

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

وزارة التعليم العالي والبحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد حطب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Mémoire de Master

Mention Génie Biomédical
Spécialité : **Instrumentation Biomédicale**

présenté par

DERBAL Mohamed

&

MENOUER Naïm

Classification des images histologiques par Deep Learning

Proposé par : BECHAR Mohammed El Amine

Année Universitaire 2018-2019

Classifying Medical Images Using Convolutional Neural Network

DERBAL MOHAMED
MENOUEUR NAIM
BECHAR MOHAMMED EL AMINE

July xx, 2019

Remerciements

Avant tout nous remercions "ALLAH" le tout puissant de nous avoir donnés le courage, la volonté, la patience et la santé durant toutes ces années et que grâce à lui ce travail a pu être réalisé.

On remercie nos très chers parents, qui ont toujours été là pour nous. On remercie nos sœurs, et nos frères, pour leurs encouragements.

Nos remerciements les plus vifs et chaleureux, vont à notre promoteur de mémoire Mr. BECHAR Mohammed El Amine, pour son aide, son orientation judicieuse et sa disponibilité, aussi pour la confiance, la patience et la compréhension qu'il nous a toujours manifesté.

Nous adressons nos sincères remerciements à tous les professeurs, intervenants et toutes les personnes qui par leurs paroles, leurs écrits, leurs conseils et leurs critiques ont guidé nos réflexions et ont accepté de nous rencontrer et de répondre à nos questions durant nos études.

Nos vifs remerciements vont également aux membres du jury d'avoir accepté de corriger et d'évaluer notre travail.

Enfin, on remercie nos amis qui ont toujours été là pour nous. Leur soutien inconditionnel et leurs encouragements ont été d'une grande aide.

Résumé :

De nos jours l'intelligence artificielle a pris une place considérable dans le quotidien de l'homme, spécialement dans le domaine médical pour le traitement des images médicales. En effet ce travail traite la classification des images histologiques du cancer de sein en utilisant les réseaux de neurones convolutionnels CNN. Au cours de cette étude on a réalisé cinq architectures CNN, qui ont donné des résultats ascendants jusqu'à arriver à 85 % de précision.

Mots clés :

Intelligence artificielle, réseaux de neurones convolutionnels, images histologiques, classification des images.

Abstract :

In our day artificial intelligence has taken a considerable place in the daily life of man, especially in the medical field for the treatment of medical images. Indeed, this work deals with the classification of histological images of breast cancer using CNN convolutional neural networks. In the course of this study, five CNN architectures were realized, which yielded up to 85 % accuracy.

Keywords :

Artificial intelligence, convolutional neural networks, histological images, image classification.

Table des matières

Remerciments	i
Introduction générale	1
1 Chapitre 1 Context medical	2
1.1 Anatomie et Physiologie du sein	3
1.2 Pathologies des seins	3
1.2.1 Pathologies bénignes	3
1.2.2 Pathologies malignes	4
1.2.2.1 Cancer non invasive	4
1.2.2.2 Cancers invasifs ou infiltrants	4
1.3 Examens du sein	5
1.3.1 Mammographie	5
1.3.2 Echographie	5
1.3.3 IRM	6
1.3.4 Biopsie	6
1.3.4.1 Examen anatomo-pathologique	7
1.3.4.2 Prise en charge d'un prélèvement	7
1.3.4.3 Les étapes techniques d'examen anatomo-pathologique	8
1.4 Conclusion	10
2 Chapitre 2 État de l'art du domaine	11
2.1 État de l'art	12
2.1.1 Etat de l'art des réseaux de neurones convolutionnels	12
2.1.2 Etat de l'art sur la classification des images histologiques du cancer de sein	14
3 Chapitre 3 les réseaux de neurones convolutoinnels	15
3.1 Introduction	16
3.2 Les composants d'un réseau CNN	17
3.2.1 Couche de convolution (CONV)	17
3.2.1.1 Poids et Biais partagés (Weights and Biases)	19
3.2.1.2 Pas de progression (Stride)	19
3.2.1.3 Remplissage par Zéro (Padding)	20
3.2.1.4 Cartes de caractéristiques (Feauture Map)	20
3.2.1.5 Filtres multiples	20
3.2.2 Couche de Pooling (Pool)	21
3.2.3 Couche de ReLU	22
3.2.4 Couche Entièrement Connectée (Fully Connected FC)	23
3.3 Fonctions utilisées dans les réseaux CNN	24
3.3.1 Fonction de Perte/ Coût (LOSS)	24
3.3.2 Surapprentissage (Overfitting)	25
3.3.2.1 Augmentation des données (data augmentation)	25
3.3.2.2 Dropout	26
3.3.3 Descente de gradient (Vanishing gradient)	26
3.4 Etude d'un exemple de réseau CNN	27
3.5 Conclusion	29

4	Chapitre 4 Architectures proposées	30
4.1	Introduction	31
4.2	Choix de l'architecture	31
4.3	Architectures proposées	31
4.3.1	Modèle 1	32
4.3.2	Modèle 2	33
4.3.3	Modèle 3	33
4.3.4	Modèle 4	34
4.3.5	Modèle 5	35
4.4	Conclusion	36
5	Chapitre 5 Résultats et discussions	37
5.1	Introduction	38
5.2	Logiciels et bibliothèque Utilisés dans l'implémentation	38
5.2.1	Python	38
5.2.2	TensorFlow	38
5.2.3	Keras	39
5.3	Configurations Utilisées dans l'implémentation	39
5.4	La base de données	40
5.5	Résultats obtenus et discussions	40
5.5.1	Résultats obtenus pour le 1er Modèle	41
5.5.2	Résultats obtenus pour le 2ème Modèle	42
5.5.3	Résultats obtenus pour le 3ème Modèle	43
5.5.4	Résultats obtenus pour le 4ème Modèle	44
5.5.5	Résultats obtenus pour le 5ème Modèle	45
5.5.6	Comparaison des résultats	47
5.6	Conclusion	48
	Conclusion générale	49
	Références	50

Table des figures

1	Anatomie du sein chez la femme [1].	3
2	Images mammographiques qui montrent la présence d'une masse tumorale [2].	5
3	Image echographique d'un sein [3].	6
4	Une tumeur détectée dans le sein droit sur une image IRM [4].	6
5	Automate de fixation et d'imprégnation.	8
6	prélèvements mis dans la paraffine.	9
7	Refroidissement de la paraffine.	9
8	Illustre des coupes au microtome.	9
9	Coupes après coloration.	10
10	Exemple qui montre les étapes d'apprentissage d'un réseau de neurones.	16
11	Couches de convolution avec plusieurs caractéristiques et images RVB à l'entrée [5].	17
12	Exemple d'une opération de convolution [6].	18
13	Convolution sur image d'entrée [6].	19
14	Opération de remplissage par zéro.	20
15	Cartes de caractéristiques formés de 32 filtres [6].	21
16	Opération de max Pooling [6].	22
17	Fonction ReLU [5].	22
18	Couche entièrement connectée « fully connected » [7].	23
19	Exemple d'une prédiction à travers les Couches « fully connected » [7].	24
20	Opération d'augmentation de données [8].	25
21	Régularisation par Dropout.	26
22	Descente de gradient.	27
23	Architecture standard d'un réseau CNN.	28
24	Architecture du réseau AlexNet.	28
25	Architecture du modèle 1	32
26	Architecture du modèle 2	33
27	Architecture du modèle 3	34
28	Architecture du modèle 4	35
29	Architecture du modèle 5	36
30	Logo de Python.	38
31	Logo de Tensorflow.	39
32	Logo de Keras.	39
33	base de donnée.	40
34	Courbes de Précision et d'Erreur pour le 1er modèle	41
35	les valeurs des Précisions et d'Erreur avec le nombre epochs du 1er modèle.	42
36	Courbes de Précision et d'Erreur pour le 2ème modèle	42
37	les valeurs des Précisions et d'Erreur avec le nombre d'époque du 2ème modèle.	43
38	Courbes de Précision et d'Erreur pour le 3ème modèle	43
39	les valeurs des Précisions et d'Erreur avec le nombre d'époque du 3ème modèle.	44
40	Courbes de Précision et d'Erreur pour le 4ème modèle	44
41	les valeurs des Précisions et d'Erreur avec le nombre d'époque du 4ème modèle.	45
42	Courbes de Précision et d'Erreur pour le 5ème modèle	45
43	les valeurs des Précisions et d'Erreur avec le nombre d'époque du 5ème modèle	47

Liste des tableaux

1	Comparaison des précision et erreurs d'apprentissage entre les modèles.	48
2	Comparaison des temps d'apprentissage entre les modèles.	48

Introduction générale

Le cancer est un problème de santé publique majeur dans le monde d'aujourd'hui. Selon le Centre international de recherche sur le cancer de l'Organisation mondiale de la santé, 8,2 millions de décès ont été causés par le cancer en 2012 et 27 millions de nouveaux cas de cette maladie sont attendus avant 2030 [9]. En particulier, le cancer du sein est l'un des types de cancer les plus courants chez les femmes. La mortalité par cancer du sein est très élevée par rapport aux autres types de cancer. La détection et le diagnostic du cancer du sein peuvent être réalisés par des procédures d'imagerie telles que des mammographies (rayons X), une imagerie par résonance magnétique (IRM), une échographie. L'imagerie pour le dépistage du cancer a été étudiée pendant plus de quatre décennies. Cependant, la biopsie est le seul moyen de diagnostiquer avec confiance si le cancer est réellement présent. Parmi les techniques de biopsie, les plus courantes sont l'aspiration à l'aiguille fine, la biopsie à l'aiguille centrale, la biopsie assistée par aspiration et la biopsie chirurgicale (ouverte). La procédure consiste à prélever des échantillons de cellules ou de tissus, qui sont fixés sur une lame de microscope pour l'examen microscopique. Le diagnostic à partir d'une image histopathologique est donc la méthode de référence pour diagnostiquer presque tous les types de cancer, y compris le cancer du sein. Le diagnostic final du cancer du sein, y compris le classement et la stadification, est effectué par des pathologistes qui appliquent une inspection visuelle des échantillons histologiques au microscope. L'analyse histopathologique est une tâche spécialisée qui prend beaucoup de temps. Elle dépend de l'expérience des pathologistes et est influencée par des facteurs tels que la fatigue et la diminution de l'attention. Comme l'ont souligné Gurcan et al. [10], il existe un besoin urgent de diagnostic assisté par ordinateur (pour alléger la charge de travail des pathologistes en filtrant des zones manifestement bénignes, afin que les experts puissent se concentrer sur les cas les plus difficiles à diagnostiquer. des efforts ont donc été consacrés au domaine de l'analyse d'images histopathologiques du cancer du sein, et en particulier à la classification automatisée d'images bénignes ou malignes, pour le diagnostic assisté par ordinateur.

L'objectif de notre travail est de développer un système d'aide au diagnostic pour le dépistage du cancer de sein, le système à pour but de classer des images histologiques qui comportent des tumeurs en bénigne ou maligne. Pour ce faire, on a opté d'utiliser l'intelligence artificielle à travers les réseaux de neurones convolutionnels.

Pour cela, nous avons structuré notre en cinq chapitres :

Chapitre 1 : introduit l'anatomie du sein, ensuite on a une vue globale sur les pathologies les plus récurrentes dans le sein ainsi que les différentes techniques de diagnostics et de dépistage du cancer de sein.

Chapitre 2 : présente un état de l'art sur la classification des images médicales en utilisant les réseaux de neurones convolutionnels en générale et les images histologies du cancer de sein en particulier.

Chapitre 3 : toutes les notions et les techniques des réseaux de neurones convolutionnels sont décrites dans ce chapitre.

Chapitre 4 : présente les architecture des réseaux de neurones convolutionnels que nous avons proposé dans ce travail.

Chapitre 5 : étudie et discute tous les résultats obtenus dans le chapitre précédent, la base de donnée est aussi présentée dans ce chapitre.

Nous terminons ce mémoire par une conclusion générale.

1 Chapitre 1 Context medical

1.1 Anatomie et Physiologie du sein

Les seins sont situés au niveau du thorax, en avant du muscle pectoral de chaque côté et s'étendent en hauteur jusqu'à la clavicule, et en largeur de l'aisselle jusqu'au milieu du sternum environ. Ils sont asymétriques et ne contiennent pas de muscle. Ils sont soutenus par des ligaments [11]. Les seins sont constitués d'une glande mammaire, de tissu conjonctif et de tissu graisseux (Adipeux). C'est ce tissu graisseux qui leur donne forme et volume. Recouverts par la peau, ils arborent en leur centre une région pigmentée appelée aréole mammaire qui elle-même entoure le mamelon (orifice d'éjection du lait). La glande mammaire est formée de 15 à 25 lobes qui contiennent des lobules. Ces lobules renferment des unités sécrétrices, les acini, qui produisent le lait. Ce dernier est acheminé des lobules jusqu'au mamelon par des canaux sécréteurs, les canaux galactophores (voir figure 1).

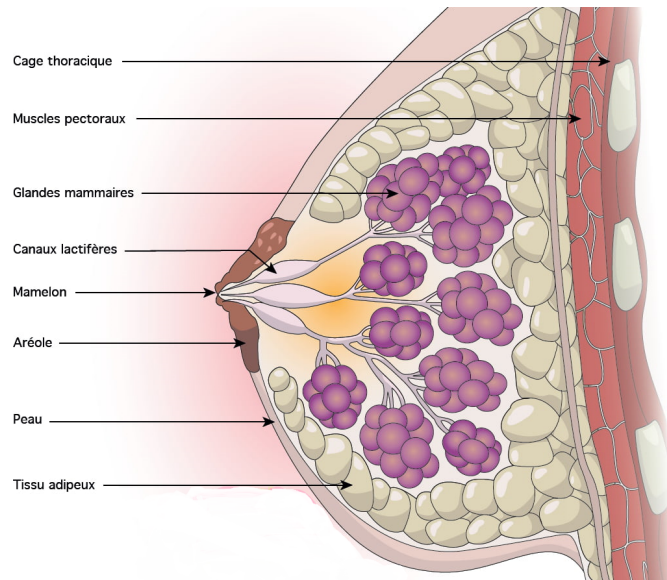


FIGURE 1 – Anatomie du sein chez la femme [1].

Le développement des seins est un des signes de la puberté chez la jeune fille. Ce développement s'effectue grâce à l'action de nombreuses hormones [11]. Ce sont les oestrogènes, la progestérone et les hormones hypophysaires.

Pendant le cycle menstruel, la glande mammaire subit des modifications et notamment augmente de volume sous l'effet des oestrogènes (2 semaines après le dernier jour de règles environ).

Au moment de la ménopause, suite à une diminution de la fabrication hormonale (oestrogènes), le volume des seins diminue.

Le rôle principal du sein est de fabriquer le lait. La préparation à la sécrétion lactée se fait pendant la grossesse grâce à 2 hormones, les oestrogènes et la progestérone (sécrétées par l'ovaire et le placenta) favorisant ainsi le développement des glandes mammaires mais également l'agrandissement des mamelons.

Dès que l'enfant est né, les seins fabriquent le colostrum (liquide blanc jaunâtre) qui va, au bout de quelques jours (3 environ), laisser sa place au lait véritable.

1.2 Pathologies des seins

1.2.1 Pathologies bénignes

La mastodynie prémenstruelle : correspond à la tension douloureuse des seins dans les jours précédents les règles [12]. Phénomène banal, il peut toutefois nécessiter un traitement si les douleurs se font trop

intenses.

Les écoulements par le mamelon : ils sont anormaux en dehors de la grossesse et de l'allaitement. La cause peut être une tumeur bénigne, une absence des règles ou une tumeur dans un canal galactophore [12]. Il est alors nécessaire de consulter un médecin.

Les fibro-adénomes : tumeurs bénignes généralement dures et isolées [12]. Il est cependant nécessaire de les surveiller.

Les mastoses : ce sont des tumeurs bénignes qui se caractérisent par une densification du tissu glandulaire. Des nodules douloureux (grosseur anormale généralement arrondie) ou des kystes fréquemment remplis de liquide peuvent apparaître [12]. Ces tumeurs sont détectables à la palpation. La prise orale de progestatif (dérivé de la progestérone) ou la ponction des kystes afin d'en évacuer le liquide s'avèrent efficaces.

1.2.2 Pathologies malignes

Un cancer signifie la présence de cellules anormales qui se multiplient de façon incontrôlée. Dans le cas du cancer du sein, les cellules peuvent rester dans le sein ou se répandre dans le corps par les vaisseaux sanguins ou lymphatiques. La plupart du temps, la progression d'un cancer du sein prend plusieurs mois et même quelques années. Le cancer du sein est le cancer le plus diagnostiqué chez les femmes à travers le monde, autant avant qu'après la ménopause. Une femme sur 9 sera atteinte d'un cancer du sein au cours de sa vie et 1 femme sur 27 en mourra. Les divers types de cancers du sein évoluent de façon différente :

1.2.2.1 Cancer non invasive

Carcinome canalaire in situ : C'est le type le plus fréquent de cancer du sein non invasif chez la femme. Comme son nom l'indique, il se forme à l'intérieur des canaux de lactation du sein [12]. On diagnostique beaucoup plus fréquemment ce type de cancer depuis l'utilisation plus répandue de la mammographie. Le traitement de ce cancer mène à la guérison dans presque tous les cas. Normalement, il ne se dissémine pas. Dans des cas exceptionnels, sans traitement, il poursuit sa croissance et peut alors devenir « infiltrant » donc se propager à l'extérieur des canaux de lactation.

1.2.2.2 Cancers invasifs ou infiltrants

Ces formes de cancer envahissent les tissus autour des canaux de lactation, mais demeurent à l'intérieur du sein. Par contre, si la tumeur n'est pas traitée, elle peut se disséminer à d'autres parties du corps (par exemple, les os, les poumons ou le foie) en générant des métastases [12].

Carcinome canalaire : Il se forme dans les canaux de lactation [12]. Les cellules cancéreuses traversent la paroi des canaux.

Carcinome lobulaire : Les cellules cancéreuses apparaissent dans les lobules regroupés dans les lobes. Puis, elles traversent la paroi des lobules et se disséminent dans les tissus environnants.

Carcinome inflammatoire : Un cancer rare qui se caractérise principalement par un sein qui peut devenir rouge, enflé et chaud. La peau du sein peut aussi prendre l'aspect d'une peau d'orange. Ce type de cancer progresse plus rapidement et est plus difficile à traiter.

Autres carcinomes : (médullaires, colloïdes ou muscinaux, tubulaires, papillaires). Ces types de cancer du sein sont plus rares. Les principales différences entre ces types de cancer reposent sur le type de cellules touchées.

Maladie de Paget : Un cancer rare qui se manifeste par une petite plaie au mamelon qui ne guérit pas.

1.3 Examens du sein

Pour confirmer et affiner le diagnostic, les médecins utilisent d'abord les examens d'imagerie puis procèdent à une analyse biologique de la tumeur suivant un protocole bien déterminé. Voici les étapes du diagnostic de la tumeur responsable des cancers du sein.

1.3.1 Mammographie

Le premier examen est souvent la mammographie (voir figure 2). C'est une radiographie, à rayon X, des seins. Une tumeur se traduit soit par une masse soit par la présence de petits points blancs, les "microcalcifications", dont le nombre, la forme, la répartition permettent de suspecter ou non l'existence d'un cancer. La mammographie est parfois complétée par deux autres examens d'imagerie.

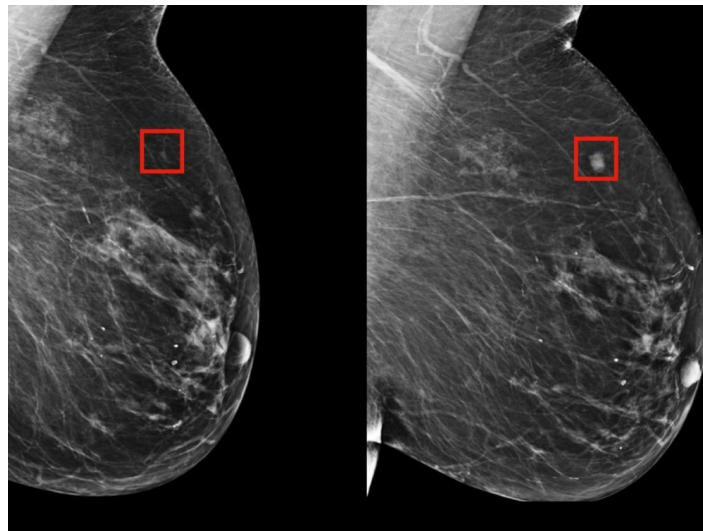


FIGURE 2 – Images mammographiques qui montrent la présence d'une masse tumorale [2].

