suppression à chaque itération. Cette technique permet de rendre notre réseau plus simple et de produire des prédictions à partir des nœuds disponibles uniquement pour gagner plus de temps dans chaque itération.

L'arrêt précoce (ou Early Stopping en anglais) est une technique de régularisation très utile pendant la phase d'apprentissage. Elle permet d'arrêter l'apprentissage si le modèle commence à sur-apprendre sur les données d'apprentissage en surveillant l'erreur de validation. Un hyperparamètre appelé patience doit être fixé pour indiquer le nombre d'itérations qu'on doit attendre pour arrêter l'apprentissage s'il n'y a pas une amélioration dans l'erreur de validation.

5.6 Résultats et discussion

Dans nos expérimentations nous avons utilisé le langage de programmation python et les bibliothèques Tensorflow et Keras. Les valeurs des paramètres des quatre autres architectures utilisées sont les valeurs par défaut et les calculs ont été exécutés sur une carte graphique GPU (NVIDIA GTX 1080).

La figure 5.4 montre la courbe de précision d'apprentissage et de test de notre architecture proposée (CytoNet).

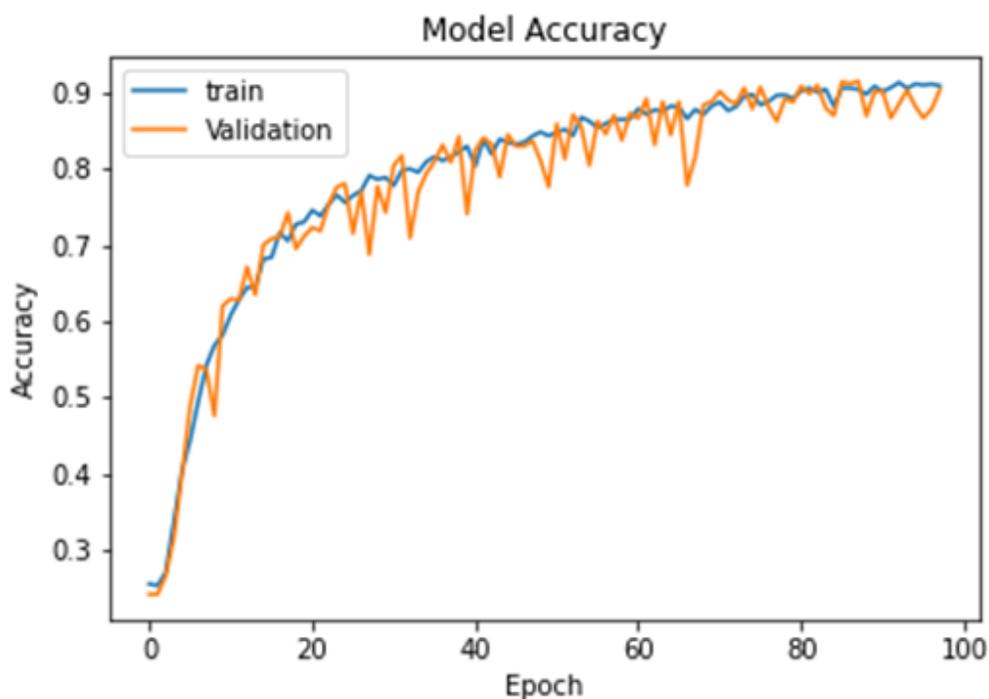


Figure 5.4 – Courbe de précision d'apprentissage et de validation de notre modèle CNN proposé (CytoNet).

Nous pouvons voir que la courbe en orange qui représente la précision de validation suit celle qui est en bleu et qui représente la précision d'apprentissage, cela indique l'absence de sur-

apprentissage. Dans la 88^{ième} époque la précision de validation atteindre son maximum (91.57%). Cela peut s'expliquer par le fait que les méthodes de régularisation rendent le modèle stable et permettent d'éviter le sur-apprentissage.

A partir de la matrice de confusion de chaque modèle dans la figure 5.5, nous pouvons évaluer leurs qualités en calculant la précision, le rappel et le F1-score des prédictions sur la base de test en utilisant les trois équations suivantes :

$$Precision = \frac{VP}{VP + FP} \quad (5.1)$$

$$Rappel = \frac{VP}{VP + FN} \quad (5.2)$$

$$F1 - score = 2 * \frac{Precision * Rappel}{Precision + Rappel} \quad (5.3)$$

Le tableau 5.1 montre les résultats obtenus par les différents modèles. Nous pouvons voir que la méthode de combinaison proposée a surpassé les autres architectures en terme de précision, de rappel, et de F-score. La précision de validation a atteint 96.5% sur 2478 images, tandis que le rappel était de 96.8%. Nous pouvons donc déduire que le modèle proposé a une grande capacité à identifier les faux négatifs ainsi que les cellules qui sont mal classées par les autres modèles.

