



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEURE ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE IBN KHALDOUN - TIARET

MEMOIRE

Présenté à :

FACULTÉ DES MATHÉMATIQUES ET DE L'INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

MASTER

Spécialité : **Génie informatique**

Par :

CHERGUI Yakout & BOUSSAHA Chaimaa

Sur le thème

Détection des anomalies par segmentation des images médicales cérébrales

Soutenu publiquement le 21 / 06 / 2022 à Tiaret devant le jury composé de :

Mr CHENINE Abdelkader	MAA Université de Tiaret	Président
Mr HAMEURLAINE Messaoud	MCA Université de Tissemsilt	Encadrant
Mr DAOUD Bachir	MAA Université de Tiaret	Examinateur

2021-2022

Dédicace

Je dédie ce modeste travail

A mes très chers parents pour leur amour inestimable, leur confiance, leur soutien, leurs sacrifices et toutes les valeurs qu'ils ont su m'accorder.

A mes frères et sœurs

A mon fiancé

Qui m'ont apporté leurs encouragements et leurs soutiens.

A Chaimaa, chère amie avant d'être binôme.

A toutes mes amies.

Yakout

Je dédie ce modeste travail

A mes très chers parents pour leur amour inestimable, leur confiance, leur soutien, leurs sacrifices et toutes les valeurs qu'ils ont su m'accorder.

A mes frères et ma sœur

A mon fiancé

Qui m'ont apporté leurs encouragements et leurs soutiens.

A Yakout, chère amie avant d'être binôme.

A toutes mes amies.

Chaimaa

Remerciement

En premier lieu, nous remercions ALLAH, le tout puissant et miséricordieux, qui nous a donné la force le courage et la patience d'accomplir ce modeste travail.

Nos remerciements et nos profondes gratitude vont à notre Encadreur Monsieur HAMEURLAINE Messaoud pour ses précieux conseils et pour tout le soutien et l'orientation. D'avoir bien voulu diriger notre travail, et d'avoir donner le meilleur de son savoir et de son aide, et surtout d'avoir fait preuve de beaucoup de patience, son aide durant toute la période du travail.

Nous tenons aussi à remercier les membres du jury pour leur précieux temps accordé à l'étude de notre mémoire.

Nous remercions nos enseignants pour leurs efforts, nos parents et nos proches pour l'amour et le soutien constant qu'ils nous ont témoigné tout au long de notre parcours. Merci à toutes et tous nos ami(e)s pour leurs encouragements.

ملخص

من خلال هذا المشروع ، نحن مهتمون بتجزئة التصوير بالرنين المغناطيسي للدماغ لاكتشاف الحالات الشاذة (الأمراض / الأورام) الموجودة داخل هذه الصور. الهدف من المشروع هو تنفيذ خوارزمية تسمح لنا باكتشاف الحالات الشاذة في صورة التصوير بالرنين المغناطيسي للدماغ. من أجل تحقيق هدفنا ، قمنا بعمل حوصلة ودراسة كل الخوارزميات المستعملة الخاصة بتجزئة و تصنيف الصور، لقد اخترنا أهمها وأكثرها فاعلية في الأدبيات من أجل تنفيذها. استخدمنا تقنيات التعلم العميق للكشف عن أورام المخ في صور التصوير بالرنين المغناطيسي. طبقنا نموذجًا يسمى VGG16 باستخدام تقنية نقل التعلم في بيئة سحابية . تم الحصول على نتائج تعليمية واختبارات مرضية .

الكلمات المفتاحية: كشف ورم الدماغ ، التصوير بالرنين المغناطيسي، التعلم العميق، نقل التعلم ، جوجل كولا، CNN ، VGG ، الذكاء الاصطناعي.