1.3.2 Echographie

Elle utilise quant à elle la technique des ultrasons. Elle permet de dire, par exemple, si la boule repérée à l'échographie, est un kyste composé de liquide ou plutôt une tumeur solide comme il est montré dans la figure 3

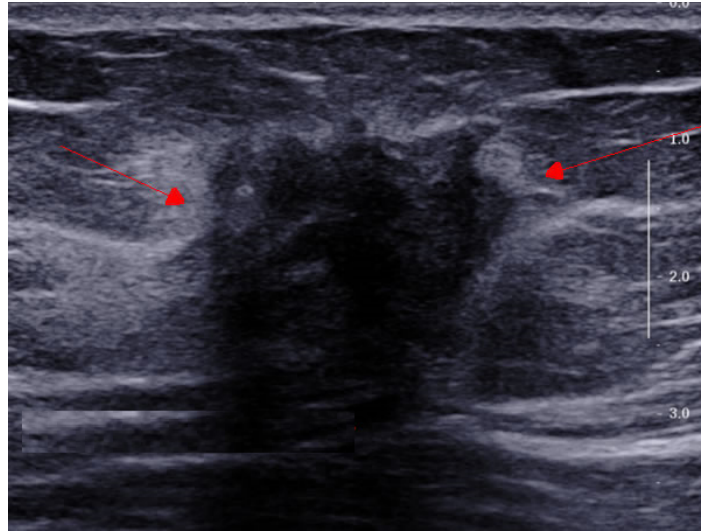


FIGURE 3 – Image échographique d'un sein [3].

1.3.3 IRM

Plus rarement, le médecin peut demander un examen d'IRM (imagerie à résonance magnétique) (voir figure 4). Il est loin d'être systématique mais peut notamment permettre, lui aussi, de faire la différence entre une anomalie bénigne et une anomalie maligne. Il peut aussi servir à vérifier si le cancer s'est propagé ou s'il y a une récurrence locale, pour les personnes ayant déjà été traitées pour un cancer du sein.

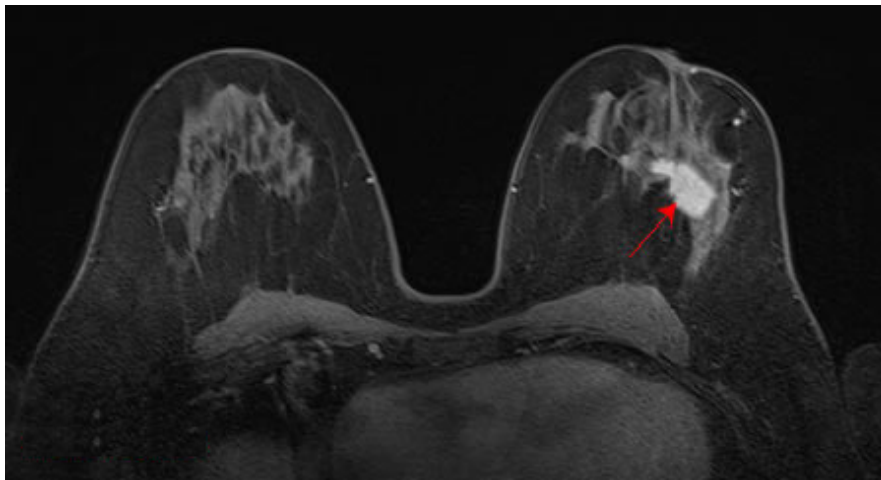


FIGURE 4 – Une tumeur détectée dans le sein droit sur une image IRM [4].

1.3.4 Biopsie

Lorsqu'une lésion suspecte est détectée dans un sein après un examen d'imagerie (mammographie, échographie ou IRM) des examens complémentaires sont nécessaires pour déterminer s'il s'agit ou non d'un cancer. Pour cela, on pratique un prélèvement de la lésion, c'est-à-dire une biopsie.

La biopsie consiste à prélever des fragments de tissus du sein, au niveau de l'anomalie détectée. Afin d'établir un diagnostic fiable, le médecin utilise l'une de ces deux techniques : la microbiopsie et la macrobiopsie. Le volume prélevé est bien plus important avec la macrobiopsie, indiquée en cas de microcalcifications. La microbiopsie est essentiellement utilisée pour le diagnostic des masses ou des nodules. Son efficacité est excellente, elle permet de distinguer les lésions bénignes et malignes dans plus de 98 pourcent des cas.

Micro-biopsie : se pratique après désinfection de la peau, sous anesthésie locale, et le médecin utilise un pistolet automatique muni d'une aiguille qui permet d'effectuer un prélèvement rapide [13]. Le plus souvent, l'acte est réalisé sous contrôle échographique afin de bien cibler l'anomalie avec l'aiguille. Plusieurs prélèvements peuvent être effectués. L'examen est indolore, et on entend simplement le bruit du pistolet qui « claque » lors de l'insertion de l'aiguille.

Macro-biopsie : se fait également sous anesthésie locale. Pour repérer l'anomalie, on utilise une technique appelée stéréotaxie [13] : la patiente est assise devant le mammographe, torse nu. Plusieurs clichés sont réalisés et transmis à un ordinateur qui va guider l'aiguille jusqu'à l'image anormale. Le sein à biopsier est comprimé par un appareil de mammographie doté d'un système permettant de repérer les coordonnées de l'anomalie et de guider le médecin. Une quinzaine de prélèvements peuvent être effectués, avec une aiguille de plus gros calibre que pour la micro-biopsie, qui « aspire » le tissu prélevé. L'examen dure moins d'une demi-heure et n'est pas trop douloureux. L'incision est refermée par un pansement.

Ces prélèvements sont ensuite passés par l'examen anatomo-pathologique qui est détaillé par la suite.

1.3.4.1 Examen anatomo-pathologique

L'examen anatomopathologique consiste à analyser au microscope des cellules ou des tissus prélevés sur le sein [14], cela permet notamment d'établir la présence d'une inflammation ou encore d'une tumeur ; on parle aussi d'examen histopathologique ou examen histologique, qui est particulièrement essentiel en oncologie afin de distinguer les tumeurs bénignes des tumeurs malignes et de déterminer le degré de la malignité. Cet examen est souvent abrégé par les professionnels de santé en « examen Anapath ». C'est l'examen anatomopathologique qui permet d'établir de façon définitive le diagnostic de cancer. On parle de preuve histologique.

1.3.4.2 Prise en charge d'un prélèvement

Après le prélèvement des échantillons qui résultent du Micro ou Macro biopsie, ces échantillons vont être visualisées par un médecin spécialiste par les yeux dans un premier temps (examen macroscopique), puis par un microscope (Techniques microscopiques) selon des étapes techniques qui sont citées par la suite, dont le but est comparer les cellules prélevées du patient avec les cellules normales, la comparaison se fait en fonction de :

- Lieu de prélèvement.
- Age.
- Type de lésion clinique.
- Histoire de la maladie.

La prochaine étape, c'est l'identification de la lésion, le médecin doit connaître la nature de lésion prélevée :

1. Si Tumoral, il va déterminer si elle est bénigne ou maligne avec le nom de type de la tumeur de façon précise.
2. Si Autres, il va déterminer si elle est Inflammatoire, Maladie congénitale ou bien Sub-normal.

1.3.4.3 Les étapes techniques d'examen anatomo-pathologique

Le plus souvent, le prélèvement (matériel histologique) est fixé, inclus, coupé et coloré afin de pouvoir l'observer au microscope. Le matériel peut être prélevé par biopsie ou provenir d'une pièce opératoire ou d'une autopsie, les étapes techniques sont :

La fixation du tissu : La fixation a pour buts la conservation des structures et le durcissement des tissus. Elle doit se faire immédiatement après le prélèvement, par immersion du prélèvement dans du fixateur : solution de formol 'CH₂O' ou solution de formaldéhyde concentré, on utilise cette solution, en la diluant dans de l'eau, on parle alors de solution de formol, qui contient en fait 4 pourcents de formaldéhyde, une concentration idéale pour la fixation.

Le formol est un produit très toxique, allergisant et cancérigène, qui est utilisé pour la fixation des cellules qui forme le tissu prélevé, pour qu'il garde sa forme initiale et n'est pas lyser ou délatté.

Le formol pénètre lentement dans les tissus (1mm/heure), donc selon la taille de la pièce à étudier, la fixation est plus ou moins longue. Cette étape est importante, en effet si le fixateur ne pénètre pas dans le tissu, celui-ci risque de se lyser.



FIGURE 5 – Automate de fixation et d'imprégnation.

L'inclusion en paraffine : L'inclusion a pour but de permettre la réalisation de coupes fines et régulières. Le milieu d'inclusion le plus utilisé est la paraffine. Comme la paraffine est hydrophobe, le prélèvement doit d'abord subir une déshydratation (par immersion dans des bains d'alcool de degré croissant puis dans des bains de toluène) avant d'être coulé dans un moule contenant de la paraffine fondue par chauffage et devenue liquide, qui infiltre alors toute la pièce. Après refroidissement, on se trouve en présence d'un bloc de paraffine, dur, à l'intérieur duquel la pièce prélevée est incluse.

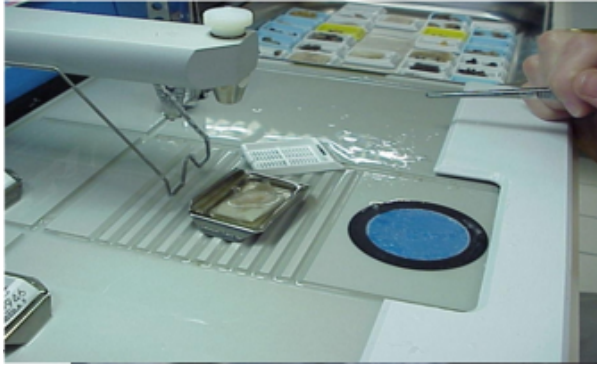


FIGURE 6 – prélèvements mis dans la paraffine.

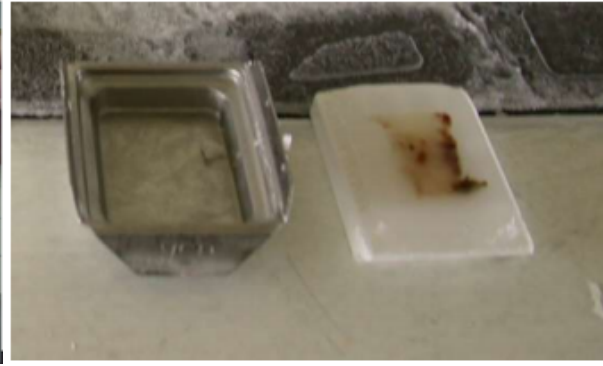


FIGURE 7 – Refroidissement de la paraffine.

Coupes : Les coupes du bloc de paraffine sont faites avec un microtome permettant de réaliser des tranches de section (coupes) de 2 à 5 micro mètres d'épaisseur. Les coupes sont recueillies sur des lames de verre et mises à sécher (sur la nuit à 40-45°C ou 1H max 60°C).

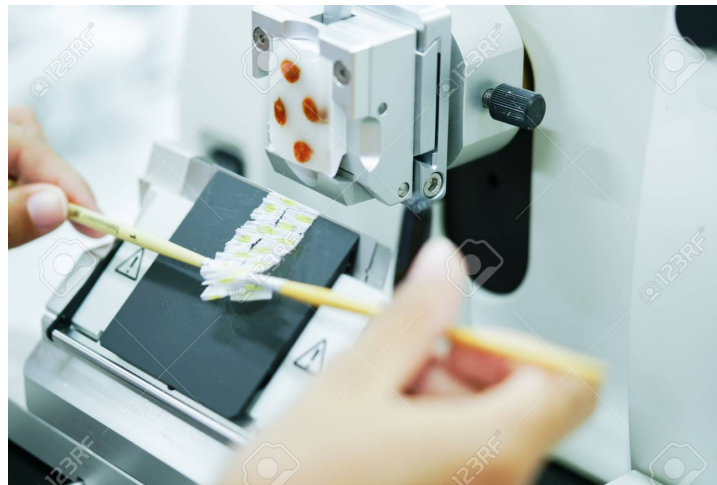


FIGURE 8 – Illustre des coupes au microtome.

Coloration : La plupart des tissus sont transparents, Les colorations réalisées sur lames, accentuent les contrastes pour pouvoir reconnaître différents éléments du tissu. Elle permet d'identifier l'architecture tissulaire (dans les tumeurs, l'architecture tissulaire est anormale).

Choix des colorations : Dans l'hôpital, le pathologiste joue un rôle important car c'est à partir de son diagnostic que le traitement sera institué.

La coloration de routine permet au pathologiste de se faire une idée de la pathologie présente et d'avoir une vision globale de la morphologie des tissus (noyau, cytoplasme et collagène).

Il peut ensuite demander à revoir le même tissu mais cette fois avec un élément tissulaire particulier qui aura été mis en évidence de façon spécifique. On parle dans ce cas de colorations spéciales.

Les colorations de routine : La coloration de routine est effectuée dans un automate. Après déparaffinage puis réhydratation, les coupes sont colorées par des passages dans des produits chimiques afin de différencier par des couleurs les composants du prélèvement (noyaux, cytoplasmes etc.).

L'éosine colore les cytoplasmes en rose et les fibres dans une gamme de roses plus ou moins vifs selon l'acidophilie des différents éléments.

Colorants :

- les colorants basiques colorent en bleu les noyaux : basophilie.
- les colorants acides ou anioniques colorent les cytoplasmes en rose : acidophilie.
- En coloration standard, les noyaux sont plutôt bleus et les cytoplasmes plutôt jaunes orangés.

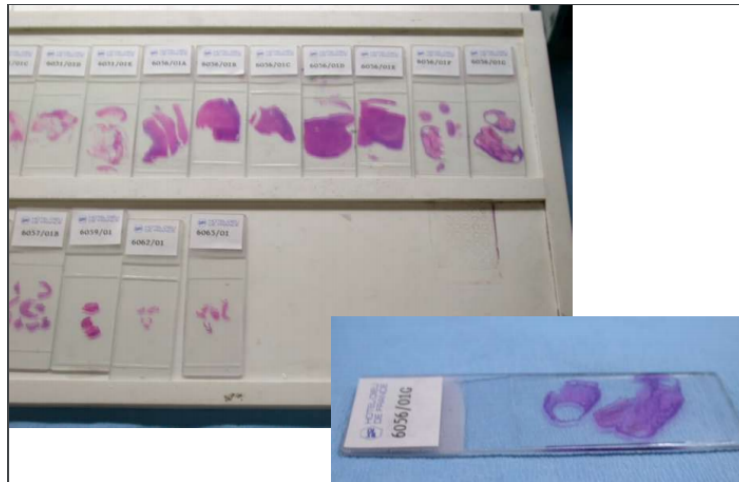


FIGURE 9 – Coupes après coloration.

1.4 Conclusion

Ce chapitre nous a permis de comprendre l'anatomie ainsi que la physiologie des seins, nous avons vu aussi les différentes pathologies qui peuvent affectées le sein. On a passé en revu toutes les méthodes d'examens pour affirmer ou non la présence d'une tumeur dans le sein. Il est claire que l'imagerie joue un rôle très important dans le diagnostic des tumeurs, pour cela l'essentiel de notre travail se repose sur le traitement des images médicale afin de porter aide aux diagnostics des médecins.

2 Chapitre 2 État de l'art du domaine

2.1 État de l'art

Les réseaux de neurones convolutionnels sont de plus en plus utilisés dans la classification des images, et en particulier des images médicales, du fait de leur précision élevée par rapport aux méthodes de classification classiques tel que l'arbre de décision, Support Vector Machine (SVM). A ce titre, plusieurs travaux ont été effectués afin de développer des outils d'aide au diagnostic, pour cela nous avons effectué un état de l'art sur les CNN en premier lieu, et la classification des images histologiques du cancer de sein en second lieu.

2.1.1 État de l'art des réseaux de neurones convolutionnels

Y.XU et al [15], ont proposé une méthode de segmentation des images IRM, de cerveau de nouveau-nés à l'aide d'un réseau de neurones totalement convolutif (FCN), en lui associant des informations pseudo-3D et transfert d'apprentissage. Ceci en vue d'évaluer, l'évolution du cerveau et de diagnostiquer les pathologies neurologiques. Cette méthode a permis de segmenter automatiquement tout un volume IRM 3D en quelques secondes, grâce à l'approche pseudo-3D; de plus elle ne nécessite que très peu de volume d'apprentissage, suite à l'utilisation du transfert d'apprentissage. Ce système, segmente des images 2D en couleurs qui sont formées à partir du volume 3D. L'image est mise à l'entrée du réseau de base FCN et la sortie de ce dernier nous donne une image segmentée; cette méthode est expliquée dans [15]. Les résultats obtenus sont d'une grande précision et le temps d'exécution, pour la segmentation d'un volume IRM entier est de quelques secondes.

A.Sankari et S.Vigneshwari [16] ont suggéré une stratégie de segmentation automatique des tumeurs cérébrales par les réseaux de neurones convolutionnels (Convolutional Neural Network (CNN)), en utilisant des filtres de petite taille 3x3, qui permet une conception plus profonde, et cela pour améliorer la qualité de l'architecture du réseau. La stratégie proposée utilise une technique de domaine spatial pour fournir une image de qualité, et la technique CNN identifie en outre l'emplacement ainsi que le territoire de la tumeur dans le cerveau. La technique proposée [16] permet une segmentation automatique et précise des tumeurs, (à l'aide d'une technique de domaine spatial et de CNN), elle propose aussi une identification et une analyse de la tumeur dans la phase antérieure, et cette proposition joue un rôle très important dans l'augmentation des chances de traitement.

S.E.Amin et M.A.Megeed [17] ont présenté un système de segmentation pour détecter et classer automatiquement divers types de tumeurs cérébrales pouvant être représentées dans une image IRM, comme « le neurinome acoustique, les astrocytomes et le gliome optique », le système se base sur les réseaux de neurones intelligents, et il est divisé en deux parties principales : la première est un réseau neuronal PCA composé de systèmes hybrides, qui extrait les composants principaux des cas d'IRM, et le réseau MLP qui sert de réseau de classification, classe les principaux composants aux cas similaires; pour un but de sauver le temps au radiologue et d'augmenter la précision et d'aider ainsi les médecins non experts à diagnostiquer les tumeurs cérébrales.

A.Q.Syed, K.Narayanan [18] ont introduit une méthode automatique de détection des tumeurs cérébrales, et de classer automatiquement les tissus cérébraux en classe normale et anormale (maligne ou bénigne) à partir des images IRM. Cette méthode [18] est basée sur un réseau neuronal artificiel. Ils ont essayé de décrire clairement les tissus cérébraux à l'aide du réseau de perceptrons multicouches, de l'énergie, de l'entropie, du contraste plus des autres caractéristiques statistiques telles que la moyenne, la médiane, la variance et la corrélation. Il est utilisé à partir d'une méthode de sélection des fonctionnalités pour réduire également l'espace des fonctionnalités. L'objectif est d'aider les radiologues et augmenter la précision et ainsi le rendement du diagnostic.

A.R.Lopez et al [19] ont proposé une approche basée sur l'apprentissage en profondeur au problème de

la classification des lésions cutanées, plus précisément, pour résoudre le problème de la classification maligne ou bénigne d'une image dermatoscopique contenant une lésion cutanée; La solution proposée repose sur l'architecture de réseau neuronal convolutif VGGNet qui utilise le paradigme du transfert d'apprentissage [19]; ils ont conçu un classifieur à deux classes qui prend en entrée des images de lésions cutanées marquées comme bénignes ou malignes, construit un modèle à l'aide de réseaux neuronaux convolutifs profonds et utilise ce modèle pour prédire soit bénigne ou maligne [19]. Cette méthode donne des bons résultats au niveau de sensibilité et précision par rapport aux l'état précédente de la technique sur ce jeu de données.