Le tableau 5.2 présente une analyse comparative de la méthode proposée qui combine les prédictions des cinq architectures avec d'autres méthodes de la littérature sur la même base de données.

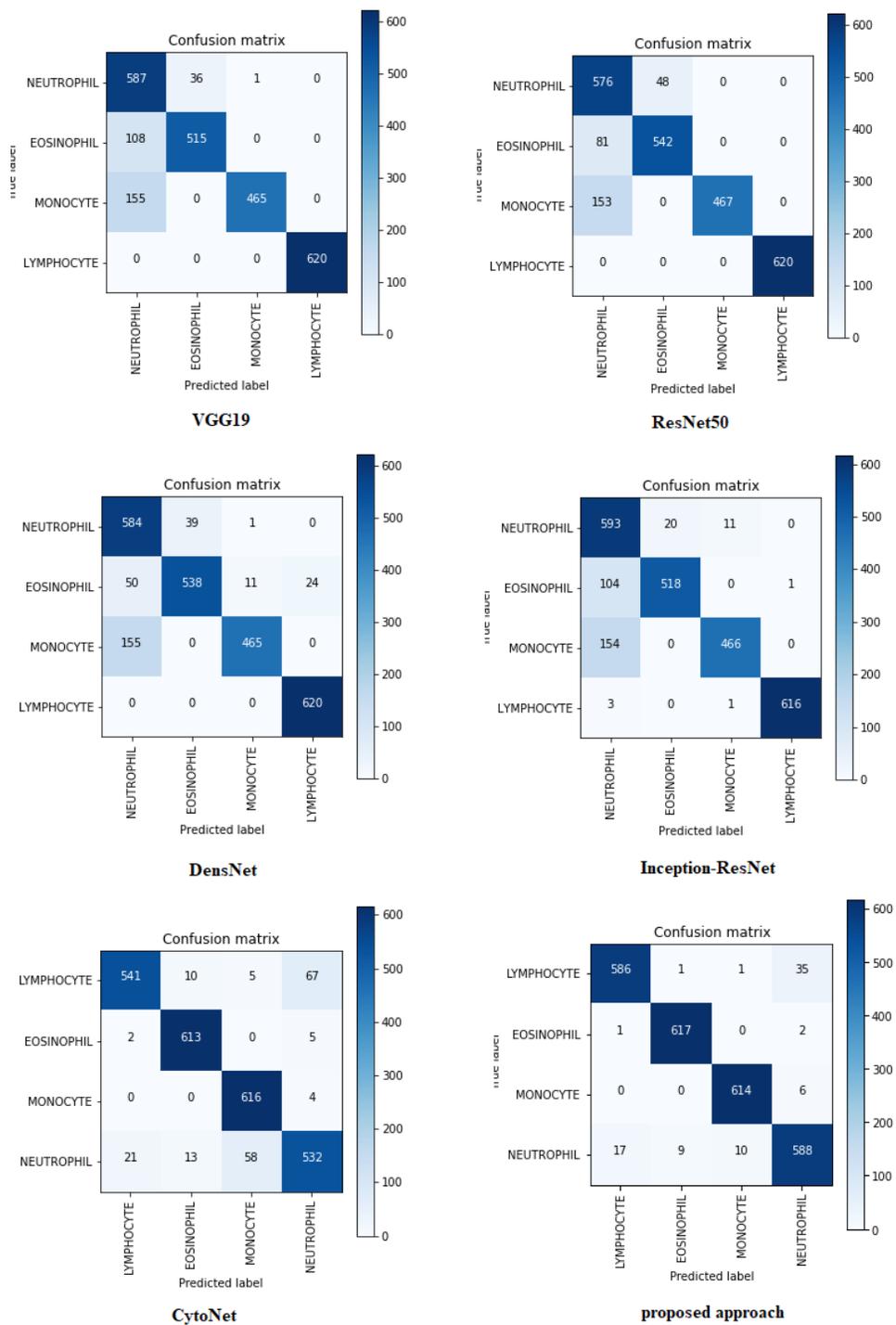


Figure 5.5 – Matrice de confusion de différent modèles

Nous pouvons voir que notre méthode proposée donne une précision plus élevée dans la classification des images WBC.