Résumé

A travers ce projet, nous s'intéressons à la segmentation des IRM cérébrales pour détecter les anomalies (pathologies/maladies) existantes au sein de ces images. Le but du projet est d'implémenter un algorithme qui nous permet de détecter les anomalies dans une image IRM cérébrale. Afin de réaliser notre objectif, nous avons fait une synthèse sur les algorithmes de segmentation/classification des images et nous avons choisi la plus importante et la plus efficace dans la littérature afin de l'implémenter. Nous avons utilisé les techniques de l'apprentissage profond pour la détection des tumeurs cérébrales dans les images IRM. Nous avons implémenté un modèle appelé VGG16 en utilisant la technique de transfert d'apprentissage dans un environnement cloud. Des résultats d'apprentissage et de test satisfaisants ont été obtenus.

Mots clés : Détection des tumeurs cérébrales, IRM : Imagerie par résonance magnétique, Deep Learning , Transfer Learning , Google Colab ,CNN , VGG , Intelligence Artificielle.

Abstract

Through this project, we are interested in the segmentation of cerebral MRIs to detect anomalies (pathologies/diseases) existing within these images. The goal of the project is to implement an algorithm that allows us to detect anomalies in a brain MRI image. In order to achieve our objective, we made a synthesis on the image segmentation/classification algorithms and we chose the most important and the most efficient in the literature in order to implement it. We used deep learning techniques for the detection of brain tumors in MRI images. We implemented a model called VGG16 using the transfer learning technique in a cloud environment. Satisfactory learning and test results have been obtained.

Keywords: Brain Tumor Detection, MRI: Magnetic Resonance Imaging, Deep Learning, Transfer Learning, Google Colab, CNN, VGG, Artificial intelligence.

Sommaire

Introduction générale.....	1
Chapitre I Imagerie Médicale Cérébrale	
Introduction.....	3
I.1. Notion générale de l'image.....	3
I.1.1 Définition d'une image.....	3
I.1.2. Types d'images.....	4
I.1.3. Imagerie médicale.....	5
I.1.4. Chaîne de traitement d'image.....	7
I.2. Anatomie cérébrale.....	8
I.2.1. Le cerveau.....	8
I.2.2. Le cervelet.....	9
I.2.3. Le tronc cérébral.....	9
I.2.4. Les composants de cerveau.....	9
I.2.4.1. La matière blanche.....	10
I.2.4.2. Matière grise.....	10
I.2.4.3. Le liquide céphalo_rachidien (LCR).....	10
I.3. Les tumeurs cérébrales.....	11
I.3.1. Une tumeur.....	12
I.3.2. La différence entre une tumeur bénigne et maligne.....	13
I.3.3. Les causes des tumeurs du cerveau.....	13
I.3.4. La gradation des tumeurs.....	14
I.3.5. Le diagnostic de la tumeur.....	14
I.4. Imagerie par résonance magnétique.....	15
I.4.1. Définition d'IRM.....	15
I.4.2. Principe de l'IRM.....	16
I.4.3. Types d'IRM.....	16
I.4.3.1. IRM.....	16
I.4.3.2. IRM fonctionnelle.....	16
I.4.4. Les caractéristiques d'une image acquise par IRM.....	17
I.4.4.1. Le rapport signal sur bruit.....	17
I.4.4.2. Le contraste.....	18
I.4.4.3. La résolution spatiale.....	18

I.4.5. La qualité d'image acquise par l'IRM.....	19
I.4.6. Risque éventuel	20
I.4.7. Matériels utilisés en IRM	20
I.4.7.1. L'aimant principal.....	20
I.4.7.2. Les bobines de gradient	21
I.4.7.3. Les antennes de champ radio fréquence.....	21
Conclusion	21