Dans le domaine de la classification des images médicales S.Khan et S.Yong [20], ont eu pour objectif de proposer une nouvelle architecture de réseaux de neurones convolutionnels (CNN) pour classifier des images contenant différents objets anatomiques et aussi de faire une étude comparative entre leur architecture et les trois architectures phares existantes dans le domaine de la classification d'image, à savoir LeNet, AlexNet et GoogLeNet. Ce travail a porté sur l'évaluation complète des trois architectures CNN citées précédemment en vue de les étudier dans la classification de l'anatomie dans les images médicales. La méthode proposée [20] a démontré sa supériorité en matière de performances pour la classification des images de l'anatomie, en fait cette méthode consiste à modifier l'architecture de base d'AlexNet ce qui nous donne la nouvelle architecture CNN. Les résultats obtenus nous montrent que la nouvelle proposition de CNN surpasse les trois autres architectures CNN en termes d'exécution, de perte d'entraînement, de précision de la validation et de précision des tests.

Pour faire la classification automatique des cellules pathologiques et des cellules saines, P.Buysens et A.Elmoataz [21], ont utilisé une approche basée sur des réseaux de neurones convolutionnels multi-échelle (MCNN); Cela contribue à améliorer le diagnostic pour les cytopathologistes. La méthode qui est utilisée pour la classification cellulaire se repose sur l'extraction automatique de caractéristiques et leur classification. De plus, une approche multi-échelle est proposée utilisant plusieurs réseaux profonds à différentes échelles, et dont les sorties sont fusionnées, cette méthode est expliquée dans [21]. Les résultats obtenus donnent un meilleur taux de classification sur une base de données de cellules par rapport aux méthodes classiques.

E.Costa et al [22], ont proposé une méthode de segmentation automatique de la valve mitrale. En utilisant les réseaux de neurones convolutionnels ils ont créé une architecture nommée UNet, cela contribue à aider les opérateurs à déterminer avec exactitude la morphologie des valves. Cette méthode ne demande pas un volume important d'apprentissage seule l'image à traiter est nécessaire à l'entrée du réseau. Les résultats obtenus de l'évaluation clinique de la segmentation sont en accord avec les résultats quantitatifs.

D.S.Kermany et al [23], ont proposé une méthode de classification des images rétiniennes OCT (Optical Coherence Tomography) pour catégoriser les causes courantes de perte de vision sévère (Dégénérescence maculaire et la rétinopathie diabétique). Dans cette étude, un algorithme de transfert d'apprentissage a été développé en vue de reconnaître des motifs distincts d'une catégorie d'images spécifiques dans un temps plus court, et moins de puissance de calcul. La solution proposée consiste à entraîner un réseau de neurones convolutif (CNN) sur ImageNet de 1000 catégories et l'adapter à un réseau qui est entraîné sur des images OCT. Des couches de convolution ont été gelées et transférées à un nouveau réseau, tandis que les couches totalement connectées (Fully Connected FC) sont recréées et reconverties à partir d'une initialisation aléatoire au-dessus des couches transférées. Les résultats obtenus ont montré une grande performance d'analyse et de classification des images OCT sans le besoin d'une base de données contenant des millions d'images, ni à un réseau d'apprentissage profond très spécialisé; bien que le réseau a été entraîné avec un nombre réduit d'images, il a conservé ses performances en précision, en sensibilité et en spécificité.

2.1.2 Etat de l'art sur la classification des images histologiques du cancer de sein

A. Cruz-Roaa et al [24], ont proposé une méthode de détection automatique du carcinome canalaire invasif (IDC) dans des images histologiques de diapositives complètes du cancer de sein. Pour ce faire un réseau de neurones convolutif (CNN) a été développé [24] pour analyser les régions tumorales. La méthode a été évaluée sur un ensemble de données de diapositives complètes de 162 patients diagnostiqués avec IDC. 113 diapositives ont été sélectionnées pour l'entraînement et 49 diapositives ont été présentées pour le test. L'objectif de ce travail est de mesurer la précision de la classification dans la détection des régions de tissu IDC dans des diapositives complètes. Les résultats obtenus donnent une meilleure détection automatique des régions IDC en termes de F- mesuré et de l'accuracy (71,80 84,23).

F. A. Spanhol et al, ont présenté dans [25] les performances de classification d'un réseau CNN, conçu pour discriminer les tumeurs bénignes et malignes, ce réseau est appliqué sur une base de données, appelée BreakeHis, qui est composé de 7909 images microscopiques de tissus de tumeurs du sein, ces images sont prises en utilisant différents facteurs de grossissement 40, 100, 200 et 400. Le système de classification est basé sur quatre modèles d'apprentissage automatique, formés avec différentes représentations de texture. Un ensemble complet d'expériences montre que les taux de précision avec ce système de base vont de 80 à 85 pourcents, en fonction du facteur de grossissement de l'image.

M. H. Motlagh et al, présentent dans [26] une architecture nommée ResNet, cela permet d'identifier les images histologiques cancéreuses issus des seins. En effet ils ont démontré que le réseau réalisé détecte et classifie de manière automatique la tumeur et ses sous-types, à cela des techniques supplémentaires comme l'augmentation des données ont été rajoutés pour améliorer la précision de prédiction. Cette étude montre que le taux de précision de classification des tumeurs bénignes et malignes est de 98,7 pourcent.

3 Chapitre 3 les réseaux de neurones convolutonnels

3.1 Introduction

Un réseau de neurones convolutionnels « CNN » (Convolutional Neural Network) est un cas concret de réseaux neuronaux Deep Learning, déjà utilisés à la fin des années 90, mais qui sont devenus extrêmement populaires ces dernières années pour obtenir des résultats impressionnants dans la reconnaissance des images, impactant profondément le domaine de la vision par ordinateur.

Les réseaux de neurones convolutifs sont formés de neurones dont les paramètres prennent la forme de poids et de biais qui peuvent être modifiés ou ajustés durant l'opération d'apprentissage [5].

Pour avoir une idée intuitive du fonctionnement de ces réseaux de neurones, réfléchissons à la façon dont nous reconnaissons les choses. Par exemple, si nous voyons un visage, nous le reconnaissons parce qu'il a des oreilles, des yeux, un nez, des cheveux, etc. Ensuite, pour décider si quelque chose est un visage, nous le faisons comme si nous avions des boîtes mentales de vérification de la caractéristique que nous marquons. Parfois, un visage n'a pas d'oreille parce qu'il est recouvert de poils, mais nous le classons également avec une certaine probabilité comme visage car nous voyons les yeux, le nez et la bouche. En fait, on peut le voir comme un classifieur qui prédit une probabilité que l'image d'entrée soit un visage ou non.

Mais en réalité, nous devons d'abord savoir à quoi ressemble une oreille ou un nez pour savoir s'ils sont dans une image; c'est-à-dire qu'il faut préalablement identifier les lignes, les contours, les textures ou les formes similaires à celles contenant les oreilles ou les nez que nous avons vus auparavant. Et c'est à cela que sont confiées les couches d'un réseau de neurones convolutionnels.

Mais identifier ces éléments ne suffit pas pour pouvoir dire que quelque chose est un visage. Nous devons également être en mesure d'identifier comment les parties d'un visage se rencontrent, les tailles relatives, etc. sinon, le visage ne ressemblerait pas à ce à quoi nous sommes habitués. Visuellement, la figure 10 présente une idée intuitive de ce que les couches apprennent.

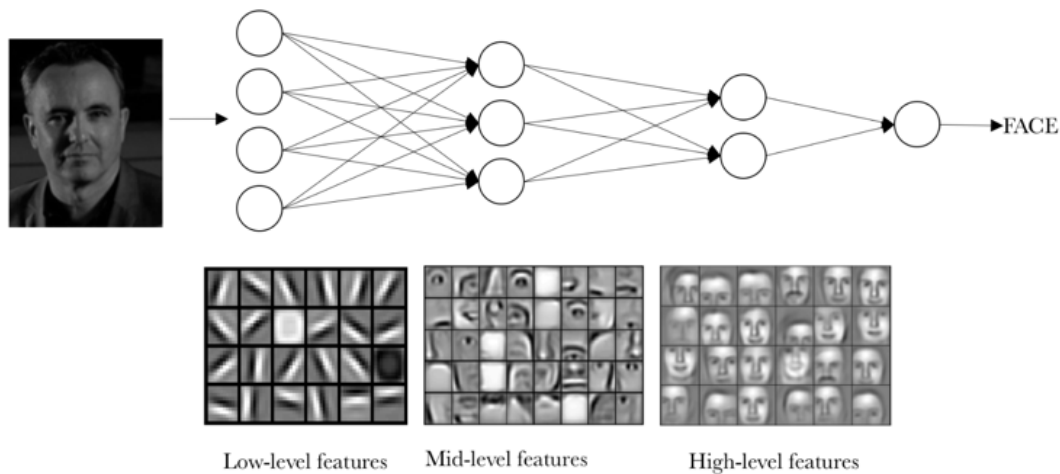


FIGURE 10 – Exemple qui montre les étapes d'apprentissage d'un réseau de neurones.

L'idée que nous voulons donner à cet exemple visuel est que, dans la réalité, dans un réseau de neurones convolutionnels, chaque couche apprend différents niveaux des motifs. On peut imaginer qu'avec des réseaux à plusieurs couches, il est possible d'identifier des structures plus complexes dans les données d'entrée.

3.2 Les composants d'un réseau CNN

Maintenant que nous avons une vision intuitive de la façon dont les réseaux de neurones de convolutionnels classifient une image, nous allons introduire les différents constituants qui définissent les réseaux CNN.

3.2.1 Couche de convolution (CONV)

De manière intuitive, on pourrait dire que le but principal d'une couche convolutive est de détecter des caractéristiques dans des images telles que des contours, des lignes, des couleurs, etc. C'est une propriété très intéressante, car une fois qu'elle a appris une caractéristique à un point spécifique de l'image, elle peut la reconnaître plus tard dans n'importe quelle partie de l'image.

Une autre caractéristique importante est que les couches convolutives peuvent apprendre les hiérarchies spatiales des modèles en préservant les relations spatiales [5]. Par exemple, une première couche de convolution peut apprendre des éléments de base tels que des contours, et une seconde couche de convolution peut apprendre des motifs composés d'éléments de base appris dans la couche précédente. Et ainsi de suite jusqu'à ce qu'il apprenne des modèles très complexes (voir figure 11). Cela permet aux réseaux CNN d'apprendre efficacement des concepts visuels de plus en plus complexes et abstraits.

En général, les couches de convolutions fonctionnent sur des tenseurs 3D, appelés « Cartes de Caractéristiques » ou bien « Carte d'Activation », avec deux axes spatiaux de hauteur et de largeur, ainsi que sur un axe de canal également appelé profondeur. Pour une image couleur RVB, l'axe des profondeurs a une dimension de 3 car l'image comporte trois canaux : rouge, vert et bleu. Pour une image en noir et blanc, la dimension de l'axe de profondeur est 1 (niveau de gris).

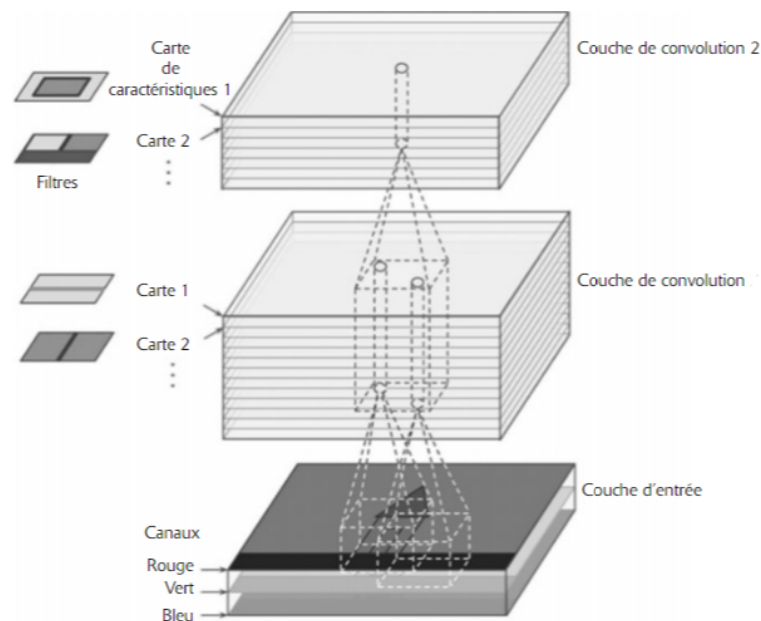


FIGURE 11 – Couches de convolution avec plusieurs caractéristiques et images RVB à l'entrée [5].

La convolution peut être considérée comme une somme pondérée entre deux signaux (en termes de jargon de traitement de signaux) ou de fonctions (en termes de mathématiques). En traitement d'image, pour calculer la convolution à un emplacement particulier (x, y), nous extrayons un bloc de la taille (k * k) de l'image centrée à l'emplacement (x, y). Nous multiplions ensuite les valeurs de ce bloc élément par élément avec le filtre de convolution (également de taille (k * k)), puis nous les ajoutons toutes pour obtenir une seule sortie [27]. C'est tout! Notez que « k » est appelé la taille du noyau.

Un exemple d'opération de convolution sur une matrice de taille 5 x 5 avec un noyau de taille 3 x 3 est présenté dans la figure 12 :
Le noyau de convolution glisse sur toute la matrice pour obtenir une carte d'activation.

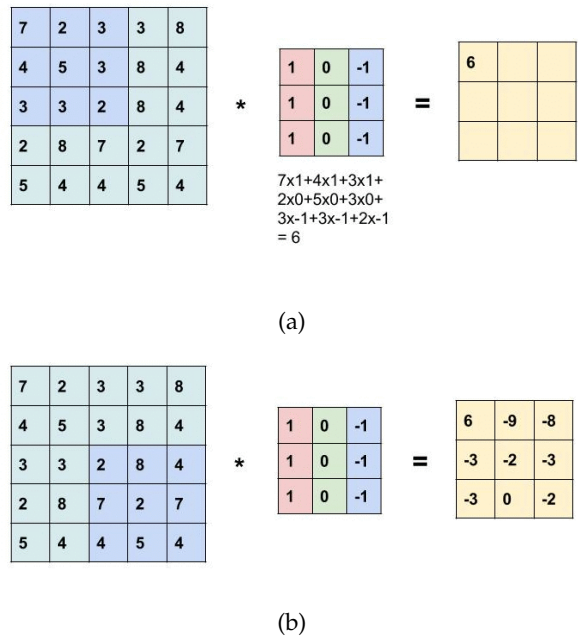


FIGURE 12 – Exemple d'une opération de convolution [6].

Regardons un exemple concret et comprenons les termes. Supposons que l'image d'entrée soit de taille 32x32x3. Ce n'est rien qu'un tableau 3D de profondeur 3. Tout filtre de convolution que nous définissons sur cette couche doit avoir une profondeur égale à la profondeur de l'entrée. Nous pouvons donc choisir des filtres de convolution de profondeur 3 (par exemple 3x3x3 ou 5x5x3 ou 7x7x3, etc.). Choisissons un filtre de convolution de taille 3x3x3. Ainsi, en référence à l'exemple ci-dessous (voir figure 13), le noyau de convolution sera ici un cube au lieu d'un carré.

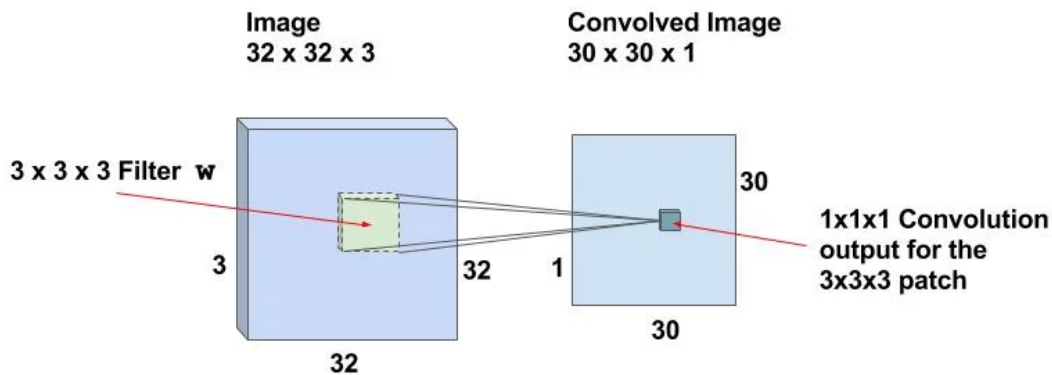


FIGURE 13 – Convolution sur image d'entrée [6].

Si nous pouvons effectuer l'opération de convolution en faisant glisser le filtre $3 \times 3 \times 3$ sur l'ensemble de l'image de taille $32 \times 32 \times 3$, nous obtiendrons une image de sortie de taille $30 \times 30 \times 1$. En effet, l'opération de convolution n'est pas définie pour une bande de 2 pixels de large autour de l'image. Nous devons nous assurer que le filtre est toujours à l'intérieur de l'image. Donc, 1 pixel est retiré de la gauche, de la droite, du haut et du bas de l'image.

Remarques :

- Les mêmes filtres sont glissés sur toute l'image pour trouver les caractéristiques pertinentes. Afin de trouver les motifs qui se ressemblent.
- Nous voudrions faire remarquer aussi que l'hypothèse retenue est que la fenêtre avance d'un pixel, horizontalement et verticalement, au début d'une nouvelle ligne. Par conséquent, à chaque étape, la nouvelle fenêtre chevauche la précédente sauf dans la première ligne de pixels. Il faut savoir que différents « Pas de Progression » ou « Stride » peuvent être utilisés. Dans les réseaux CNN, on peut également appliquer une technique de « Remplissage des Zéros » autour de la marge de l'image pour améliorer le balayage effectué avec la fenêtre glissante. Le paramètre permettant de définir cette technique est appelé « Padding ».

3.2.1.1 Poids et Biais partagés (Weights and Biases)

Cependant, l'importance des réseaux convolutifs est que nous utilisons le même filtre, ça veut dire que la matrice partage les mêmes « Poids W » (Paramètres de filtre) et le même « Biais B » (constant ajouté après convolution) pour tous les neurones de la couche cachée [5] : Dans notre exemple, il y a $30 \times 30 = 900$ neurones car il existe autant d'emplacements où le filtre $3 \times 3 \times 3$ peut être appliqué. Contrairement aux réseaux de neurones traditionnels où les poids et les biais des neurones sont indépendants les uns des autres, dans le cas de CNN, les neurones correspondant à un filtre dans une couche partagent les mêmes poids et biais.

3.2.1.2 Pas de progression (Stride)

C'est combien de pixel la fenêtre (le filtre) glisse à la fois. On peut également faire glisser la fenêtre de 1 à plusieurs pixels.

3.2.1.3 Remplissage par Zéro (Padding)

Comme vous pouvez le constater, après chaque convolution, la taille de la sortie est réduite (comme dans ce cas (voir figure 14), nous passons de 32×32 à 30×30). Pour plus de commodité, il est de pratique courante d'ajouter des zéros à la limite de l'image d'entrée de sorte que la sortie ait la même taille que l'entrée. Ainsi, dans cet exemple, si nous ajoutons un remplissage de taille 1 des quatre côtés de la couche d'entrée, la taille de la couche de sortie sera de 32×32 , ce qui simplifie également la mise en œuvre. Supposons que vous avez une entrée de taille $N \times N$, un filtre de taille F et que vous utilisez un Stride S et qu'un remplissage de zéro P est ajouté à l'image d'entrée. La sortie sera alors de taille $M \times M$ où,

$$M = \frac{N - F + 2P}{S} + 1 \quad (1)$$

0	0	0	0	0	0
0	35	19	25	6	0
0	13	22	16	53	0
0	4	3	7	10	0
0	9	8	1	3	0
0	0	0	0	0	0

FIGURE 14 – Opération de remplissage par zéro.