Table 5.1 – Précision, Rappel et F1-score sur les données de test

Modèle	Type	Neutro	Eosino	Mono	Lympho	Moyenne
VGG19	Précision	0.69	0.93	1.00	1.00	0.91
	Rappel	0.94	0.83	0.75	1.00	0.88
	F-score	0.80	0.88	0.86	1.00	0.88
ResNet50	Précision	0.71	0.92	1.00	1.00	0.91
	Rappel	0.92	0.87	0.75	1.00	0.89
	F-score	0.80	0.89	0.86	1.00	0.89
DensNet	Précision	0.74	0.93	0.97	0.96	0.90
	Rappel	0.94	0.86	0.75	1.00	0.89
	F-score	0.83	0.90	0.85	0.98	0.89
Inception-ResNet	Précision	0.69	0.96	0.97	1.00	0.91
	Rappel	0.95	0.83	0.75	0.99	0.88
	F-score	0.80	0.89	0.85	1.00	0.88
CytoNet	Précision	0.88	0.96	0.91	0.96	0.93
	Rappel	0.85	0.99	0.99	0.87	0.93
	F-score	0.86	0.99	0.95	0.91	0.92
Notre approche	Précision	0.93	0.98	0.98	0.97	0.965
	Rappel	0.94	1.00	0.99	0.94	0.968
	F-score	0.94	0.99	0.99	0.96	0.97

5.7 Conclusion

Dans ce chapitre nous avons présenté notre approche pour la classification des quatre sous-types de leucocytes dans les

Méthodes	Précision
Livieris et al. 2018 (61)	93.29
Bani-Hani et al. 2018 (62)	91.00
Liang et al. 2018 (63)	90.79
Banik et al. 2019 (64)	90.79
Fatih et al. 2020 (65)	96.03
Notre approche	96.50

Table 5.2 – Comparaison avec d'autres méthodes sur la même base de données

images microscopiques. La méthode que nous avons proposée utilise quatre architectures CNN pour l'extraction des caractéristiques à partir des images d'entrée, à savoir VGG19, ResNet50, DensNet, InceptionResNet ainsi qu'une autre nouvelle architecture proposée que nous avons appelée CytoNet. Plusieurs techniques de régularisation ont été appliquées afin de contrôler le sur-apprentissage tel que l'arrêt précoce, l'augmentation des données, dropout, et la régularisation L2.

Nos résultats expérimentaux ont montré l'efficacité de l'approche proposée qui utilise une prédiction combinée entre les cinq architectures par rapport à chaque modèle individuellement.

Dans un travail futur, nous souhaitons utiliser des méthodes de sélection des caractéristiques pour optimiser l'efficacité et les performances du modèle ainsi que leur coût de calcul. En outre, un aspect intéressant consiste à utiliser d'autres types de données médicales numériques avec les images cytologiques pour effectuer une classification multimodale qui peut être fiable dans un diagnostic médical.

6 Conclusion générale

L'analyse microscopique des frottis du sang ou de la moelle osseuse joue un rôle primordial dans les procédures de diagnostic et de contrôle de plusieurs maladies cancéreuses. Cependant, la reconnaissance manuelle des cellules reste une tâche très difficile qui nécessite beaucoup de temps et d'expérience pour discriminer et analyser les différents éléments. Par conséquent, un système de segmentation et de classification automatique est nécessaire pour aider les médecins et rendre le processus de diagnostic à la fois rapide et précis.

L'objectif de cette thèse était de proposer un système informatique efficace et robuste pour la reconnaissance automatique des cellules dans les images cytologiques. cela passait premièrement par un apprentissage pour maîtriser le domaine scientifique, puis l'apprentissage et la maîtrise des principes et des techniques d'apprentissage profond. Ensuite une familiarisation avec les outils nécessaires pour développer nos modèles tel que les bibliothèques logicielles d'apprentissage profond.

Au cours de cette thèse, nous avons réalisé deux contributions principales qui se basent sur les techniques d'intelligence artificielle et en particulier l'apprentissage profond, pour la reconnaissance automatique des cellules dans les images microscopiques afin d'aider les experts dans le diagnostic médical.