Chapitre II Segmentation des images

Introduction.....	23
II.1 Définition de segmentation	23
II.2 Différentes approches de segmentation	23
II.2.1 Approche contour.....	23
II.2.2. Approches régions	24
II.2.2.1 Segmentation par seuillage	25
II.2.2.2 Croissance de région.....	26
II.2.2.3 Méthode de fusion-diffusion	26
II.3. Méthodes basées sur la Classification	27
II.3.1. Méthodes basées sur la Classification Supervisée	27
II.3.1.1. Réseaux de Neurones.....	28
II.3.1.2. L'algorithme K-plus proches voisins [K-Nearest Neighbor] (KNN).....	31
II.3.2. Méthodes basées sur la Classification non Supervisée	32
II.3.2.1 Classification par K-Means (K-Moyens)	33
II.3.2.2. Classification par C_moyenne floue (FCM)	35
II.3.2.3. Classification par K_moyenne possibiliste (PCM).....	36
II.4. Méthodes intelligentes.....	36
II.4.1. La Médecine et l'Intelligence Artificielle	37
II.4.2. L'Imagerie Médicale et l'Intelligence Artificielle	37
II.4.3. Solutions d'Intelligence Artificielle pour le diagnostic des tumeurs.....	38
II.5. L'apprentissage approfondie.....	38
II.5.1. L'apprentissage supervisé.....	39
II.5.2. L'apprentissage non supervisé	39
II.5.3. L'apprentissage semi supervisé	39
II.5.4. L'apprentissage renforcé	39
II.5.5. Types de modèles d'apprentissage en profondeur	39

II.5.6. Réseau de neurones convolutifs (CNN).....	40
II.5.7.Modèle VGG16	43
II.6. Evaluation de la segmentation des images.....	44
II.6.1. Evaluation non-supervisée.....	44
II.6.2. Évaluation supervisée.....	45
Conclusion	46

Chapitre III Contribution et résultats

III.1. Introduction	48
III.2. Environnement de travail.....	48
III.2.1. Python.....	48
III.2.2. Google Colab	51
III.2.2.1. Google Colab	52
III.2.2.2. Google drive.....	53
III.3. Modèle VGG16	56
III.3.1. Implémentation	56
III.3.2. Résultats	63
III.4. Comparaison entre les différents accélérateurs matériels.....	65
Conclusion	68
Conclusion générale	69
Bibliographie	71

LISTE DES FIGURES

Chapitre I : Imagerie Médicale Cérébrale

Figure I.1 : chaîne de segmentation d'image.	7
Figure I.2 : Schéma d'un système de traitement d'images.	7
Figure I.3 : cerveau, cervelet, tronc cérébral.	9
Figure I.4 : Les composants de cerveau.	11
Figure I.5 : présentation d'une tumeur.	12
Figure I.6 : schéma fonctionnel de matériel utilisé.	20

Chapitre II : Segmentation des images

Figure II.1 : Illustration de la détection de contours par une méthode dérivative.	24
Figure II.2 : Exemples d'histogrammes.	26
Figure II.3 : La segmentation par division-Fusion.	27
Figure II.4 : Description d'un neurone.	28
Figure II.5 : Architecture PMC.	30
Figure II.6 : concept de KNN.	32
Figure II.7 : KNN algorithme.	32
Figure II.8 : schéma résumé sur k-means_algorithm.	33
Figure II.9 : les types d'apprentissage.	38
Figure II.10 : Les algorithmes d'apprentissage en profondeur.	40
Figure II.11 : Représentation d'un Réseau de neurones Convolutif.	40
Figure II.12 : exemple d'une couche convolution.	42
Figure II.13 : Exemple de réseau convolutif deux étapes CONV-POL suivies par deux couches FC.	43
Figure II.14 : Le réseau de neurones VGG16.	43
Figure II.15 : L'architecture du modèle VGG16.	44

Chapitre III : Contribution et résultats

Figure III.1 : Les bibliothèques Python.	48
Figure III.2 : La page d'accueil de la plateforme Colab.	52
Figure III.3 : Illustration d'ouvrir un notebook.	52
Figure III.4 : Illustration du nouveau notebook ouvert dans la plateforme colab.	53
Figure III.5 : Illustration : comment utiliser drive.	53
Figure III.6 : Illustration : comment ouvrir un notebook dans drive.	54
Figure III.7 : Illustration : comment importer une bibliothèque dans colab.	54
Figure III.8 : Illustration : choisir l'accélération matérielle.	55
Figure III.9 : L'accélération matériel GPU.	55