3.2.1.4 Cartes de caractéristiques (Feature Map)

Pour une image d'entrée $32 \times 32 \times 3$ et une taille de filtre de $3 \times 3 \times 3$, nous avons 30 emplacements $30 \times 30 \times 1$ et un neurone correspond à chaque emplacement. Ensuite, les sorties ou activations $30 \times 30 \times 1$ de tous les neurones sont appelées cartes de caractéristiques (cartes d'activations). La carte de caractéristiques d'une couche sert d'entrée à la couche suivante.

3.2.1.5 Filtres multiples

En règle générale, nous utilisons plus d'un filtre dans une couche de convolution. Si nous utilisons 32 filtres, nous aurons une carte d'activation de taille $30 \times 30 \times 32$. voir la figure 15.

Notez que tous les neurones associés au même filtre partagent les mêmes poids et biais. Ainsi, le nombre de poids lors de l'utilisation de 32 filtres est simplement $3 \times 3 \times 3 \times 32 = 288$ et le nombre de biais est de 32.

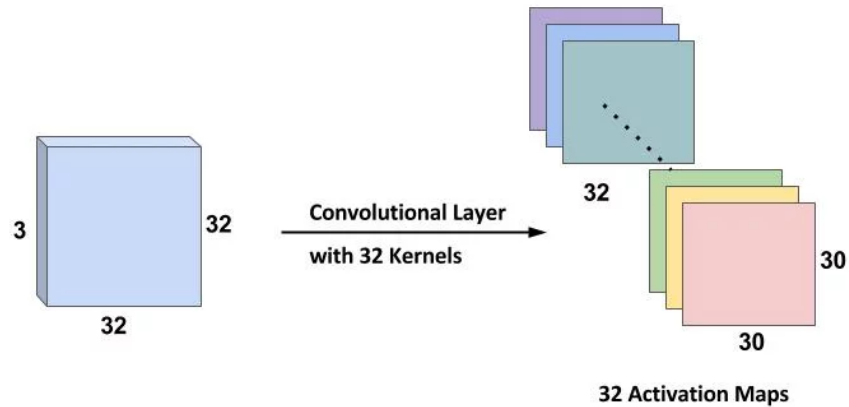


FIGURE 15 – Cartes de caractéristiques formés de 32 filtres [6].

Résumé : Pour résumer la couche de convolution :

- Accepte un volume de taille « une image » de « largeur $L1$ * hauteur $H1$ * profondeur $P1$ ».
- Nécessite quatre hyper paramètres :
 1. Nombre de filtre K (ex : 32 , 64 , 128 ...)
 2. Taille de filtre F (ex : 3 , 5 , 7 ...)
 3. Le Pas de progression (Stride) S (ex : 1, 2, 3 ...)
 4. Quantité de remplissage zéro (Padding) P (ex : 1, 2 ...)
- Produit un volume de taille « une carte de caractéristique » de ($L2 * H2 * P2$), tels que :
 - $L2 = (L1 - F + 2P) / S + 1$
 - $H2 = (H1 - F + 2P) / S + 1$
 - $P2 = K$
- Avec le partage de paramètres, il introduit un nombre de poids égal à « $F \times F \times P1$ » par filtre, pour un total des paramètres égal à : « $(F \times F \times P1) \times K$ » poids en plus de ' K ' biais.
- Chaque carte d'activation est le résultat d'une opération de convolution entre 1 filtre et l'image d'entrée. (Plusieurs filtres résultent = plusieurs cartes d'activations).

3.2.2 Couche de Pooling (Pool)

La couche de Pooling (Regroupement) est principalement utilisée immédiatement après la couche de convolution afin de réduire la taille spatiale (uniquement la largeur et la hauteur, et non la profondeur). Cela réduit le nombre de paramètres, donc le calcul est réduit [5]. Il est donc fréquent d'insérer périodiquement une couche de Pooling entre deux couches convolutives successives pour éviter les surajustements (le surajustement est expliqué dans la partie 2.3.2).

La forme de pooling la plus courante est le « maximum pooling », dans lequel nous prenons un filtre de taille P et appliquons l'opération maximale sur la partie dimensionnée de l'image.

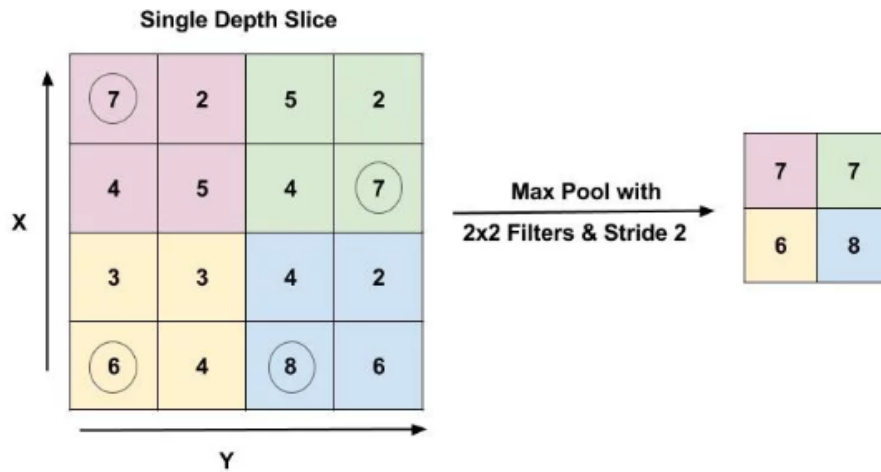


FIGURE 16 – Opération de max Pooling [6].

La figure 16 illustre la couche de max pool avec une taille de filtre de 2×2 et un Stride de 2. La sortie correspond à la valeur maximale dans une région 2×2 , représentée par des chiffres encadrés.

La mise en commun la plus courante est effectuée avec le filtre de taille 2×2 avec une Stride de 2. Elle réduit essentiellement la taille de l'entrée de moitié.

Average-pooling ou bien regroupement moyen peut également être utilisé au lieu du regroupement maximal, où chaque groupe de points d'entrée est transformé en la valeur moyenne du groupe de points au lieu de sa valeur maximale. Mais en général, le max-pooling a tendance à fonctionner mieux que les solutions alternatives.

3.2.3 Couche de ReLU

ReLU signifie unité linéaire rectifiée et constitue un type de fonction d'activation. Mathématiquement, il est défini comme étant $y = \max(0, x)$. Visuellement, cela ressemble au graphe de la figure 17.

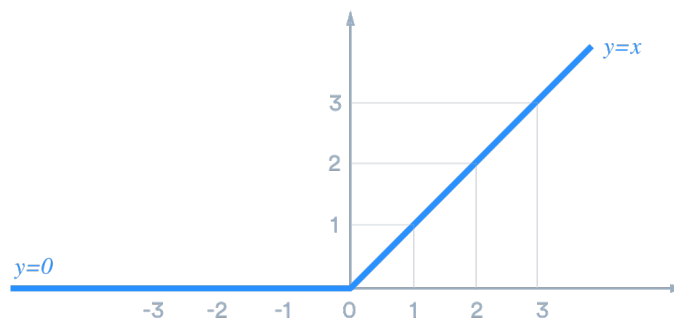


FIGURE 17 – Fonction ReLU [5].

ReLU est la fonction d'activation la plus couramment utilisée dans les réseaux de neurones, en parti-

culier dans les CNN. Si vous ne savez pas quelle fonction d'activation utiliser sur votre réseau, ReLU est généralement un bon premier choix.

ReLU est linéaire (identité) pour toutes les valeurs positives et nul pour toutes les valeurs négatives [5]. Cela signifie que :

- C'est peu coûteux à calculer car il n'y a pas de calculs compliqués. Le modèle peut donc prendre moins de temps à s'entraîner.
- Cela converge plus vite. La linéarité signifie que la pente ne se stabilise pas, ni ne "sature" lorsque x devient grand.

3.2.4 Couche Entièrement Connectée (Fully Connected FC)

Après la dernière couche de Pooling, tout les cartes de caractéristiques vont être vectorisés et placés dans un seul vecteur grâce à l'opération d'aplatissement (flattening), Les caractéristiques que nous avons distillées au cours des étapes précédentes sont codées dans ce vecteur.

Par la suite, ces données sont introduites à des couches « entièrement connectées » (Fully Connected Layers), Le terme « entièrement connecté » implique que chaque neurone de la couche précédente est connecté à chaque neurone de la couche suivante.

La sortie des couches de convolution et de Pooling représente des caractéristiques de haut niveau de l'image d'entrée. L'objectif de la couche FC est d'utiliser ces fonctionnalités pour classer l'image d'entrée en différentes classes en fonction de la base de données d'apprentissage [5].

Par exemple, la tâche de classification des images que nous avons l'intention d'effectuer comporte deux sorties possibles, comme illustré dans la figure 18 :

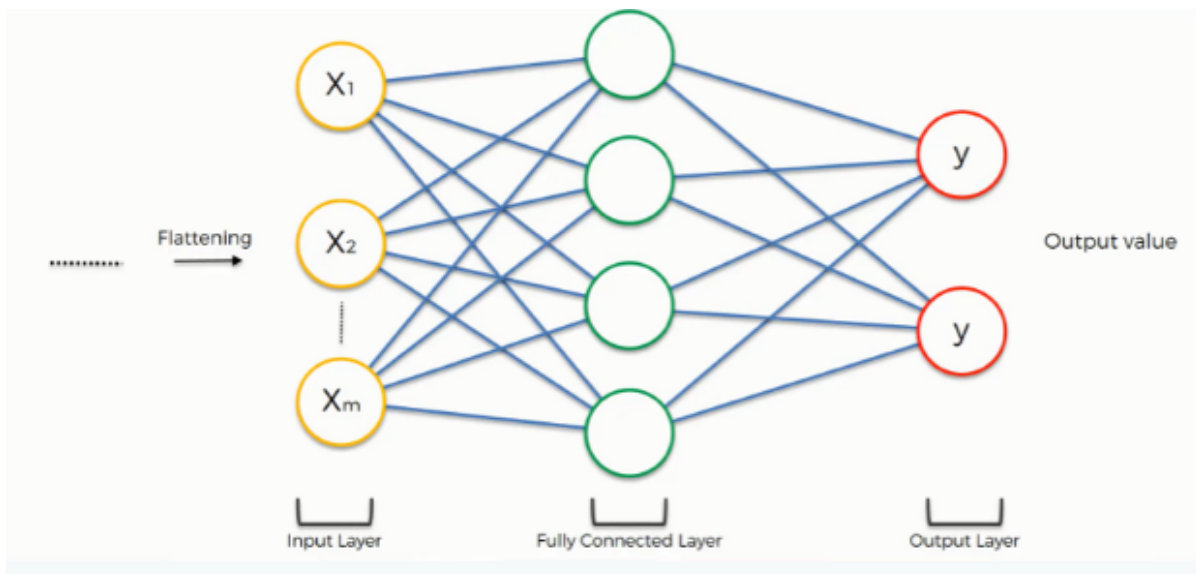


FIGURE 18 – Couche entièrement connectée « fully connected » [7].

Afin de comprendre comment cela va se dérouler, nous devons vérifier les poids placés sur chaque synapse reliant cette classe afin de pouvoir identifier les attributs / caractéristiques les plus pertinents.

Ce processus des couches FC fonctionne pratiquement comme suit :

- Le neurone dans la couche FC détecte une certaine caractéristique ; dis, un nez par exemple.
- Il conserve sa valeur.
- Il communique cette valeur aux classes "chien" et "chat".
- Les deux classes vérifient la fonctionnalité et décident si elle est pertinente pour elles.

Dans notre exemple, le poids placé sur la synapse nez-chien est élevé (1,0), ce qui signifie que le réseau est convaincu qu'il s'agit du nez d'un chien comme il est montré dans la figure 19 :

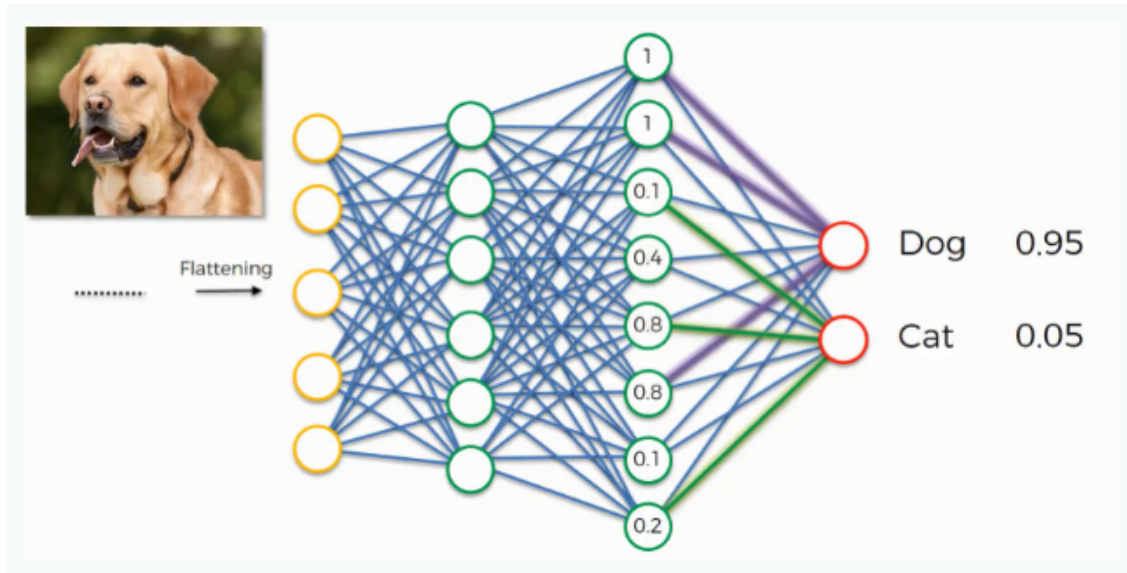


FIGURE 19 – Exemple d’une prédiction à travers les Couches « fully connected » [7].

La somme des probabilités en sortie de la couche FC est 1. Cela est garanti en utilisant « [Softmax](#) » comme fonction d’activation dans la couche en sortie de la couche FC. La fonction Softmax prend un vecteur de scores arbitraires à valeurs réelles et le réduit à un vecteur de valeurs comprises entre zéro et un qui égal à un.

3.3 Fonctions utilisées dans les réseaux CNN

3.3.1 Fonction de Perte/ Coût (LOSS)

La fonction de perte indique l’écart entre la sortie prédite et la sortie réelle, et se situe dans la dernière couche d’un réseau de neurones, elle aide à déterminer le Taux d’erreur ou Pourcentage d’erreur durant et après un entraînement [5].

la fonction de perte qui est couramment utilisée est l’erreur quadratique moyenne :

$$EQM = (1/n) * \sum_{t=1}^n e_t^2 \quad (2)$$

- n : est le nombre de classes.
- e : est la différence entre la valeur vérité terrain et le résultat prédit.

Par exemple, s'il y a 3 classes au total, pour une image avec l'étiquette 0, la vérité terrain peut être représentée par un vecteur [1, 0, 0] et la sortie du réseau de neurones peut être [0.5, 0.2, 0.3], alors le EQM peut être calculé par :

$$\frac{(0.5 - 1)^2 + (0.2 - 0)^2 + (0.3 - 0)^2}{3} = 1.26666$$

3.3.2 Surapprentissage (Overfitting)

Les réseaux de neurones profonds possèdent en général des dizaines de milliers de paramètres, voire des millions. Avec une telle quantité, un réseau dispose d'une liberté incroyable et peut s'adapter à une grande diversité de jeux de données complexes. Mais cette grande souplesse signifie également qu'il est sujet au surapprentissage du jeu d'entraînement.

Le surapprentissage survient lorsque notre modèle devient vraiment capable de classer et de prédire des données incluses dans l'ensemble de la base de données de l'entraînement, mais n'est pas si bon pour classer des données pour lesquelles ils n'étaient pas entraînés. Le modèle a donc trop bien ajusté les données dans l'ensemble de l'entraînement.

Dans cette partie, on va présenter quelques-unes des techniques pour réduire le surajustement, qui sont :

3.3.2.1 Augmentation des données (data augmentation)

L'une des principales raisons du surapprentissage est que vous ne disposez pas de suffisamment de données pour former votre réseau, donc l'augmentation des données est un autre moyen très efficace pour lutter contre le surapprentissage. C'est le processus de création artificielle d'un plus grand nombre d'images à partir des images que vous avez déjà, cela veut dire apporter des modifications mineures à votre base de données existant. Des changements mineurs tels que des retournements(flips), des translations ou des rotations (voir figure 20).

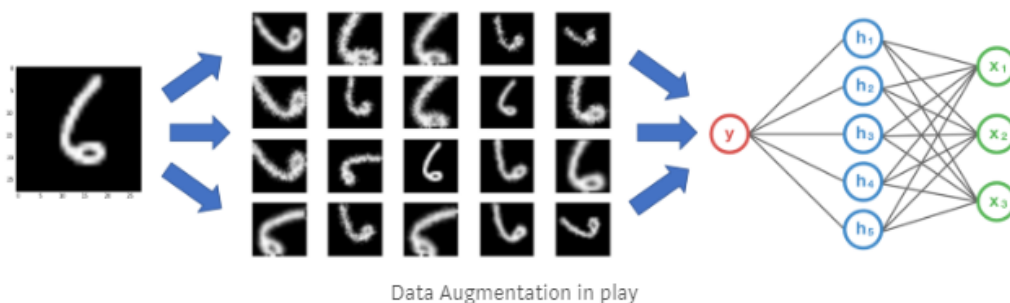


FIGURE 20 – Opération d'augmentation de données [8].

Un réseau de neurones convolutionnel est capable de classer des objets de manière robuste, même s'ils sont placés dans différentes orientations, le réseau est doté d'une propriété appelée "invariance". Plus spécifiquement, un CNN peut être invariant vis-à-vis de la translation, changement de la taille ou de l'éclairage (ou une combinaison de ce qui précède). C'est essentiellement le principe de l'augmentation des données.

3.3.2.2 Dropout

Cette technique est très efficace contre le surapprentissage. En fait, l'algorithme de dropout est relativement simple. A chaque étape d'entraînement, chaque neurone (y compris les neurones d'entrée, mais pas les neurones de sortie) a une probabilité P d'être temporairement « éteint ». Autrement dit, il sera totalement ignoré au cours de cette étape d'entraînement, mais il pourra être actif lors de la suivante (voir figure 11).

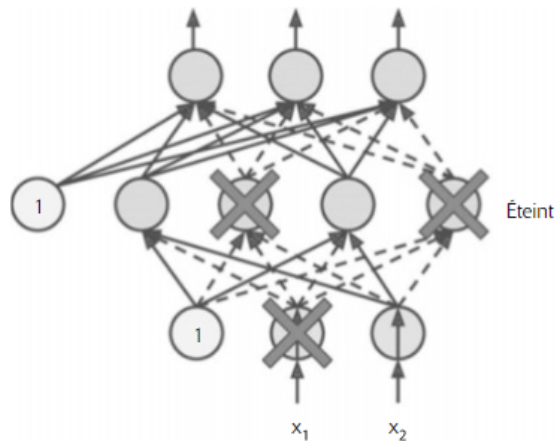


FIGURE 21 – Régularisation par Dropout.

L'hyper-paramètre P est appelé « taux d'extinction » (dropout rate) et il est en général fixé à 50 pourcent. Après un entraînement, les neurones ne sont plus jamais éteints.

Pour bien comprendre la puissance du dropout, il faut réaliser que chaque étape d'entraînement génère un réseau de neurone unique. Puisque chaque neurone peut être présent ou absent, il existe un total de 2^N réseaux possible (ou N est le nombre total de neurones éteignables). Ce nombre est tellement énorme qu'il est quasi impossible que le même réseau de neurones soit produit à deux reprises. Après 10000 étapes d'entraînement, nous avons en réalité entraîné 10000 réseaux de neurones différents (chacun avec une seule instance d'entraînement). Ces réseaux de neurones ne sont évidemment pas indépendants, car ils partagent beaucoup de leurs poids, mais ils n'en restent pas moins tous différents. Le réseau de neurones résultant peut-être vu comme un ensemble moyen de tous ces réseaux de neurones plus petits.