Dans la première contribution, l'objectif principal était de dé-

tecter automatiquement tous les globules blancs dans l'image et de classer leurs composantes en tant que noyaux ou cytoplasmes, tout en formant un masque binaire pour segmenter les instances détectées. Nous avons entraîné un modèle neuronal profond (Mask R-CNN) sur l'ensemble de données cytologiques. Plusieurs techniques de régularisation ont été appliquées pour éviter le problème de sur apprentissage.

Ce réseau nous a permis d'obtenir de bons résultats de détection, mais la mauvaise qualité de segmentation produite par ce modèle nous a conduit à réaliser des améliorations sur les sorties. La première idée consiste à utiliser une prédiction combinée entre une image originale et ces cas de transformations en sélectionnant seulement les points prédits avec un fort chevauchement. Ensuite nous avons appliqué les opérations morphologique érosion suivi par dilatation pour remplir les vides dans les masques résultants, et finalement nous avons sélectionné seulement les objets détectés avec un score élevé. Ces améliorations nous ont donnés de bons résultats, mais certains objets n'étaient toujours pas bien segmentés, ce qui nous a conduit à proposer notre nouvel algorithme LBLM qui utilise une combinaison coopérative entre les résultats du modèle et les informations spatiales de l'image d'entrée.

Les résultats obtenus ont été très prometteurs et encourageants, et montrent la puissance et la rapidité des méthodes d'apprentissage profond dans la tâche de reconnaissance des cellules, ce qui est parfait pour un système d'aide au diagnostic médical.

La deuxième contribution réalisée dans cette thèse concerne

L'identification des globules blancs où une nouvelle méthode a été proposée pour la classification de quatre sous-types de cellules (neutrophile, éosinophile, monocyte et lymphocyte). Nous avons utilisé cinq modèles profonds pour l'extraction des caractéristiques à partir des images cytologiques. plusieurs techniques de régularisations ont été appliqués pour contrôler le sur-apprentissage et réduire le temps d'exécution. Cette approche de classification cellulaire nous a permis d'obtenir de bons résultats par rapport à chaque modèle individuellement.

Les travaux futurs viseront à améliorer notre système de reconnaissance automatique, plusieurs pistes de recherche restent inexploitées. Généralement, Les bases de données est l'un des éléments les plus importants pour l'amélioration d'un système d'apprentissage profond. En effet, il serait très difficile de collecter une grande base de données en prenant en considération la proportion équitable de chaque type dans la base, ce qui provoque un déséquilibre qui peut influencer la phase d'apprentissage des caractéristiques. Pour cette raison, L'utilisation des techniques d'équilibrage pourrait résoudre certains problèmes.

Le choix des hyperparamètres optimaux d'un système profond se fait empiriquement et dépend principalement des types de données, les objectifs et la nature du problème à résoudre. En effet, même avec une expérience dans le domaine, la tâche d'optimisation manuelle des hyperparamètres peut prendre beaucoup de temps. Il n'existe pas un choix fixe qui convient à tous les modèles et les problèmes, Les procédures d'optimisation peuvent être efficaces pour détecter automatiquement un ensemble op-

timal d'hyperparamètres pour un modèle donné. En outre, les méthodes d'optimisation et de sélection peuvent être également utiliser pour réduire le nombre des caractéristiques extraites par les modèles d'apprentissage profond afin de minimiser leur coût de calcul et améliorer les performances et l'efficacité du traitement.

Un autre aspect très important pour améliorer le système d'aide au diagnostic médical consiste à utiliser d'autre type de données numériques avec les données spatiales (images) pour effectuer une prédiction multimodale qui peut être fiable dans un diagnostic médical.

Ce travail de thèse fait partie des recherches importantes dans le domaine du traitement d'images médicales qui nécessite plus d'interaction entre les chercheurs et les cliniciens pour apporter plus d'avancées dans les systèmes d'aide au diagnostic, non seulement pour faciliter la tâche au praticien mais aussi pour être utilisés pour former les nouvelles générations des médecins.

Bibliographie

- [1] M. Benazzouz, analyse intelligente des images médicales : application aux images microscopiques de cytologie, Ph.D. thesis, Tlemcen university (2014).
- [2] I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, Deep learning, Vol. 1, MIT press Cambridge, 2016.
- [3] V. Dumoulin, F. Visin, A guide to convolution arithmetic for deep learning, arXiv preprint arXiv :1603.07285.
- [4] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (11) (1998) 2278–2324.
- [5] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in neural information processing systems 25 (2012) 1097–1105.
- [6] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in : Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1–9.
- [7] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, J. Garcia-Rodriguez, A review on deep learning techniques applied to semantic segmentation (2017). [arXiv:1704.06857](https://arxiv.org/abs/1704.06857).