Figure III.10 : L'accélération matériel TPU.....	56
Figure III.11 : Les étapes d'implémentation de réseau de Classification.	56
Figure III.12 : Code de préparation de réseau et de l'environnement.....	57
Figure III.13 : Code d'installer les bibliothèques.....	57
Figure III.14 : Importer les bibliothèques utilisées.	58
Figure III.15 : Code d'extraction du fichier compressé.....	59
Figure III.16 : Code d'afficher les images saines.....	59
Figure III.17 : Code d'afficher les images malades.	59
Figure III.18 : Code de Division des données apprentissage/testes	60
Figure III.19 : quelques types d'augmentation de données.	60
Figure III.20 : l'application de l'augmentation des données.	60
Figure III.21 : Construction et compilation du modèle	61
Figure III.22 : Code des paramétrage	61
Figure III.23 : Code d'apprentissage de modèle	62
Figure III.24 : Code d'évaluation du modèle	62
Figure III.25 : Code d'affichage des résultats (précision).	62
Figure III.26 : Code des tracés des courbes (précision et perte).	62
Figure III.27 : Les images saines.	63
Figure III.28 : Les images maladies.....	63
Figure III.29 : Visualisation du modèle.	64
Figure III.30 : Visualisation de l'apprentissage.	64
Figure III.31 : La courbe de perte(loss) et la courbe de précision(accuracy)de réseau VGG16.....	65
Figure III.32 : Le temps d'exécution ave le CPU.....	66
Figure III.33 : Le temps d'exécution ave le GPU.	66
Figure III.34 : Le temps d'exécution ave le TPU.....	67
Figure III.35 : L'histogramme de la Comparaison entre les différents accélérateurs matériels.	67

Liste des tableaux

Chapitre I : Imagerie Médicale Cérébrale

Tableau I.1 : Une étude comparative entre modalités de l'imagerie médicale.....	6
Tableau I.2 : gradation des tumeurs selon l'OMS (Organisation Mondiale de la Santé).....	14

Introduction générale

Introduction générale

La segmentation d'image est une opération de traitement d'image qui a pour but de rassembler des pixels entre eux suivant des critères prédéfinis. La segmentation automatique des tissus de cerveau est devenue une étape fondamentale pour les analyses des images cérébrales. La détection des tumeurs cérébrales signifie l'identification non seulement de la partie affectée du cerveau, mais également de la forme, de la taille, de la limite et de la position de la tumeur. Plusieurs technologies d'imagerie telles que l'image par résonance magnétique (IRM) sont utilisées. Le traitement des images IRM est extrêmement compliqué et constamment utilisé par les chercheurs pour donner aux pathologistes une expérience améliorée afin diagnostiquer les patients.

Dans le domaine médical, il n'existe aucune méthode standard qui puisse être élaborée pour la segmentation des tumeurs cérébrales. Récemment, des techniques d'apprentissage en profondeur ont été adoptées dans les études de segmentation des tumeurs cérébrales après leur succès dans les domaines généraux de l'analyse d'image, tels que la classification d'images, la détection<<d'objets>> et la segmentation sémantique. En particulier, les réseaux de neurones Convolutifs (CNN) ont été adoptés pour détecter les tumeurs cérébrales.

A travers ce projet, nous s'intéressons à la segmentation des IRM cérébrales pour détecter les anomalies (pathologies/maladies) existantes au sein de ces images. Le but du projet est d'implémenter un algorithme qui nous permet de détecter les anomalies dans une image IRM cérébrale. Afin de réaliser notre objectif, nous avons fait une synthèse sur les algorithmes de segmentation/classification des images et nous avons choisi la plus importante et la plus efficace dans la littérature afin de l'implémenter. Nous avons utilisé les techniques de l'apprentissage approfondi pour la détection des tumeurs cérébrales dans les images IRM. Nous avons implémenté un modèle appelé VGG16 en utilisant la technique de transfert d'apprentissage dans un environnement cloud. Des résultats d'apprentissage et de test satisfaisants ont été obtenus.

Autre que l'introduction et la conclusion générale, notre mémoire est composé de trois chapitres :

Chapitre I : Imagerie Médicale Cérébrale.

Chapitre II : Segmentation des images.

Chapitre III : Contribution et Résultats.