3.3.3 Descente de gradient (Vanishing gradient)

La descente de gradient est un algorithme d'optimisation très général, capable de trouver des solutions optimales à un grand nombre de problèmes. L'idée essentielle de la descente de gradient est de corriger petit à petit les paramètres (poids et biais) dans le but de minimiser la fonction de coût (LOSS) (voir figure 22).

Elle calcule le gradient de la fonction de coût au point θ , puis progresse en direction du gradient descendant, on atteint le minimum lorsque le gradient est nul.

On commence par donner une valeur aléatoire à θ , (Initialisation aléatoire); puis on l'améliore progressivement pour décroître la fonction de coût.

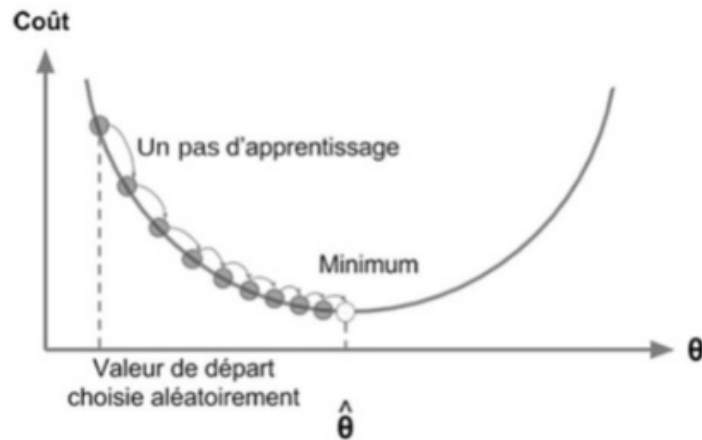


FIGURE 22 – Descente de gradient.

La dimension des pas de progression est déterminée par l'hyper paramètre Learning Rate (taux d'apprentissage).

Il ne faut pas que le taux d'apprentissage soit trop petit, car l'algorithme prendra beaucoup de temps et un grand nombre d'itérations pour converger, au même temps, il faut pas qu'il soit trop élevé, car on risque de dépasser le point le plus bas, et notre algorithme risque de diverger.

3.4 Etude d'un exemple de réseau CNN

Tout d'abord, il faut savoir que les réseaux CNN en général ont une architecture standard qui est montrée dans la figure 23. , ça veut dire qu'il y a une image à l'entrée du réseau, suivi d'une succession des couches CONV et POOL dont l'activation est RELU , et à la fin, on trouve des couches entièrement connectées qui nous donnent les classes des images entraînées à leurs dernière couche.

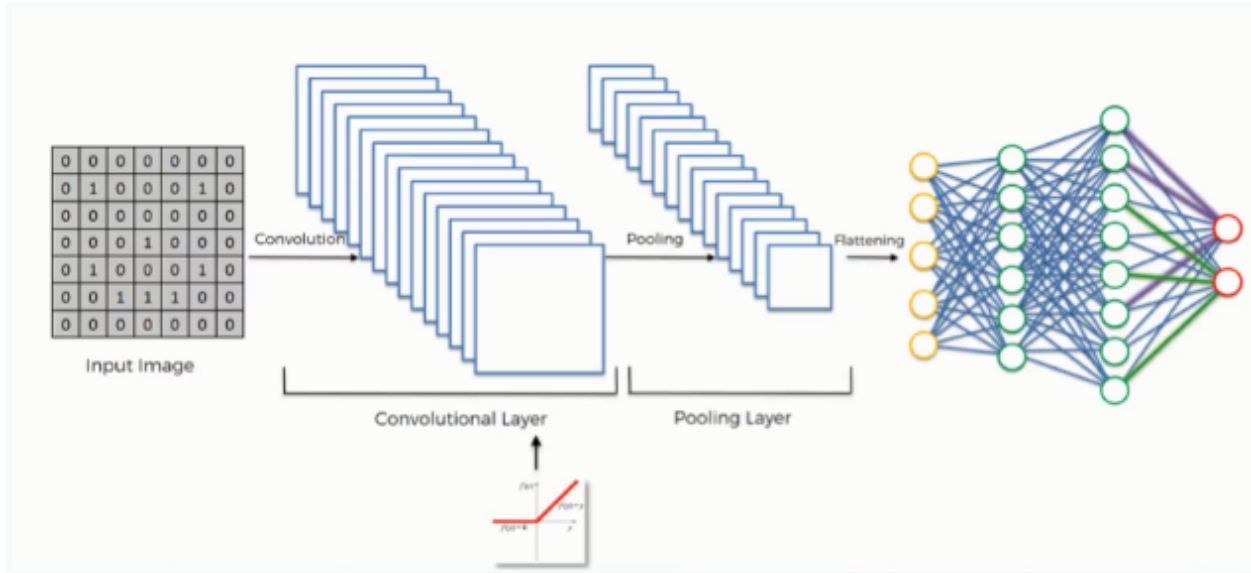
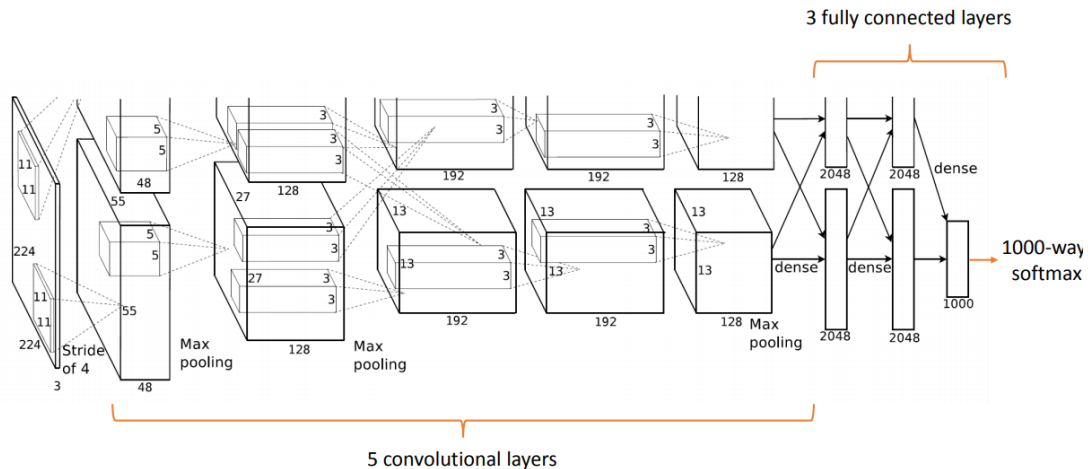


FIGURE 23 – Architecture standard d'un réseau CNN.

Dans cette partie, nous allons étudier l'un parmi les réseaux CNN connus qui existent, et montrer tout ce qu'on a vu dans la partie précédente.

Le réseau choisi est AlexNet qui a l'Architecture suivante :

Architecture



[A. Krizhevsky, I. Sutskever, G.E. Hinton, *ImageNet Classification with Deep Convolutional Neural Networks*, 2012]

FIGURE 24 – Architecture du réseau AlexNet.

Comme il est illustré dans la figure 24, le réseau AlexNet contient 8 couches : 5 de convolution (CONV), 3 couches de Pooling (Pool) et 3 couches entièrement connectées (FC), et le réseau génère une sortie qui contient 1000 classe.

Les couches de ce réseau et leurs paramètres sont successivement comme suit :

- L'Entrée : Image de [227x227x3] (RVB).
- **CONV1** : couche de convolution avec 96 filtre de taille [11x11], et un Pas de progression (Stride=4), et dans cette couche, il n'y a pas de remplissage par zéro (Pad=0) et l'activation utilisé est RELU. La carte de caractéristique générée est de taille [55x55x96].
- **MAX POOL1** : filtre de [3x3], Stride=2 et carte de caractéristique [27x27x96].
- **CONV2** : 256 filtre [5x5], Stride=1, Pad=2, carte de caractéristique [27x27x256] et Activation=RELU.
- **MAX POOL2** : filtre de [3x3], Stride=2 et carte de caractéristique [13x13x256].
- **CONV3** : 384 filtre [3x3], Stride=1, Pad=1, carte de caractéristique [13x13x384] et Activation=RELU.
- **CONV4** : 384 filtre [3x3], Stride=1, Pad=1, carte de caractéristique [13x13x384] et Activation=RELU.
- **CONV5** : 256 filtre [3x3], Stride=1, Pad=1, carte de caractéristique [13x13x256] et Activation=RELU.
- **MAX POOL3** : filtre de [3x3], Stride=2 et carte de caractéristique [6x6x256].
- **FC1** : contient 4096 neurone. Activation=RELU.
- **FC2** : contient 4096 neurone. Activation=RELU.
- **FC3** : contient 1000 neurone, Activation=SoftMax qui va générer le score de chacune des 1000 classes.

Cette architecture a remporté le défi ImageNet en 2012, Elle est arrivée à un taux d'erreur Top-5 de 17 pourcent tandis que le second n'a obtenu que 26 pourcent.

3.5 Conclusion

Puisque nous avons terminé avec ce chapitre, résumons rapidement ce que nous avons appris sur les réseaux de neurones convolutionnels. Les notes suivantes nous montrent tout le processus de création et d'optimisation d'un réseau CNN que nous avons abordé auparavant.

- Nous commençons avec une image d'entrée.
- Nous appliquons des filtres à l'image, ce qui nous donne des cartes de caractéristiques (feature maps).
- Nous appliquons ensuite la fonction Relu, qui a pour rôle de minimiser le temps d'entraînement du réseau et facilite le calcul.
- L'image est maintenant prête pour la phase de pooling, dont l'objectif est dimensionner la taille et extraire les caractéristiques pertinents de l'image, ainsi que la réduction du nombre de paramètres du réseau .
- Une fois l'opération de pooling terminée, nous obtenons une carte de caractéristiques qui est ensuite aplati et mise à l'entrée des couches entièrement connectées (FC).

Tout au long de ce processus, les blocs de construction du réseau, tels que les poids et les cartes de caractéristiques, sont formés et modifiés à plusieurs reprises afin que le réseau atteigne les performances optimales lui permettant de classer les images et les objets aussi précisément que possible.

À ce stade, nous avons acquis toutes les connaissances nécessaires, dont nous avons besoin pour procéder aux applications plus pratiques du concept des réseaux de neurones convolutionnels.

4 Chapitre 4 Architectures proposées

4.1 Introduction

Pour la classification des images, les réseaux de neurones convolutionnels sont les plus utilisés dans ce domaine, du fait de la grande efficacité qu'ils ont montré, de plus, le chemin court qui est suivi lors de la classification, contrairement aux méthodes classiques de classification où l'image en entrée nécessite un prétraitement, après une segmentation, ensuite une extraction des variables et en fin la classification. Les réseaux CNN ont raccourci ce cheminement en deux étapes seulement, l'étape d'extraction des paramètres en utilisant un grand nombre de filtres ce qui permet de balayer et extraire les variables nécessaires, ensuite vient l'étape de classification qui détermine la classe de l'image qui est en entrée, ces deux étapes sont faites de manière automatique à l'inverse des méthodes classiques.

4.2 Choix de l'architecture

Le problème majeur dans les réseaux CNN est de choisir une architecture adaptée à la base de données afin d'obtenir un taux de précision élevée à la sortie. Le défi est de sélectionner les hyper-paramètres qui définissent le réseau choisi, qui sont :

- Le nombre et la taille des couches de convolution avec le pas de progression « Stride » et le remplissage par zéro « Padding ».
- la taille des couches de POOL.
- le nombre des couches FC et le nombre des neurones utilisées dans chacune.
- le pourcentage de Dropout et le type d'Activation sélectionné (Relu, Soft-max ...).

Remarque : Il faut savoir aussi que le nombre des images constituant la base de données joue un rôle important dans l'entraînement du réseau, de manière que plus le nombre des images est important, plus la précision atteinte est meilleur après la phase d'apprentissage. Dans le cas où le nombre des images de la base de données n'est pas aussi important, il est nécessaire d'utiliser l'Augmentation de données afin d'augmenter le nombre des images présentés en entrée de réseau, ce processus est expliqué dans le chapitre précédent.

Tous ces paramètres cités nécessitent un tel savoir sur les réseaux CNN, afin de les appliquées correctement et de choisir leurs valeurs de façon qu'ils vont augmenter la précision à la sortie du réseau entraîné. Il est improbable d'atteindre la précision maximale dès le premier entraînement, donc le changement des paramètres est recommandé, ce qui fait la différence entre une architecture et une autre.

4.3 Architectures proposées

Au cours de nos expérimentations, nous avons divisé notre travail en 2 étapes, la 1er est de concevoir de simples architectures avec paramètres différents, puis lancer l'entraînement pour prendre une idée sur le comportement du réseau avec la base de données et voir quelle architecture va donner la meilleure précision. La 2eme étape est de développer une architecture un peu profonde suivant les résultats de la 1ere étape, ça veut dire en prend le modèle qui donne une bonne précision et rajouter des couches de CONV et POOL et autres paramètres pour obtenir un maximum de précision.

Etape 1 Dans cette étape, nous avons créé des modèles de simples architectures (2 ou 3 couches de convolution avec MaxPool dans chacune), où on a joué sur les paramètres du chaque modèle, dont le nombre et la taille des filtres utilisés, aussi que la taille de la couche POOL et FC, et ça nous a donné 4 modèles qui sont détaillées par la suite :

4.3.1 Modèle 1

Le premier modèle qui est présenté dans la figure 25 est composé de 2 couches de convolution, 2 couches de MaxPool et deux couches entièrement connectée FC.

L'image en entrée est de taille $50 \times 50 \times 3$, elle passe d'abord par la première couche de convolution, qui est composée de 32 filtres de taille $15 \times 15 \times 3$. Chacune des couches de convolution est suivie d'une fonction d'activation ReLU, cette fonction force les neurones à retourner des valeurs positives ; après cette convolution, 32 carte de caractéristique de taille 36×36 sont créés.

Les 32 carte de caractéristique qui sont obtenus auparavant, ils sont mis en entrée de la première couche de MaxPool qui est composée de 32 filtres de 2×2 , ce qui permet de réduire la taille de l'image ainsi que les paramètres les moins importants et le temps de calcul. À la sortie de cette couche, nous aurons 32 carte de caractéristique de taille 18×18 .

Le même processus est répété avec la couches de convolution et MaxPool suivantes, ces couches sont composées de 64 filtres de taille 3×3 et 2×2 successivement, la fonction d'activation ReLU est appliquée toujours. À la sortie de cette couche, nous aurons 64 carte de caractéristique de taille 8×8 . Le vecteur de caractéristiques issu a une dimension de 4096.

Après l'étape d'extraction des caractéristiques, nous utilisons un réseau de neurones composé de deux couches entièrement connectée FC. La première couche a 128 neurones où la fonction d'activation utilisée est le ReLU, et la deuxième couche est un Softmax qui permet de calculer la distribution de probabilité des 2 classes Maligne ou Bénigne.

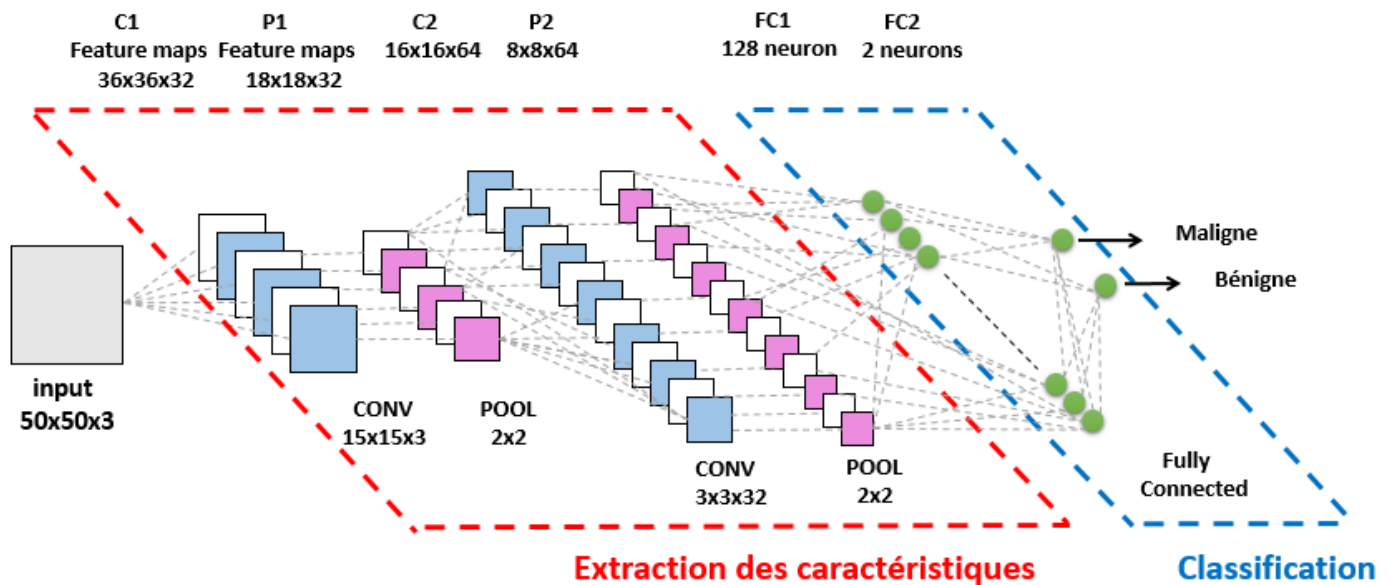


FIGURE 25 – Architecture du modèle 1 .

4.3.2 Modèle 2

Le deuxième modèle est composé de trois couches de convolution, trois couches de MaxPool et de deux couches FC (voir figure 26).

L'image en entrée est de taille 50*50, elle passe d'abord par la première couche de convolution. Cette couche est composée de 32 filtres de taille 15*15, la fonction d'activation Relu est utilisée, après cette convolution 32 carte de caractéristiques de taille 36*36 seront créés. Le MaxPool appliqué est de taille 2x2, à la sortie de cette couche, nous aurons 32 carte de caractéristiques de taille 18*18.

Le même processus est répété avec les couches de convolution et MaxPool suivantes, tel que la 2eme couche de CONV est constituée de 32 filtre 5x5, la 3eme est de 64 filtre de 3x3 et les deux couches de MaxPool sont de taille 2x2. Le résultat est 64 cartes de caractéristiques de taille 3x3, le vecteur résultant est de dimension de 576.

Comme le modèle précédent, 2 couches FC sont utilisées, la 1ere est de 128 neurones, la 2eme est un Softmax de 2 classe maligne et bénigne.

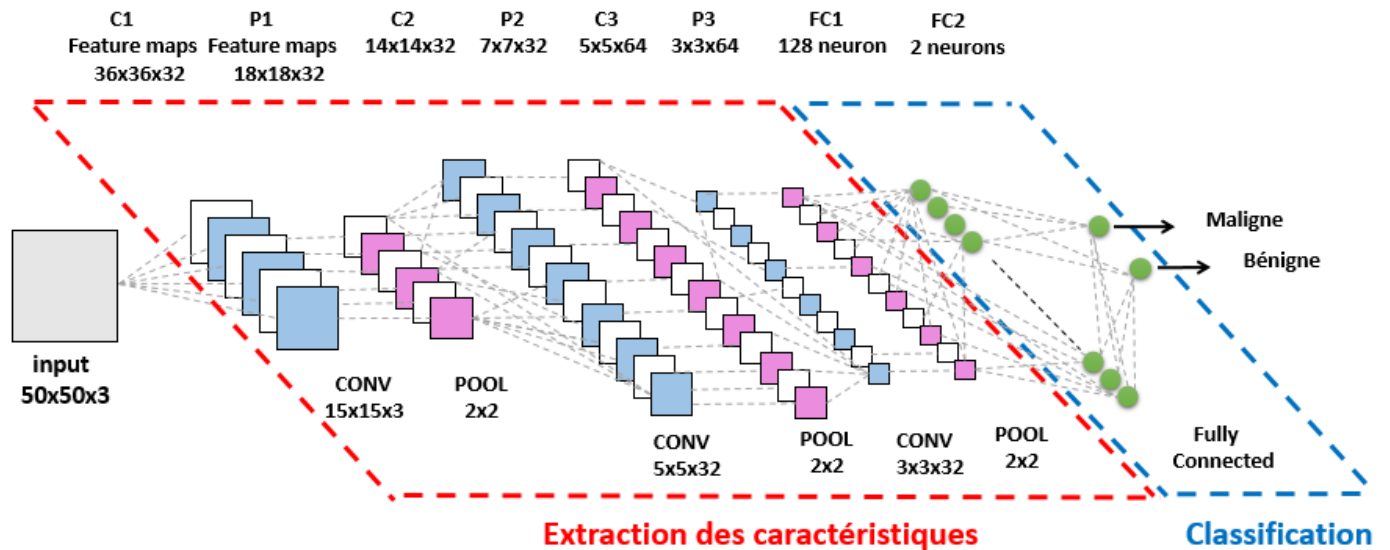


FIGURE 26 – Architecture du modèle 2 .