- [8] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in : Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
- [9] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, D. Terzopoulos, Image segmentation using deep learning : A survey, IEEE Transactions on Pattern Analysis and Machine Intelligence.
- [10] A. Khouani, M. E. H. Daho, S. A. Mahmoudi, M. A. Chikh, B. Benzineb, Automated recognition of white blood cells using deep learning, Biomedical Engineering Letters 10 (3) (2020) 359–367.
- [11] Y. LeCun, Y. Bengio, et al., Convolutional networks for images, speech, and time series, The handbook of brain theory and neural networks 3361 (10) (1995) 1995.
- [12] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv :1409.1556.
- [13] G. Huang, Z. Liu, L. Van Der Maaten, K. Q. Weinberger, Densely connected convolutional networks, in : Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.
- [14] C. Szegedy, S. Ioffe, V. Vanhoucke, A. Alemi, Inception-v4, inception-resnet and the impact of residual connections on learning, in : Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 31, 2017.

- [15] A. L. Samuel, Some studies in machine learning using the game of checkers, *IBM Journal of research and development* 3 (3) (1959) 210–229.
- [16] F. Rosenblatt, The perceptron : a probabilistic model for information storage and organization in the brain., *Psychological review* 65 (6) (1958) 386.
- [17] M. Minsky, S. A. Papert, *Perceptrons : An introduction to computational geometry*, MIT press, 2017.
- [18] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel, Backpropagation applied to handwritten zip code recognition, *Neural computation* 1 (4) (1989) 541–551.
- [19] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning internal representations by error propagation, Tech. rep., California Univ San Diego La Jolla Inst for Cognitive Science (1985).
- [20] N. Qian, On the momentum term in gradient descent learning algorithms, *Neural networks* 12 (1) (1999) 145–151.
- [21] Y. Nesterov, A method for unconstrained convex minimization problem with the rate of convergence $o(1/k^2)$, in : *Doklady an ussr*, Vol. 269, 1983, pp. 543–547.
- [22] M. D. Zeiler, Adadelata : an adaptive learning rate method, arXiv preprint arXiv :1212.5701.
- [23] Y. N. Dauphin, H. De Vries, Y. Bengio, Equilibrated adaptive

learning rates for non-convex optimization, arXiv preprint arXiv :1502.04390.

- [24] D. P. Kingma, J. Ba, Adam : A method for stochastic optimization, arXiv preprint arXiv :1412.6980.
- [25] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet : A large-scale hierarchical image database, in : 2009 IEEE conference on computer vision and pattern recognition, Ieee, 2009, pp. 248–255.
- [26] J. Xu, L. Xiang, Q. Liu, H. Gilmore, J. Wu, J. Tang, A. Madabhushi, Stacked sparse autoencoder (ssae) for nuclei detection on breast cancer histopathology images, IEEE transactions on medical imaging 35 (1) (2015) 119–130.
- [27] R. Kohler, A segmentation system based on thresholding, Computer Graphics and Image Processing 15 (4) (1981) 319–338.
- [28] J. MacQueen, et al., Some methods for classification and analysis of multivariate observations, in : Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, Vol. 1, Oakland, CA, USA, 1967, pp. 281–297.
- [29] S. Hojjatoleslami, J. Kittler, Region growing : a new approach, IEEE Transactions on Image processing 7 (7) (1998) 1079–1084.
- [30] L. Vincent, P. Soille, Watersheds in digital spaces : an efficient algorithm based on immersion simulations, IEEE Computer Architecture Letters 13 (06) (1991) 583–598.

- [31] M. Kass, A. Witkin, D. Terzopoulos, Snakes : Active contour models, *International journal of computer vision* 1 (4) (1988) 321–331.
- [32] Y. Boykov, G. Funka-Lea, Graph cuts and efficient nd image segmentation, *International journal of computer vision* 70 (2) (2006) 109–131.
- [33] Y. Y. Boykov, M.-P. Jolly, Interactive graph cuts for optimal boundary & region segmentation of objects in nd images, in : *Proceedings eighth IEEE international conference on computer vision. ICCV 2001, Vol. 1, IEEE, 2001, pp. 105–112.*
- [34] H. Derin, H. Elliott, R. Cristi, D. Geman, Bayes smoothing algorithms for segmentation of binary images modeled by markov random fields, *IEEE Transactions on Pattern Analysis and Machine Intelligence* (6) (1984) 707–720.
- [35] N. Settouti, M. E. A. Bechar, M. E. H. Daho, M. A. Chikh, An optimised pixel-based classification approach for automatic white blood cells segmentation, *International Journal of Biomedical Engineering and Technology* 32 (2) (2020) 144–160.
- [36] M. Saidi, M. E. A. Bechar, N. Settouti, M. A. Chikh, Instances selection algorithm by ensemble margin, *Journal of Experimental & Theoretical Artificial Intelligence* 30 (3) (2018) 457–478.
- [37] M. Benazzouz, I. Baghli, A. Benomar, M. Ammar, Y. Benmouna, M. Chikh, Evidential segmentation scheme of bone