Chapitre I

Imagerie Médicale Cérébrale

Introduction

L'imagerie médicale est en constante évolution ces dernières années, elle fournit un nombre croissant de données. Ce volume important de données doit ensuite être analysé par des experts en utilisant diverses méthodes automatiques de traitement et d'analyse d'images qui sont récemment multipliées afin faciliter leur interprétation. Ces méthodes doivent prendre en considération d'une part la quantité des données à analyser et d'autre part, la complexité structurelle des images IRM.

En effet, les méthodes automatiques de traitement et d'analyse d'images se sont récemment multipliées pour assister l'expert dans l'analyse quantitative et qualitative des images.

Le premier chapitre est organisé en trois parties : la première, c'est une généralisation de quelques notions de l'image, dans lequel nous allons voir la définition de l'image et les types, le traitement de la chaîne d'image.

La deuxième, c'est une présentation de l'anatomie cérébrale, nous allons voir le cerveau et ses composants, le cervelet et le tronc cérébral.

La troisième partie est une explication du principe des images à résonance magnétique et ses types et ses caractéristiques, ainsi nous allons voir la qualité d'image acquise par IRM et le matériel qui utilisés en IRM.

I.1. Notion générale de l'image

I.1.1 Définition d'une image

Déf1 : L'image est une description ou reproduction visuel d'un objet donnée par différents moyens ou support. Une image est le nombre de points (pixels). D'une façon générale ne peut pas écrire une définition précisée sur une image.

Déf2 : Une image est plutôt difficile à décrire d'une façon générale. Une image est une représentation du monde. En traitement d'image, la majorité du temps, on considère qu'il s'agit d'une fonction mathématique de $R \times R$ dans R où le couplet d'entrée est considéré comme une position spatiale, le singleton de sortie comme l'intensité (couleur ou niveaux de gris) du phénomène physique. Il arrive cependant que l'image soit dite "3D" donc la fonction est de $R \times R \times R$ dans R . Les images couleurs peuvent être représentées soit par trois images représentant les

trois couleurs fondamentales, soit par une image de $R \times R$ dans $R \times R \times R$ [Wikipédia].

I.1.2. Types d'images

I.1.2.1. Image numérique

L'image numérique est l'image dont la surface est divisée en éléments de tailles fixes appelés cellules ou pixels, ayant chacun comme caractéristique un niveau de gris ou de couleurs prélevé à l'emplacement correspondant dans l'image réelle, ou calculé à partir d'une description interne de la scène à représenter.

Une image numérisée correspond à une image acquise, puis traitée de façon à être stockée sous forme binaire (succession de bits à 0 ou 1) [1].

Une image numérique est définie par :

- le nombre de pixels qui la composent en largeur et en hauteur.
- la valeur que peut prendre chaque pixel. Elle est représentée par un scalaire dans le cas d'images en niveau de gris et par un vecteur à trois composantes Rouge, Vert et Bleu dans le cas d'images couleur. Ces valeurs sont incluses dans N .

I.1.2.2. Image aux niveaux de gris

Une image peut être définie comme une fonction bidimensionnelle $f(x, y)$, où x et y sont les coordonnées spatiales (plan), f est l'amplitude à tout couple de coordonnées (x, y) appelée l'intensité ou niveau gris de l'image à ce point. Alors que x , y et les valeurs d'amplitude de f sont tous finis, des quantités discrètes. Nous appelons cette image une image numérique. Le domaine du traitement numérique de l'image se réfère au traitement des images numériques au moyen d'un ordinateur numérique [2].

I.1.2.3. Image en couleurs

Généralement les images couleurs sont basées sur trois couleurs primaires: Rouge, Vert et Bleu (RVB) (RGB en anglais), et utilisent typiquement 8 bits pour chaque composante de couleur, donc chaque pixel nécessite $3 \times 8 = 24$ bits pour coder les trois composantes. Chaque composante de couleur peut prendre l'une des valeurs de l'intervalle $[0 \dots 255]$ [3].

La plus simple méthode selon plusieurs méthodes pour voir la possibilité de convertir l'image RVB en niveau de gris est de faire :

$$gris = \frac{bleu+vert+rouge}{3}$$

C'est l'équivalent d'affecter le couleur gris à chacune des trois composants RVB.