4.3.3 Modèle 3

Le troisième modèle est composé de deux couches de convolution, deux couches de MaxPool et de deux couches FC (voir la figure 27).

L'image en entrée est de taille 50*50, elle passe d'abord par la première couche de convolution. Cette couche est composée de 32 filtres de taille 15*15, la fonction d'activation Relu est utilisée, après cette convolution 32 carte de caractéristiques de taille 36*36 seront créés, Le MaxPool de taille de 2x2 est appliqué. À la sortie de cette couche, nous aurons 32 carte de caractéristiques de taille 18*18.

Le même processus est répété avec les couches de convolution et MaxPool suivantes, tel que la couche de CONV est constituée de 32 filtres de taille 11x11, et la couche MaxPool d'une taille de 2x2. Le résultat est 32 carte de caractéristiques de taille 4x4, le vecteur résultant est de dimension de 512.

Deux couches FC sont utilisées, la 1ere est de 128 neurones, la 2eme est un Softmax de 2 classe maligne et bénigne.

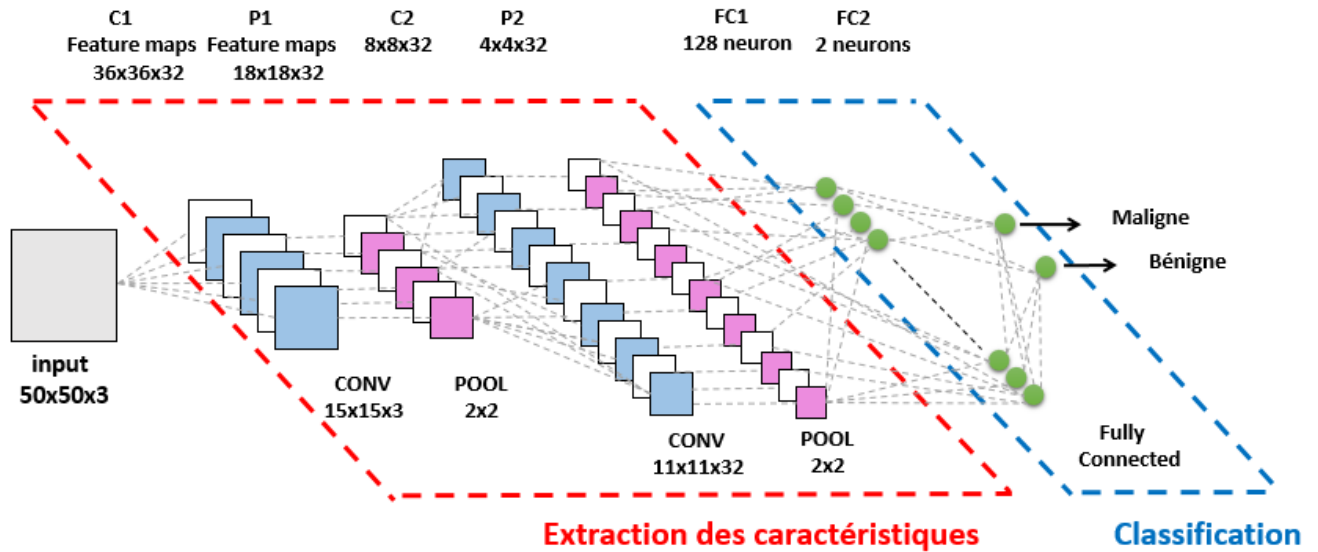


FIGURE 27 – Architecture du modèle 3 .

4.3.4 Modèle 4

Le quatrième modèle est composé de trois couches de convolution, trois couches de MaxPool et de deux couches FC (voir figure 28).

L'image en entrée est de taille 50*50, elle passe par la première couche de convolution. Cette couche est composée de 32 filtres de taille 15*15, la fonction d'activation Relu est utilisée, après cette convolution 32 carte de caractéristiques de taille 36*36 seront créés. Le MaxPool de taille 2x2 est appliqué, à la sortie de cette couche, nous aurons 32 carte de caractéristiques de taille 18*18.

Le même processus est répété avec les couches de convolution et MaxPool suivantes, tel que la 2eme couche de CONV est constituée de 32 filtre 11x11, la 3eme e 32 filtre de taille 3x3, pour le MaxPool on a 2 couches de 2x2. Le résultat est 32 carte de caractéristiques de taille 1x1, le vecteur résultant est de dimension de 32.

Deux couches FC sont utilisées, la 1ere est de 128 neurones, la 2eme est un Softmax de 2 classe maligne et bénigne.

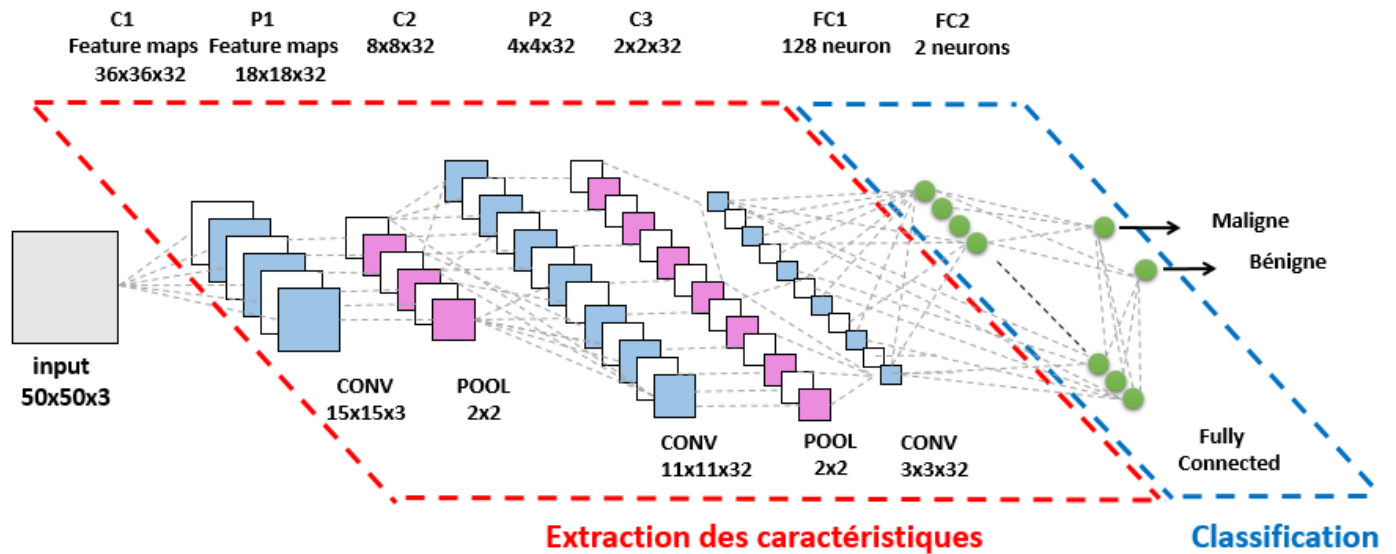


FIGURE 28 – Architecture du modèle 4 .

Etape 2 Après avoir étudié les quatre modèles précédents, on est arrivé à créer une architecture qui comporte un certain nombre de couches de CONV avec une taille des filtres idéale, ainsi que les couches de MaxPool et les FC. Cette architecture est détaillée par la suite :

4.3.5 Modèle 5

Le modèle illustré dans la figure 29, comporte dans son intégralité 6 couches de CONV, 3 couches de MaxPool et 2 couches FC.

L'image en entrée est de taille 50x50x3, on lui applique la première couche de CONV dont la taille des filtres est 3x3x3 avec une profondeur de 32 filtres, ce qui génère une carte de caractéristique de 48x48x32, après on applique une couche de MaxPool de taille 2x2, la carte de caractéristique obtenue après cette étape est de 24x24x32. Ensuite 2 couches de CONV qui se succèdent de taille 3x3x32 et 3x3x64 sont appliquées, suivit d'une couche de MaxPool de 2x2, ceci nous donne une carte de caractéristique de 10x10x64. Le reste de l'étape d'extraction des caractéristiques est composé de 3 couches CONV suivi d'une couche de MaxPool, à la fin, on aura un vecteur de 2x2x128.

En dernier, un réseau de neurones de 2 couches est mis en place, la première couche est composée de 256 neurones avec une activation Relu, la deuxième est une softmax de 2 classes maligne et bénigne.

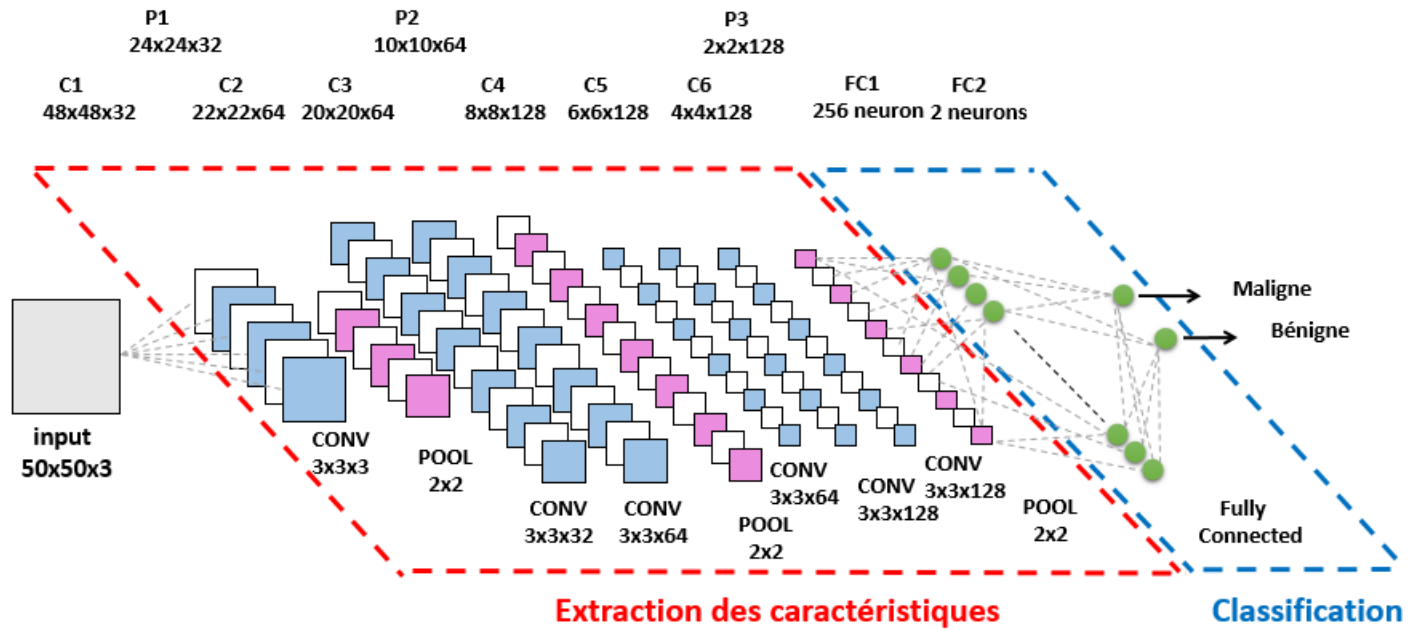


FIGURE 29 – Architecture du modèle 5 .

4.4 Conclusion

À travers ce chapitre, nous avons vu l'importance du choix d'architecture pour avoir une bonne classification des images. On est passé d'une simple architecture qui comporte 2 couches de convolutions, 2 couches de MaxPool et 2 couches de FC à une architecture beaucoup plus profonde, plusieurs couches de convolutions et MaxPool, ceci nous a permis d'avoir de bon résultats de classification. Dans le chapitre suivant on va discuter les résultats obtenus dans ce chapitre.

5 Chapitre 5 Résultats et discussions

5.1 Introduction

Après avoir exécuter tous les modèles d'architecture conçus dans le chapitre précédent, nous allons procéder dans ce chapitre à l'étude des résultats obtenus. Tout d'abord on va présenter les logiciels et bibliothèques utilisés dans l'implémentation ainsi que la base de données et enfin on passe aux discussions des résultats.

5.2 Logiciels et bibliothèque Utilisés dans l'implémentation

5.2.1 Python

Python (voir figure 30) est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Le langage Python est placé sous une licence libre et fonctionne sur la plupart des plates-formes informatiques, des smartphones aux ordinateurs centraux, de Windows à Unix avec notamment GNU/Linux en passant par macOS, ou encore Android, iOS, et peut aussi être traduit en Java ou .NET. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser. Python est un langage qui peut s'utiliser dans de nombreux contextes et s'adapter à tout type d'utilisation grâce à des bibliothèques spécialisées. Il est cependant particulièrement utilisé comme langage de script pour automatiser des tâches simples mais fastidieuses, comme un script qui récupérerait la météo sur Internet ou qui s'intégrerait dans un logiciel de conception assistée par ordinateur afin d'automatiser certains enchaînements d'actions répétitives. On l'utilise également comme langage de développement de prototype lorsqu'on a besoin d'une application fonctionnelle avant de l'optimiser avec un langage de plus bas niveau. Il est particulièrement répandu dans le monde scientifique, et possède de nombreuses bibliothèques optimisées destinées au calcul numérique.



FIGURE 30 – Logo de Python.

5.2.2 TensorFlow

TensorFlow (voir figure 31) est une bibliothèque de logiciels gratuits et à code source ouvert pour le flux de données et la programmation différentiable dans diverses tâches. C'est une bibliothèque mathématique symbolique, qui est également utilisée pour des applications d'apprentissage automatique telles que les réseaux de neurones. Il est utilisé à la fois pour la recherche et la production chez Google.

Son nom est notamment inspiré du fait que les opérations courantes sur des réseaux de neurones sont principalement faites via des tables de données multi-dimensionnelles, appelées Tenseurs (Tensor). Un Tensor à deux dimensions est l'équivalent d'une matrice. Aujourd'hui, les principaux produits de Google sont basés sur TensorFlow : Gmail, Google Photos, Reconnaissance de voix.



FIGURE 31 – Logo de Tensorflow.

5.2.3 Keras

Keras (voir figure 32) est une bibliothèque de réseau de neurones open source écrite en Python. Il est capable de fonctionner sur TensorFlow, Microsoft Cognitive Toolkit, Theano ou PlaidML. Conçu pour permettre une expérimentation rapide avec des réseaux de neurones profonds, il se veut convivial, modulaire et extensible. Il a été développé dans le cadre des efforts de recherche du projet ONEIROS (système d'exploitation de robot intelligent neuroélectronique à extrémité ouverte), et son auteur et responsable principal est François Chollet, ingénieur chez Google.

En 2017, l'équipe TensorFlow de Google a décidé de prendre en charge Keras dans la bibliothèque principale de TensorFlow. Chollet a expliqué que Keras avait été conçu pour être une interface plutôt qu'un cadre autonome d'apprentissage automatique. Il offre un ensemble d'abstractions de niveau supérieur et plus intuitif facilitant le développement de modèles d'apprentissage en profondeur, quel que soit le système de traitement utilisé.



FIGURE 32 – Logo de Keras.

5.3 Configurations Utilisées dans l'implémentation

Le deep learning et en particulier les réseaux de neurones convolutionnels, sont très coûteux en terme de puissance et temps de calcul, pour cela il faut avoir des machines de calcul performantes. Dans ce travail nous avons utilisé 2 PC portables :

- Le 1er est un Dell Inspiron :
 - i7 CPU 2.20 GHz
 - Carte graphique NVIDIA GeForce GT525M
 - RAM de 8GB

Système d'exploitation Windows 10

- Le 2eme est un HP :
 - i3 CPU 2.00 GHz
 - Carte graphique de base HD Family
 - RAM de 4GB
 - Système d'exploitation Windows 10

5.4 La base de données

Le carcinome canalaire invasif (IDC) est le sous-type le plus courant de tous les cancers du sein. Pour attribuer un degré d'agressivité à un échantillon complet, les pathologistes se concentrent généralement sur les régions contenant l'IDC. Par conséquent, l'une des étapes courantes du prétraitement de la gradation automatique de l'agressivité consiste à délimiter les régions exactes d'IDC dans une image.

La base de données original consiste en 162 images de diapositives de monture entières de spécimens de cancer du sein numérisés à 40x. À partir de là, 277 524 patches de taille 50 x 50 ont été extraits (198 738 IDC négatifs et 78 786 IDC positifs). Le nom de fichier de chaque patch a le format suivant : u-xX-yY-classC.png -> exemple 10253-idx5-x1351-y1101-class0.png.

Où u est l'ID du patient (10253-idx5), X est la coordonnée x de l'endroit où ce patch a été pris, Y est l'ordonnée de l'endroit où ce patch a été pris, et C indique la classe où 0 n'est pas IDC et 1 est IDC.

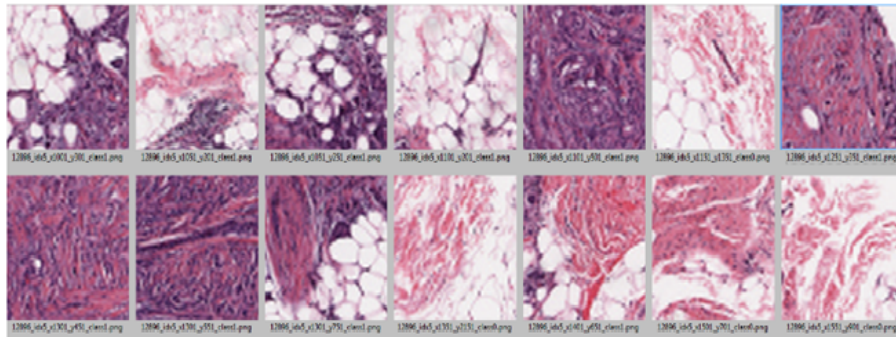


FIGURE 33 – base de donnée.

5.5 Résultats obtenus et discussions

Après avoir vu les architectures des différents modèles qu'on a déjà démontré dans le chapitre précédent, nous allons maintenant discuter les résultats obtenus lors et après la phase d'entraînement, en analysant les graphes de précision et d'erreur de chaque modèle, comparer les architectures vu auparavant et voir la déférence entre eux.