marrow images, *Advances in Image and Video Processing* 4 (1) (2016) 37.

- [38] I. Baghli, A. Nakib, E. Sellam, M. Benazzouz, A. Chikh, E. Petit, Hybrid framework based on evidence theory for blood cell image segmentation, in : *Medical Imaging 2014 : Biomedical Applications in Molecular, Structural, and Functional Imaging*, Vol. 9038, International Society for Optics and Photonics, 2014, p. 903815.
- [39] M. L. Benomar, A. Chikh, X. Descombes, M. Benazzouz, Multi-feature-based approach for white blood cells segmentation and classification in peripheral blood and bone marrow images, *International Journal of Biomedical Engineering and Technology* 35 (3) (2021) 223–241.
- [40] Z. Zhang, P. Luo, C. C. Loy, X. Tang, Facial landmark detection by deep multi-task learning, in : *European Conference on Computer Vision*, Springer, 2014, pp. 94–108.
- [41] N. Dhungel, G. Carneiro, A. P. Bradley, Deep learning and structured prediction for the segmentation of mass in mammograms, in : *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2015, pp. 605–612.
- [42] O. Ronneberger, P. Fischer, T. Brox, U-net : Convolutional networks for biomedical image segmentation, in : *International Conference on Medical image computing and computer-assisted intervention*, Springer, 2015, pp. 234–241.

- [43] H. R. Roth, L. Lu, A. Farag, H.-C. Shin, J. Liu, E. B. Turkbey, R. M. Summers, Deeporgan : Multi-level deep convolutional networks for automated pancreas segmentation, in : International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2015, pp. 556–564.
- [44] H. Chen, Q. Dou, D. Ni, J.-Z. Cheng, J. Qin, S. Li, P.-A. Heng, Automatic fetal ultrasound standard plane detection using knowledge transferred recurrent neural networks, in : International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2015, pp. 507–514.
- [45] Q. Dou, H. Chen, L. Yu, L. Zhao, J. Qin, D. Wang, V. C. Mok, L. Shi, P.-A. Heng, Automatic detection of cerebral microbleeds from mr images via 3d convolutional neural networks, IEEE transactions on medical imaging 35 (5) (2016) 1182–1195.
- [46] Q. Dou, H. Chen, Y. Jin, L. Yu, J. Qin, P.-A. Heng, 3d deeply supervised network for automatic liver segmentation from ct volumes, in : International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2016, pp. 149–157.
- [47] M. Żejmo, M. Kowal, J. Korbicz, R. Monczak, Classification of breast cancer cytological specimen using convolutional neural network, in : Journal of Physics : Conference Series, Vol. 783, IOP Publishing, 2017, p. 012060.

- [48] S. Graham, Q. D. Vu, S. E. A. Raza, A. Azam, Y. W. Tsang, J. T. Kwak, N. Rajpoot, Hover-net : Simultaneous segmentation and classification of nuclei in multi-tissue histology images, *Medical Image Analysis* (2019) 101563.
- [49] S. U. Akram, J. Kannala, L. Eklund, J. Heikkilä, Cell segmentation proposal network for microscopy image analysis, in : *Deep Learning and Data Labeling for Medical Applications*, Springer, 2016, pp. 21–29.
- [50] J. Dai, K. He, J. Sun, Instance-aware semantic segmentation via multi-task network cascades, in : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3150–3158.
- [51] J. Yi, P. Wu, Q. Huang, H. Qu, B. Liu, D. J. Hoepfner, D. N. Metaxas, Multi-scale cell instance segmentation with keypoint graph based bounding boxes, in : *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2019, pp. 369–377.
- [52] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, in : *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.
- [53] M. Benazzouz, I. Baghli, M. A. Chikh, Microscopic image segmentation based on pixel classification and dimensionality reduction, *International Journal of Imaging Systems and Technology* 23 (1) (2013) 22–28.
- [54] R. Girshick, Fast r-cnn, in : *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

- [55] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks ?, in : Advances in neural information processing systems, 2014, pp. 3320–3328.
- [56] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, J. Liang, Convolutional neural networks for medical image analysis : Full training or fine tuning ?, *IEEE transactions on medical imaging* 35 (5) (2016) 1299–1312.
- [57] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Noguees, J. Yao, D. Mollura, R. M. Summers, Deep convolutional neural networks for computer-aided detection : Cnn architectures, dataset characteristics and transfer learning, *IEEE transactions on medical imaging* 35 (5) (2016) 1285–1298.
- [58] H. Chen, X. Qi, L. Yu, Q. Dou, J. Qin, P.-A. Heng, Dcan : Deep contour-aware networks for object instance segmentation from histology images, *Medical image analysis* 36 (2017) 135–146.
- [59] W. Stock, R. Hoffman, White blood cells 1 : non-malignant disorders, *The Lancet* 355 (9212) (2000) 1351–1357.
- [60] A. Chaudhury, L. Noiret, J. M. Higgins, White blood cell population dynamics for risk stratification of acute coronary syndrome, *Proceedings of the National Academy of Sciences* 114 (46) (2017) 12344–12349.
- [61] I. E. Livieris, E. Pintelas, A. Kanavos, P. Pintelas, Identification of blood cell subtypes from images using an im-

proved ssl algorithm, Biomed J Sci Tech Res. <https://doi.org/10.26717/BJSTR.1755>.

- [62] D. Bani-Hani, N. Khan, F. Alsultan, S. Karanjkar, N. Nagarur, Classification of leucocytes using convolutional neural network optimized through genetic algorithm, in : Proc. of the 7th Annual World Conference of the Society for Industrial and Systems Engineering, 2018.
- [63] G. Liang, H. Hong, W. Xie, L. Zheng, Combining convolutional neural network with recursive neural network for blood cell image classification, IEEE Access 6 (2018) 36188–36197.
- [64] P. P. Banik, R. Saha, K.-D. Kim, Fused convolutional neural network for white blood cell image classification, in : 2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC), IEEE, 2019, pp. 238–240.
- [65] F. Özyurt, A fused cnn model for wbc detection with mrmr feature selection and extreme learning machine, Soft Computing 24 (11) (2020) 8163–8172.
- [66] A. Kumar, J. Kim, D. Lyndon, M. Fulham, D. Feng, An ensemble of fine-tuned convolutional neural networks for medical image classification, IEEE journal of biomedical and health informatics 21 (1) (2016) 31–40.
- [67] P. Mooney, [Kaggle blood cell images](#), accessed : 2021-02-20 (2018).
URL <https://www.kaggle.com/paultimothymooney/bloodcells/>

- [68] M. Mohamed, B. Far, A. Guaily, An efficient technique for white blood cells nuclei automatic segmentation, in : 2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, 2012, pp. 220–225.
- [69] S. Ruder, An overview of gradient descent optimization algorithms, arXiv preprint arXiv :1609.04747.
- [70] A. C. Wilson, R. Roelofs, M. Stern, N. Srebro, B. Recht, The marginal value of adaptive gradient methods in machine learning, arXiv preprint arXiv :1705.08292.
- [71] N. S. Keskar, R. Socher, Improving generalization performance by switching from adam to sgd, arXiv preprint arXiv :1712.07628.