Etape 1

5.5.1 Résultats obtenus pour le 1er Modèle

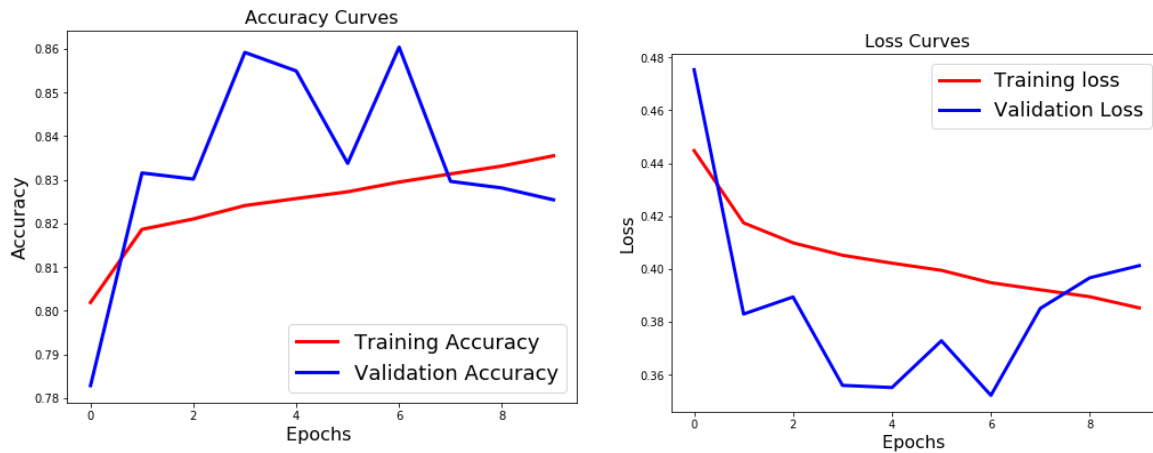


FIGURE 34 – Courbes de Précision et d’Erreur pour le 1er modèle

la figure 34 montre les courbes de Précision et d’Erreur obtenus pendant la phase d’entraînement du 1er modèle, on remarque comme début que « la précision d’apprentissage » (Training Accuracy) augmente de manière rapide lors des premiers epochs à partir de 80 % jusqu’à 82 %, ce qui signifie que le réseau est en phase d’apprentissage des caractéristiques qui constituent l’image pour la première fois, et par la suite, on voit une augmentation légère jusqu’à la fin qui se termine à 83.55 %, car le réseau à ce moment cherche à rajouter des nouvelles caractéristiques qu’il n’a pas apprises lors des epochs qui passent.

Au même temps, l’évaluation de cet apprentissage se fait par le réseau lui-même en utilisant des images étiquetées des deux classes maligne et bénigne qui n’a pas vu lors de la phase d’apprentissage, ces images sont présentées en entrée du réseau, ensuite il va essayer de les classer suivant leur catégorie (maligne ou bénigne), ce qui donne la courbe de « Précision du Test » (Validation Accuracy), ce Test se fait parallèlement avec la phase d’apprentissage, il permet de suivre le réseau et de connaître son efficacité d’apprentissage ; on voit que dans les premiers epochs, le réseau a montré une « Précision du Test » minimum (78.29 %) car il n’était pas suffisamment entraîné, et par la suite, elle a augmentée jusqu’à dépasser la précision d’apprentissage (86.04 %), et à ce moment, on peut dire que le réseau est en bonne état d’apprentissage. A la fin, on remarque une chute au niveau de la courbe de Précision du Test jusqu’à 82.54 %, et donc le réseau a sur-appris (Overfitting).

Pour les deux courbes de l’Erreur d’Apprentissage « Training Loss », et du Test « Validation Loss », il faut savoir que ces deux paramètres sont reliés de manière inverse avec la « précision d’apprentissage et du Test », lorsque la précision d’apprentissage augmente, l’erreur diminue, et vice versa ; la même chose pour la précision et l’erreur du test.

Au niveau de la courbe d’Erreur d’apprentissage, on observe une diminution de 44.47 % jusqu’à 38.52 % pendant la phase d’apprentissage, ce qui signifie que le réseau est entrain d’apprendre ; et au niveau d’Erreur de Test, on constate une chute de 47.53 % jusqu’à 35.21 %, ensuite une augmentation jusqu’à 40.12 %, et donc le réseau a un bon apprentissage, mais une mauvaise validation de résultat (sur-apprentissage). Dans ce cas, il est recommandé d’utiliser soit l’augmentation des données, ou bien le Dropout (voir chapitre CNN).

La figure 35 illustre toutes les valeurs de précisions et d'Erreurs qu'on vient de discuter à travers les courbes :

```

Found 190267 images belonging to 2 classes.
Found 82702 images belonging to 2 classes.
WARNING:tensorflow:From C:\Users\winsido\Anaconda3\lib\site-packages\tensorflow\python\ops\math_ops.py:3066: to_int32 (from
tensorflow.python.ops.math_ops) is deprecated and will be removed in a future version.
Instructions for updating:
Use tf.cast instead.
Epoch 1/10
6000/6000 [=====] - 4883s 814ms/step - loss: 0.4447 - acc: 0.8019 - val_loss: 0.4753 - val_acc: 0.7829
Epoch 2/10
6000/6000 [=====] - 4708s 785ms/step - loss: 0.4174 - acc: 0.8187 - val_loss: 0.3829 - val_acc: 0.8316
Epoch 3/10
6000/6000 [=====] - 4721s 787ms/step - loss: 0.4098 - acc: 0.8210 - val_loss: 0.3894 - val_acc: 0.8302
Epoch 4/10
6000/6000 [=====] - 4555s 759ms/step - loss: 0.4052 - acc: 0.8241 - val_loss: 0.3559 - val_acc: 0.8592
Epoch 5/10
6000/6000 [=====] - 4501s 750ms/step - loss: 0.4022 - acc: 0.8257 - val_loss: 0.3551 - val_acc: 0.8549
Epoch 6/10
6000/6000 [=====] - 4592s 765ms/step - loss: 0.3994 - acc: 0.8273 - val_loss: 0.3728 - val_acc: 0.8337
Epoch 7/10
6000/6000 [=====] - 4643s 774ms/step - loss: 0.3948 - acc: 0.8295 - val_loss: 0.3521 - val_acc: 0.8604
Epoch 8/10
6000/6000 [=====] - 4518s 753ms/step - loss: 0.3921 - acc: 0.8313 - val_loss: 0.3851 - val_acc: 0.8296
Epoch 9/10
6000/6000 [=====] - 4522s 754ms/step - loss: 0.3895 - acc: 0.8332 - val_loss: 0.3966 - val_acc: 0.8281
Epoch 10/10
6000/6000 [=====] - 4526s 754ms/step - loss: 0.3852 - acc: 0.8355 - val_loss: 0.4012 - val_acc: 0.8254

```

FIGURE 35 – les valeurs des Précisions et d'Erreur avec le nombre epochs du 1er modèle.

5.5.2 Résultats obtenus pour le 2ème Modèle

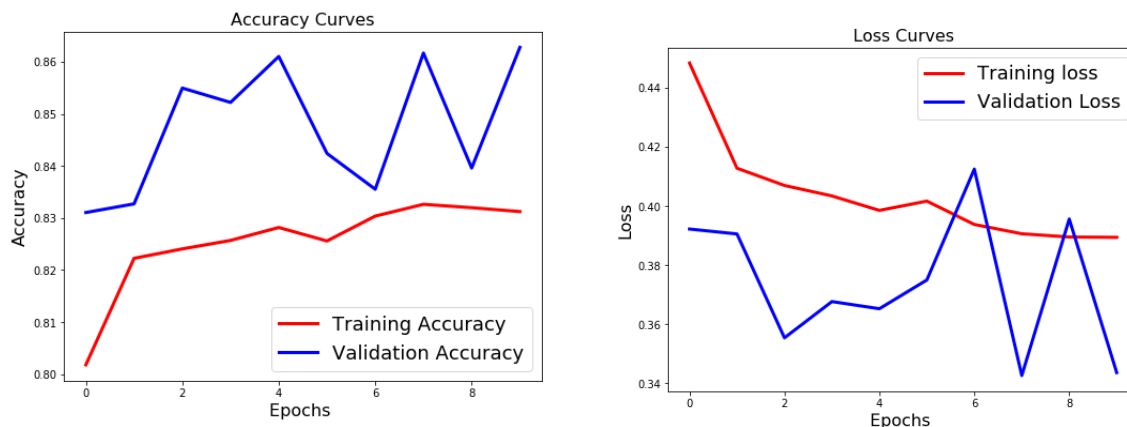


FIGURE 36 – Courbes de Précision et d'Erreur pour le 2ème modèle

Dans ce modèle, la 1ère chose à remarquer (voir figure 36) c'est que dès le 1er epoch nous avons une bonne précision de Test (86.28 %) de façon qu'elle est au-dessus de la courbe de précision d'apprentissage, et dans ce cas le réseau arrive à bien classer les images qui n'a pas entraîné, car on a ajouté une 3ème couche de convolution dans ce modèle ce qui a permis d'extraire plus de caractéristiques et donc bien identifier et classer les images à la sortie.

Par rapport au réseau précédent, la précision d'apprentissage a diminué un peu (83.12 %), mais l'erreur d'apprentissage a arrivé est 38.94 %, et l'erreur du Test a diminué jusqu'à 34.36 % ce qui démontre

l'efficacité de ce réseau par rapport au 1er réseau.

Tous les valeurs de précisions et d'erreur sont montrées dans la figure 37 :

```

Found 190267 images belonging to 2 classes.
Found 82702 images belonging to 2 classes.
Epoch 1/10
6000/6000 [=====] - 5954s 992ms/step - loss: 0.4483 - acc: 0.8018 - val_loss: 0.3922 - val_acc: 0.8310
Epoch 2/10
6000/6000 [=====] - 5427s 905ms/step - loss: 0.4128 - acc: 0.8223 - val_loss: 0.3905 - val_acc: 0.8327
Epoch 3/10
6000/6000 [=====] - 5025s 838ms/step - loss: 0.4069 - acc: 0.8241 - val_loss: 0.3554 - val_acc: 0.8549
Epoch 4/10
6000/6000 [=====] - 4675s 779ms/step - loss: 0.4034 - acc: 0.8257 - val_loss: 0.3677 - val_acc: 0.8522
Epoch 5/10
6000/6000 [=====] - 4436s 739ms/step - loss: 0.3985 - acc: 0.8282 - val_loss: 0.3652 - val_acc: 0.8610
Epoch 6/10
6000/6000 [=====] - 4244s 707ms/step - loss: 0.4016 - acc: 0.8256 - val_loss: 0.3750 - val_acc: 0.8424
Epoch 7/10
6000/6000 [=====] - 4441s 740ms/step - loss: 0.3937 - acc: 0.8304 - val_loss: 0.4125 - val_acc: 0.8355
Epoch 8/10
6000/6000 [=====] - 4365s 727ms/step - loss: 0.3906 - acc: 0.8326 - val_loss: 0.3426 - val_acc: 0.8617
Epoch 9/10
6000/6000 [=====] - 4375s 729ms/step - loss: 0.3895 - acc: 0.8320 - val_loss: 0.3956 - val_acc: 0.8396
Epoch 10/10
6000/6000 [=====] - 4916s 819ms/step - loss: 0.3894 - acc: 0.8312 - val_loss: 0.3436 - val_acc: 0.8628

```

FIGURE 37 – les valeurs des Précisions et d'Erreur avec le nombre d'époque du 2ème modèle.

5.5.3 Résultats obtenus pour le 3ème Modèle

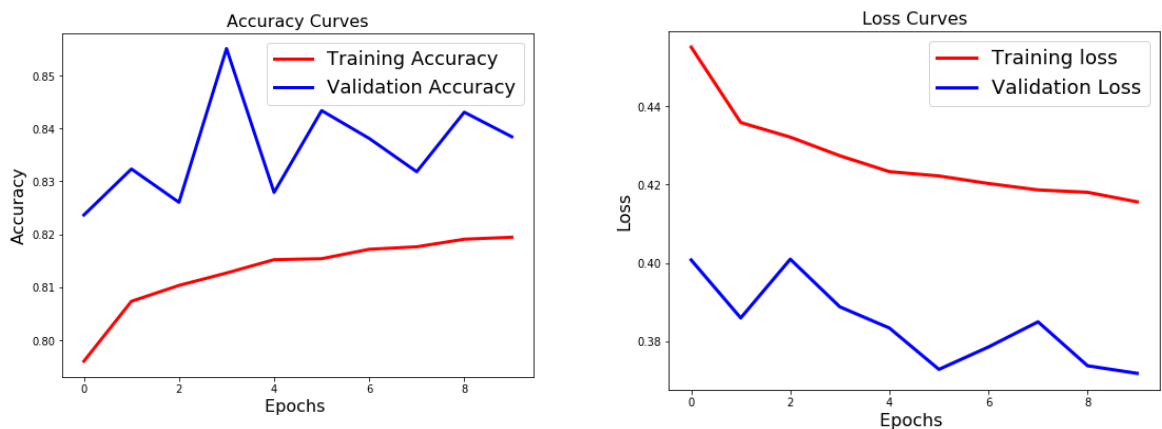


FIGURE 38 – Courbes de Précision et d'Erreur pour le 3ème modèle

Pour le 3ème modèle, on a voulu voir l'importance du choix de la taille et le nombre des filtres utilisée pendant l'entraînement, donc nous avons utilisé des filtres de grande taille; nous constatons depuis les deux graphes (voir figure 38) que la précision d'apprentissage de ce modèle est la minimum parmi les 3 modèles précédents qui est autour de 81.94 %, avec une précision du Test de 83.84 %; on peut dire que ce réseau est bon en classification car la précision de test est grande par rapport au précision d'apprentissage, mais les valeurs des deux précisions sont faibles comparant ce modèle avec les deux autres modèles. A partir de ça, on conclut que plus la taille des filtres est petite, plus il extrait plus de caractéristiques, et plus la précision obtenue à la sortie est bonne.

La figure 39 illustre les valeurs de précision et d'erreur :

```

Found 190267 images belonging to 2 classes.
Found 82702 images belonging to 2 classes.
Epoch 1/10
6000/6000 [=====] - 5064s 844ms/step - loss: 0.4551 - acc: 0.7960 - val_loss: 0.4007 - val_acc: 0.8236
Epoch 2/10
6000/6000 [=====] - 4391s 732ms/step - loss: 0.4358 - acc: 0.8073 - val_loss: 0.3859 - val_acc: 0.8323
Epoch 3/10
6000/6000 [=====] - 4595s 766ms/step - loss: 0.4321 - acc: 0.8103 - val_loss: 0.4009 - val_acc: 0.8260
Epoch 4/10
6000/6000 [=====] - 4574s 762ms/step - loss: 0.4273 - acc: 0.8127 - val_loss: 0.3888 - val_acc: 0.8551
Epoch 5/10
6000/6000 [=====] - 4398s 733ms/step - loss: 0.4233 - acc: 0.8152 - val_loss: 0.3834 - val_acc: 0.8279
Epoch 6/10
6000/6000 [=====] - 4502s 750ms/step - loss: 0.4222 - acc: 0.8154 - val_loss: 0.3728 - val_acc: 0.8434
Epoch 7/10
6000/6000 [=====] - 5273s 879ms/step - loss: 0.4203 - acc: 0.8172 - val_loss: 0.3785 - val_acc: 0.8381
Epoch 8/10
6000/6000 [=====] - 4993s 832ms/step - loss: 0.4186 - acc: 0.8176 - val_loss: 0.3849 - val_acc: 0.8318
Epoch 9/10
6000/6000 [=====] - 4577s 763ms/step - loss: 0.4180 - acc: 0.8191 - val_loss: 0.3737 - val_acc: 0.8431
Epoch 10/10
6000/6000 [=====] - 5330s 888ms/step - loss: 0.4156 - acc: 0.8194 - val_loss: 0.3718 - val_acc: 0.8384

```

FIGURE 39 – les valeurs des Précisions et d’Erreur avec le nombre d’époque du 3ème modèle.

5.5.4 Résultats obtenus pour le 4ème Modèle

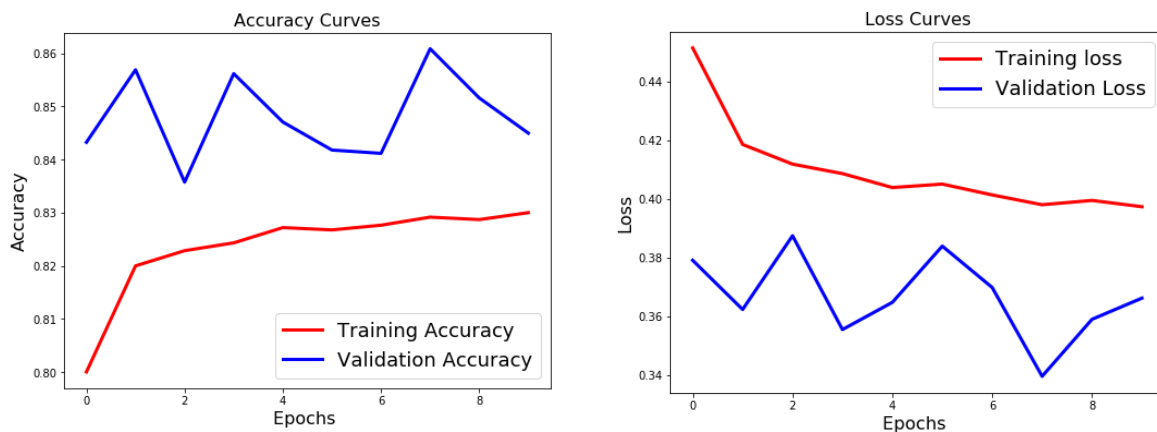


FIGURE 40 – Courbes de Précision et d’Erreur pour le 4ème modèle

Comparant ce modèle avec le 2ème, dans celui-là nous avons gardé le même nombre de filtres utilisées dans les trois couches de CONV, pour montrer l’importance du choix de nombre de filtres dans l’entraînement et son rôle majeur pendant l’apprentissage; la précision d’apprentissage est autour de 83 % et la précision de Test est à 84.49 %, encore que les valeurs d’Erreur d’apprentissage et de Test sont inférieures à celle de 2ème modèle, ce qui prouve que plus on augmente le nombre de filtres dans les dernières couches, plus vous aurez une bonne précision et moins d’erreur.

Les valeurs de précisions et d’erreur sont montrés dans la figure 41 :

```

Found 190267 images belonging to 2 classes.
Found 82702 images belonging to 2 classes.
WARNING:tensorflow:From C:\Users\winsido\Anaconda3\lib\site-packages\tensorflow\python\ops\math_ops.py:3066: to_int32 (from
tensorflow.python.ops.math_ops) is deprecated and will be removed in a future version.
Instructions for updating:
Use tf.cast instead.
Epoch 1/10
6000/6000 [=====] - 6289s 1s/step - loss: 0.4514 - acc: 0.8000 - val_loss: 0.3791 - val_acc: 0.8432
Epoch 2/10
6000/6000 [=====] - 5820s 970ms/step - loss: 0.4185 - acc: 0.8200 - val_loss: 0.3623 - val_acc: 0.8569
Epoch 3/10
6000/6000 [=====] - 5041s 840ms/step - loss: 0.4118 - acc: 0.8228 - val_loss: 0.3874 - val_acc: 0.8357
Epoch 4/10
6000/6000 [=====] - 4761s 793ms/step - loss: 0.4086 - acc: 0.8243 - val_loss: 0.3555 - val_acc: 0.8561
Epoch 5/10
6000/6000 [=====] - 4646s 774ms/step - loss: 0.4038 - acc: 0.8272 - val_loss: 0.3648 - val_acc: 0.8470
Epoch 6/10
6000/6000 [=====] - 4444s 741ms/step - loss: 0.4050 - acc: 0.8267 - val_loss: 0.3839 - val_acc: 0.8418
Epoch 7/10
6000/6000 [=====] - 4591s 765ms/step - loss: 0.4013 - acc: 0.8276 - val_loss: 0.3698 - val_acc: 0.8411
Epoch 8/10
6000/6000 [=====] - 4546s 758ms/step - loss: 0.3980 - acc: 0.8291 - val_loss: 0.3396 - val_acc: 0.8608
Epoch 9/10
6000/6000 [=====] - 4415s 736ms/step - loss: 0.3995 - acc: 0.8287 - val_loss: 0.3590 - val_acc: 0.8516
Epoch 10/10
6000/6000 [=====] - 4276s 713ms/step - loss: 0.3973 - acc: 0.8300 - val_loss: 0.3662 - val_acc: 0.8449

```

FIGURE 41 – les valeurs des Précisions et d’Erreur avec le nombre d’époque du 4ème modèle.

Etape 2 D’après les résultats précédemment obtenus, et d’après les connaissances acquises de l’état de l’art, ce modèle finalise notre travail et comporte tous les points relevés lors de l’entraînement des quatre modèles précédents tel que l’architecture et la répartition des couches CONV, MaxPool et FC, aussi que le nombre de filtres utilisé dans chaque couche et leurs tailles, rajouter à cela les autres paramètres comme Dropout et le nombre d’époques.

5.5.5 Résultats obtenus pour le 5ème Modèle

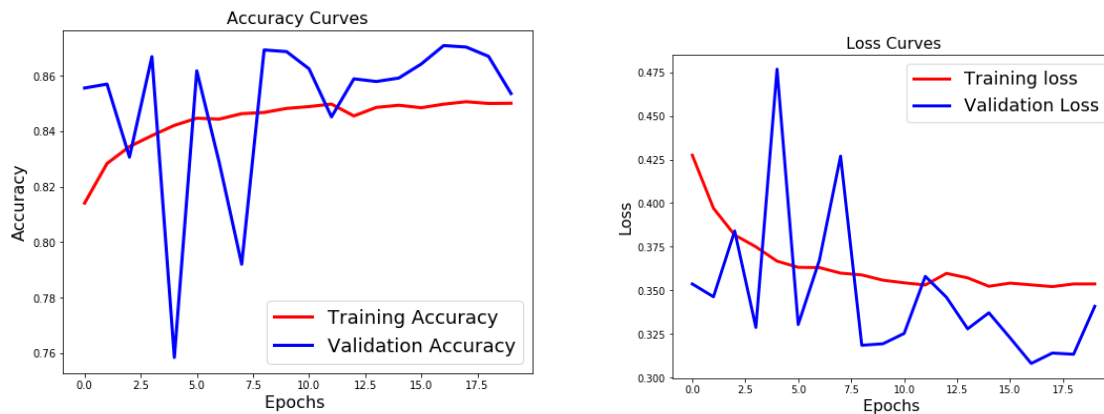


FIGURE 42 – Courbes de Précision et d’Erreur pour le 5ème modèle

Nous constatons dans ce modèle que les précisions d’apprentissage et de Test sont les meilleurs parmi les autres modèles avec des valeurs successives de 85.01 % et 85.36 % comme il est montré dans la figure 42, ce qui indique que ce modèle est bon en apprentissage et en classification au même temps, à bien que les valeurs d’Erreur sont minimales de 35 % et 34 % d’apprentissage et de Test. Nous remarquons qu’il y a

des variations intenses au niveau de la précision et l'erreur d'apprentissage lors des premières epochs, car le réseau était dans la phase d'apprentissage donc Il n'était pas capable de bien classer les images ; on voit qu'à partir du 10ème epoch jusqu'au 17ème, les variations ont diminué le maximum et la précision du Test est la plus grande dans l'entraînement de valeur de 87.09 %, et c'est encore mieux de manière que l'erreur d'apprentissage et de test dans ce point sont les minimums avec des valeurs successives de 35.31 % et 30.80 %, et sa prouve l'efficacité de l'architecture choisi de notre réseau et sa profondeur par rapport aux autres modèles.

A partir du 17ème epoch, on voit une diminution au niveau de la précision du test, de manière qu'elle s'approche de la précision d'apprentissage, mais elle reste toujours au-delà de cette dernière, cela est expliqué, par que le réseau n'a pas reconnu les derniers groupes des images du test malgré il a été bon en apprentissage, ce qui a provoqué une augmentation légère au niveau d'erreur de test, mais la précision et l'erreur d'apprentissage reste toujours les mêmes.

Dans ce cas, il existe quelques suggestions pour éliminer cette diminution, il est recommandé d'augmenter le nombre d'epochs jusqu'à 25 ou 30 pour vérifier qu'après le 20ème epoch que le réseau a amélioré son apprentissage de manière qu'il va bien classer les images du test, raison pour laquel la précision de test remonte, si oui, à ce moment tous va bien, si le cas inverse, ça veut dire que la précision de test continue à diminué, dans ce cas le réseau a sur appris, et donc il existe deux méthodes pour résoudre ce problème :

La 1ère est d'utiliser l'augmentation de données qui eset détaillée dans le chapitre 3, la 2ème est d'essayer plusieurs configurations du Dropout qui est un facteur important pour éliminer ou diminuer le sur-apprentissage.

Les valeurs de précisions et d'erreur sont montrés dans la figure 43 :

```

----- //
Found 190267 images belonging to 2 classes.
Found 82702 images belonging to 2 classes.
Epoch 1/20
6000/6000 [=====] - 4673s 779ms/step - loss: 0.4275 - acc: 0.8141 - val_loss: 0.3536 -
val_acc: 0.8556
Epoch 2/20
6000/6000 [=====] - 4550s 758ms/step - loss: 0.3970 - acc: 0.8284 - val_loss: 0.3462 -
val_acc: 0.8570
Epoch 3/20
6000/6000 [=====] - 4488s 748ms/step - loss: 0.3817 - acc: 0.8345 - val_loss: 0.3840 -
val_acc: 0.8307
Epoch 4/20
6000/6000 [=====] - 4473s 745ms/step - loss: 0.3749 - acc: 0.8384 - val_loss: 0.3287 -
val_acc: 0.8669
Epoch 5/20
6000/6000 [=====] - 4564s 761ms/step - loss: 0.3667 - acc: 0.8421 - val_loss: 0.4768 -
val_acc: 0.7584
Epoch 6/20
6000/6000 [=====] - 4529s 755ms/step - loss: 0.3631 - acc: 0.8447 - val_loss: 0.3303 -
val_acc: 0.8618
Epoch 7/20
6000/6000 [=====] - 4401s 734ms/step - loss: 0.3630 - acc: 0.8444 - val_loss: 0.3672 -
val_acc: 0.8287
Epoch 8/20
6000/6000 [=====] - 4452s 742ms/step - loss: 0.3598 - acc: 0.8463 - val_loss: 0.4269 -
val_acc: 0.7920
Epoch 9/20
6000/6000 [=====] - 4550s 758ms/step - loss: 0.3588 - acc: 0.8468 - val_loss: 0.3184 -
val_acc: 0.8693
Epoch 10/20
6000/6000 [=====] - 4569s 761ms/step - loss: 0.3558 - acc: 0.8482 - val_loss: 0.3193 -
val_acc: 0.8687
Epoch 11/20
6000/6000 [=====] - 4422s 737ms/step - loss: 0.3543 - acc: 0.8489 - val_loss: 0.3252 -
val_acc: 0.8512
Epoch 12/20
6000/6000 [=====] - 4465s 744ms/step - loss: 0.3530 - acc: 0.8498 - val_loss: 0.3580 -
val_acc: 0.8452
Epoch 13/20
6000/6000 [=====] - 4534s 756ms/step - loss: 0.3597 - acc: 0.8455 - val_loss: 0.3460 -
val_acc: 0.8589
Epoch 14/20
6000/6000 [=====] - 4602s 767ms/step - loss: 0.3571 - acc: 0.8486 - val_loss: 0.3279 -
val_acc: 0.8579
Epoch 15/20
6000/6000 [=====] - 4851s 809ms/step - loss: 0.3523 - acc: 0.8494 - val_loss: 0.3371 -
val_acc: 0.8592
Epoch 16/20
6000/6000 [=====] - 5362s 894ms/step - loss: 0.3541 - acc: 0.8485 - val_loss: 0.3227 -
val_acc: 0.8642
Epoch 17/20
6000/6000 [=====] - 4737s 790ms/step - loss: 0.3531 - acc: 0.8498 - val_loss: 0.3080 -
val_acc: 0.8709
Epoch 18/20
6000/6000 [=====] - 4725s 787ms/step - loss: 0.3521 - acc: 0.8506 - val_loss: 0.3140 -
val_acc: 0.8703
Epoch 19/20
6000/6000 [=====] - 4417s 736ms/step - loss: 0.3536 - acc: 0.8500 - val_loss: 0.3133 -
val_acc: 0.8670
Epoch 20/20
6000/6000 [=====] - 4493s 749ms/step - loss: 0.3536 - acc: 0.8501 - val_loss: 0.3408 -
val_acc: 0.8536
----- //

```

FIGURE 43 – les valeurs des Précisions et d’Erreur avec le nombre d’époque du 5ème modèle

5.5.6 Comparaison des résultats

Parmi les 5 modèles qui sont mentionnées dans la table 1, on voit que tous les réseaux montrent une bonne précision d’apprentissage, avec différents ratios d’erreurs, mais le 5ème modèle présente la meilleur précision d’apprentissage avec 85.01 % ce qui signifie que le réseau a été bien entraîné grâce à son architecture profonde et le bon choix des tailles de filtres utilisés, avec un minimum d’erreur d’apprentissage et de test, ceci indique que le réseau a bien classer les images qu’il n’a jamais entraîné; on voit aussi qu’il présente une bonne précision de test de 85.36 % qui se situe après le 2ème modèle de 86.26 %, ce qui indique

Modèle	Modèle 1	Modèle 2	Modèle 3	Modèle 4	Modèle 5
Précision d'Apprentissage finale	83.55%	83.12%	81.94%	83.00%	85.01%
Précision de Test finale	82.54%	86.26%	83.84%	84.49%	85.36%
Précision d'Apprentissage Moyenne	82.482%	82.539%	81.302%	82.364%	84.409%
Précision de Test Moyenne	83.36%	84.738%	83.597%	84.791%	84.90%
Erreur d'apprentissage	38.52%	38.94%	41.56%	39.73%	35.36%
Erreur de Test	40.12%	34.36%	37.18%	36.62%	34.08%

TABLE 1 – Comparaison des précision et erreurs d'apprentissage entre les modèles.

que le 5ème modèle a sur appris en quelque sort dans les derniers epoch de l'entraînement, sinon il sera le meilleur en résultat.

La différence entre les deux est dans le nombre de couches utilisés dans chaque modèle, le 2ème contient moins de couches et moins de nombres de filtres, donc il extrait moins de caractéristiques, la précision est faible et l'erreur est grande; par contre le 5ème modèle est constitué de plusieurs couches et plusieurs filtres, et donc il extrait plusieurs caractéristiques et la précision augmente et l'erreur diminue, et tout ça est montré dans la table 1 avec des valeurs de précision et erreur de chaque modèle.

Modèle	Modèle 1	Modèle 2	Modèle 3	Modèle 4	Modèle 5
Temps d'Apprentissage	12h 52mn	13h 17mn	13h 14mn	13h 33mn	25h 30mn

TABLE 2 – Comparaison des temps d'apprentissage entre les modèles.

Le grand inconvénient des réseaux de neurones convolutifs c'est le temps d'apprentissage, ils prennent beaucoup de temps lors de l'entraînement. La table 2 montre le temps pris par chaque modèle pour l'entraînement des réseaux. nous constatons que plus l'architecture du modèle est grande et profonde, plus le temps d'apprentissage est important, nous pouvons voir cette différence entre le 1er modèle qui a pris presque 13 h et le 5ème modèle qui a pris 25 h30mn.

La différence de temps s'explique dans comment l'architecture du modèle est-elle conçue, le 1er modèle est d'une architecture accès simple par rapport au 5ème modèle, il comporte que 2 couches de convolutions avec de grandes tailles de filtres (15x15 pour la première couche), le 5ème modèle comporte 5 couches de convolution tous de taille 3x3 et ceci rend l'opération de convolution plus lente du fait du nombre de caractéristiques recherchés dans l'image d'entrée.

5.6 Conclusion

Dans ce chapitre nous avons présenté une approche de classification des images histologiques basée sur les réseaux de neurones convolutionnels, pour cela on a utilisé cinq modèles avec différentes architectures, et on a montré les différents résultats obtenus en termes de précision et d'erreur. La comparaison des résultats trouvés a montré que le choix de : l'architecture, nombre et taille des filtres, nombre des couches MaxPool et FC, nombre des epochs, la taille de la base de donnée et la profondeur de réseaux, sont des facteurs importants pour l'obtention de meilleurs résultats.

Conclusion générale

Dans ce travail, nous nous sommes concentrés sur l'utilisation de l'intelligence artificielle dans la classification des images histologiques pour la détection précoce et le diagnostic du cancer du sein, Surtout après la propagation généralisée de ces dernières années. Par conséquent, l'utilisation de méthodes modernes est indispensable pour automatiser la détection et le diagnostic et ainsi accélérer le temps nécessaire pour prendre les mesures nécessaires et accélérer le processus de guérison. Raison pour lequel on a utilisé les réseaux de neurones convolutionnels, puisqu'ils permettent d'extraire un nombre énorme des caractéristiques et des petits détails que nous n'avons pas l'habitude de les voir avec l'œil nu, et les classifier de manière plus rapide et précise comparant ces réseaux par les autres méthodes utilisées dans la classification des images.

Nous avons introduit ces réseaux de neurones convolutionnels en présentant les différents types de couches utilisées dans la classification : la couche de CONV, la couche de Pool et la couche FC ; les hyper paramètres comme poids et biais ; les autres paramètres comme remplissage par zéro, le pas de progression ; les méthodes de régularisation : dropout et data augmentation utilisées pour éviter le problème de sur apprentissage.

Les paramètres du réseau sont difficiles à définir a priori. C'est pour cette raison que nous avons défini différents modèles avec des architectures différentes afin d'obtenir des meilleurs résultats en terme de précision et d'erreur ; le modèle le plus puissant a atteint 85.01 % et 85.36 % de précision d'apprentissage et de test, avec un pourcentage faible d'erreur d'apprentissage et de test de 35.36 % et 34.08 %.

Nous avons rencontré quelques problèmes dans le premier temps, spécifiquement à la recherche d'une base de données compatible avec notre travail, car ils sont très rares et il faut chercher avec grande précision, mais enfin nous avons trouvé cette base qui est devant vous ; pour le 2ème problème, dans la phase d'implémentation, l'utilisation d'un CPU a fait que le temps d'exécution était trop couteux.

À la lumière de ce qui a été fait, ce travail est amené à être prolongé dans plusieurs directions, des idées futures se dessinent :

Utiliser des réseaux de neurones convolutionnels plus profonds cela permettra d'extraire un nombre plus élevé de caractéristiques des images en entrée, élargir la base de données pour assurer une bonne précision d'apprentissage, travailler avec le data augmentation pour éviter le surapprentissage, exécuter les calculs sur un GPU au lieu d'un CPU, cela permet de réduire le temps d'apprentissage vers le minimum possible. Tous ces points abordés font que la précision de classification sera meilleur.

Références

- [1] Sante-medecine, “Sein : anatomie, examens et maladies,” shorturl.at/AHN49, date 07.10.19, visite 21/05/2019.
- [2] Hugo Jaliniere, “Depistage,” https://www.sciencesetavenir.fr/sante/cancer/une-intelligence-artificielle-pour-predire-le-cancer-du-sein_134056, date 30.05.2019, visite 22/05/2019.
- [3] info radiographie.ch, “Roles de l echographie dans le depistage du cancer du sein,” <https://www.info-radiologie.ch/echographie-cancer-sein.php>, le 10.10.2017, visite 22/05/2019.
- [4] info radiographie.ch, “Cancer du sein,” <https://www.info-radiologie.ch/sein.php>, date 10.10.2017, visite 22/05/2019.
- [5] Aurelien Geron, *Hands-On Machine Learning with Scikit-Learn and TensorFlow : Concepts, Tools, and Techniques to Build Intelligent Systems*, OReilly Media, 2017.
- [6] Vikas Gupta, “Convolutional layer,” <https://www.learnopencv.com/image-classification-using-convolutional-neural-networks-in-keras/>, date NOVEMBER 29, 2017, visite 01/06/2019.
- [7] SuperDataScience Team, “Convolutional neural networks (cnn) : Step 4 - full connection,” <https://www.learnopencv.com/image-classification-using-convolutional-neural-networks-in-keras/>, date Saturday Aug 18, 2018, visite 06/06/2019.
- [8] Bharath Raj, “Data augmentation | how to use deep learning when you have limited data part 2,” <https://medium.com/nanonets/how-to-use-deep-learning-when-you-have-limited-data-part-2-data-augmentation-c26971dc8>, date Apr 11, 2018, visite 06/06/2019.
- [9] B. Levin P. Boyle, *World Cancer Report 2008*, World Health Organization, 2009.
- [10] Metin N Gurcan, Laura Boucheron, Ali Can, Anant Madabhushi, Nasir Rajpoot, and Bulent Yener, “Histopathological image analysis : A review,” *IEEE reviews in biomedical engineering*, vol. 2, pp. 147, 2009.
- [11] Carol DerSarkissian, “The breast (human anatomy),” shorturl.at/gvQX0, date : May 18, 2019, visite 21/05/2019.
- [12] British Journal of Cancer, “Breast tumors,” <https://www.nationalbreastcancer.org/breast-tumors/>, visite 22/05/2019.
- [13] Centers for Disease Control, Agency for Healthcare Research Prevention, and Quality, “Biopsy,” <https://www.nationalbreastcancer.org/breast-cancer-biopsy>, visite 27/05/2019.
- [14] “Examen anatomopathologique,” <https://www.e-cancer.fr/Patients-et-proches/Les-cancers/Cancer-du-sein/Diagnostic/Examen-anatomopathologique>, visite 27/05/2019.
- [15] Yongchao XU, Thierry Geraud, and Isabelle BLOCH, “Segmentation dirm de cerveaux de nouveau-nés en quelques secondes a laide dun réseau de neurones convolutif pseudo-3d et de transfert dapprentissage,” in *Actes du 26e Colloque GRETSI, Juan-les-Pins, France, 2017*.
- [16] A Sankari and S Vigneshwari, “Automatic tumor segmentation using convolutional neural networks,” in *2017 Third International Conference on Science Technology Engineering & Management (ICONSTEM)*. IEEE, 2017, pp. 268–272.
- [17] Safaa E Amin and MA Megeed, “Brain tumor diagnosis systems based on artificial neural networks and segmentation using mri,” in *2012 8th International Conference on Informatics and Systems (INFOS)*. IEEE, 2012, pp. MM–119.
- [18] Aqhsa Q Syed and K Narayanan, “Detection of tumor in mri images using artificial neural networks,” *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 3, no. 9, pp. 11749–11754, 2014.

- [19] Adria Romero Lopez, Xavier Giro-i Nieto, Jack Burdick, and Oge Marques, "Skin lesion classification from dermoscopic images using deep learning techniques," in *2017 13th IASTED International Conference on Biomedical Engineering (BioMed)*. IEEE, 2017, pp. 49–54.
- [20] Sameer Khan and Suet-Peng Yong, "A deep learning architecture for classifying medical images of anatomy object," in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2017, pp. 1661–1668.
- [21] Pierre Buysens and Abderrahim Elmoataz, "Réseaux de neurones convolutionnels multi-échelle pour la classification cellulaire," 2016.
- [22] Eva Costa, Nelson Martins, Malik Saad Sultan, Diana Veiga, Manuel Ferreira, Sandra Mattos, and Miguel Coimbra, "Mitral valve leaflets segmentation in echocardiography using convolutional neural networks," in *2019 IEEE 6th Portuguese Meeting on Bioengineering (ENBENG)*. IEEE, 2019, pp. 1–4.
- [23] Daniel S Kermany, Michael Goldbaum, Wenjia Cai, Carolina CS Valentim, Huiying Liang, Sally L Baxter, Alex McKeown, Ge Yang, Xiaokang Wu, Fangbing Yan, et al., "Identifying medical diagnoses and treatable diseases by image-based deep learning," *Cell*, vol. 172, no. 5, pp. 1122–1131, 2018.
- [24] Angel Cruz-Roa, Ajay Basavanahally, Fabio González, Hannah Gilmore, Michael Feldman, Shridar Ganesan, Natalie Shih, John Tomaszewski, and Anant Madabhushi, "Automatic detection of invasive ductal carcinoma in whole slide images with convolutional neural networks," in *Medical Imaging 2014 : Digital Pathology*. International Society for Optics and Photonics, 2014, vol. 9041, p. 904103.
- [25] Fabio A Spanhol, Luiz S Oliveira, Caroline Petitjean, and Laurent Heutte, "A dataset for breast cancer histopathological image classification," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 7, pp. 1455–1462, 2015.
- [26] Nima Habibzadeh Motlagh, Mahboobeh Jannesary, HamidReza Aboulkheyr, Pegah Khosravi, Olivier Elemento, Mehdi Totonchi, and Iman Hajirasouliha, "Breast cancer histopathological image classification : A deep learning approach," *bioRxiv*, p. 242818, 2018.
- [27] Aaron Courville Ian Goodfellow, Yoshua Bengio, *Deep Learning*, MIT Press, 2016.