العنوان: نحو استعمال طرق التعلم العميق للتعرف الذكي على البيانات الطبية

الملخص: في التحليل المجهرى لأمراض الدم، قد يكون لمورفولوجيا ونسبة الخلايا في الشرائح أهمية طبية في تشخيص العديد من الأمراض السرطانية مثل الماييلوما وسرطان الدم الخ. ومن جهة أخرى، فإن التعرف اليدوي يعتبر مهمة شاقة تتطلب الكثير من التركيز والخبرة للتمييز بين العناصر الخلوية المختلفة وحسابها. لذلك، فإنه لمن الضروري وجود نظام تعرف تلقائي لمساعدة الطبيب وجعل قراءة الشرائح سريعة ودقيقة. تعتمد أنظمة القياس المعمول بها حالياً بشكل أساسي على تجزئة وتصنيف خلايا الدم البيضاء. من خلال هاته الأطروحة، سنقترح مساهمتين رئيسيتين، الأولى تخص اكتشاف وتجزئة خلايا الدم البيضاء حيث قمنا بتدريب نموذج عميق ليكون قادراً على اكتشاف جميع نوى وسيتوبلازم الخلايا وتقسيمها أوتوماتيكياً. لقد طبقنا العديد من تقنيات التنظيم وبعض التحسينات على تنبؤات النموذج من خلال إضافة تصحيحات وتنبؤات مشتركة، بالإضافة إلى خوارزمية جديدة تستخدم التعاون بين مخرجات النموذج والمعلومات المكانية الأولية من أجل الحصول على جودة تجزئة أفضل. كانت نتائج هذا العمل الأول واعدة للغاية وتظهر تحسناً كبيراً في الدقة ووقت للصورة الحساب. تتعلق المساهمة الثانية بتصنيف الأنواع الفرعية لخلايا الدم البيضاء. لقد اقترحنا طريقة جديدة بخمس معماريات عميقة لاستخراج خصائص الصور حيث أظهرت النتائج التي تم الحصول عليها فعالية النهج المقترح الذي يستخدم التنبؤ المشترك بين البنى مقارنة مع كل نموذج على حده.

الكلمات المفتاحية: الصورة الميكروسكوبية، التعلم العميق، تجزئة الصور، التصنيف

Titre : Vers une reconnaissance intelligente des données biomédicales par apprentissage en profondeur

Résumé : Dans l'analyse microscopique en hématologie, la morphologie et la proportion des cellules dans les lames peuvent avoir une signification médicale dans le diagnostic de plusieurs maladies cancéreuses telles que le myélome, les leucémies etc. Cependant, la reconnaissance manuelle reste une tâche fastidieuse qui nécessite beaucoup de concentration et d'expérience pour différencier et compter les différents éléments cellulaires. Par conséquent, un système de reconnaissance automatique est nécessaire afin d'aider le praticien et rendre la lecture des lames rapide et précise. Les systèmes automatiques standard sont basés principalement sur la segmentation et la classification des globules blancs. Dans cette thèse, nous proposons deux contributions principales, dans la première, nous nous sommes intéressés à la détection et la segmentation des globules blancs. Nous avons formé un modèle profond pour qu'il soit capable de détecter et de segmenter automatiquement tous les noyaux et les cytoplasmes. Nous avons appliqué plusieurs techniques de régularisations et quelques améliorations sur les prédictions du modèle en ajoutant des corrections et des prédictions combinées, ainsi d'un nouvel algorithme qui utilise une coopération entre les sorties du modèle et les informations spatiales de l'image initiale pour une meilleure qualité de segmentation. Les résultats de ce premier travail ont été très prometteurs et montrent une amélioration significative dans la précision et le temps de calcul. La deuxième contribution concerne la classification des sous-types des globules blancs. Nous avons proposé une nouvelle méthode avec cinq architectures profondes pour l'extraction des caractéristiques à partir des images. Les résultats obtenus ont montré l'efficacité de l'approche proposée qui utilise une prédiction combinée entre les architectures par rapport à chaque modèle individuellement.

Mots-clés : Image microscopique, Apprentissage profond, Segmentation d'image, Classification

Title: Towards intelligent recognition of biomedical data by deep learning

Abstract: In microscopic analysis in hematology, the morphology and proportion of cells in the slides can have medical significance in the diagnosis of several cancerous diseases such as myeloma, leukemia etc. However, manual recognition is a very tedious task that requires a lot of concentration and experience to differentiate and count the different cellular elements. For this reason, an automatic recognition system is needed to assist the practitioner and make the reading of the slides fast and accurate. Standard automatic systems are mainly based on segmentation and classification of white blood cells. In this thesis, we propose two main contributions, in the first one, we focused on the detection and segmentation of white blood cells. We trained a deep model to be able to automatically detect and segment all nuclei and cytoplasms. We applied several regularization techniques and some improvements on the model predictions by adding corrections and combined predictions, as well as a new algorithm that uses a cooperation between the model outputs and the initial image spatial information for a better segmentation quality. The results of this first work were very promising and show a significant improvement in accuracy and computation time. The second contribution concerns the classification of white blood cell subtypes. We proposed a new method with five deep architectures for feature extraction from images. The results obtained showed the effectiveness of the proposed approach that uses a combined prediction between architectures compared to each model individually.

Keywords: Microscopic image, Deep learning, Image segmentation, Classification