I.1.3. Imagerie médicale

On peut définir une image médicale comme la matérialisation sous forme d'images d'information anatomique ou fonctionnelle in vivo de parties (organes tissus, cellule) de corps humaines, ainsi que les données extraites ou dérivées de ces images [4]. L'imagerie médicale est aussi un élément essentiel à la recherche clinique, l'étude des maladies et la mise au point de nouveaux traitements.

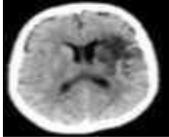
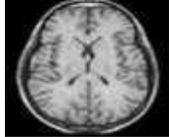
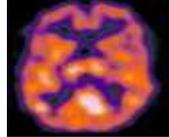
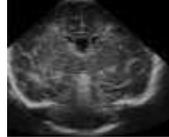
Les techniques d'imagerie ont révolutionné la médecine grâce au progrès de l'informatique en permettant de visualiser indirectement l'anatomie. L'imagerie recouvre à une grande variété de technologies développées grâce à l'exploitation des grandes découvertes de la physique du 20 siècle :

- Les ondes radio et rayon X.
- La radioactivité de certains éléments.
- Les champs magnétiques.

L'objectif est non seulement de diagnostiquer les maladies, suivre leur évolution, découvrir leur fonctionnement, mais aussi de mieux les soigner.

On présente une comparative entre quelques modalités de l'imagerie médicale dans ce tableau :

Tableau 1 : Une étude comparative entre modalités de l'imagerie médicale.

	CT	MRI	PET	SPECT	Ultrasound
Modalités					
Propriétés	Scanne les organes du corps à l'aide de rayons X et produit une série d'images transversales	Produit des «tranches» qui représentent le corps humain par l'application de signaux magnétiques	Exemple de Technique d'imagerie nucléaire où les traceurs sont utilisés dans le diagnostic des maladies	Une technique nucléaire basée sur des gamma caméras pour réaliser des images et des reconstructions en trois dimensions	Basée sur les ondes sonores, possède une haute fréquence temporelle, capable de produire des informations quantitatives et qualitatives de diagnostic
Avantages	<ul style="list-style-type: none"> - Large champ de vision - Détection des différences entre les tissus du corps - Élimine la superposition des images de la structure de recouvrement - Haute résolution spatiale - Grande profondeur de pénétration 	<ul style="list-style-type: none"> - Une résolution plus élevée - Capable de montrer des détails anatomiques - aucun rayonnement ionisant utilisé - Aucun effet à court terme n'est observé 	<ul style="list-style-type: none"> - Utilisé pour distinguer les tumeurs bénignes et malignes - Peut représenter des phénomènes biochimiques et physiologiques - Haute sensibilité - Grande profondeur de pénétration 	<ul style="list-style-type: none"> - Images libres de fond - Confirmer les maladies (Alzheimer, Parkinson) - Haute sensibilité (mais inférieure à PET) - Grande profondeur de pénétration 	<ul style="list-style-type: none"> - À bas prix - Profil sécurisé - Non invasifs (pas de nouilles ou injections) - Largement disponible - Facile à utiliser - Pas de rayonnement
Inconvénients	<ul style="list-style-type: none"> - Sensibilité limitée - Radiation - Haute dose par examen - Coût - Pauvre contraste des tissus mous 	<ul style="list-style-type: none"> - Forte Perturbation du champ magnétique - Ne peut pas être utilisé chez les patients avec des dispositifs métalliques, tels que les stimulateurs cardiaques - Coût 	<ul style="list-style-type: none"> - Résolution spatiale limitée - Radiation - Les coûts élevés - Artefacts de mouvement - Résolution inférieure par rapport à la TDM et l'IRM - L'interprétation est très difficile - Ne permettent qu'à un nombre limité de fois qu'un patient subit une procédure 	<ul style="list-style-type: none"> - Effets Blurring est produit - La compensation d'atténuation est impossible en raison de la diffusion multiple des électrons - Ne réussit pas à prédire les déficits neuropsychologiques - Résolution spatiale limitée - Radiation 	<ul style="list-style-type: none"> - Dépendant de l'opérateur - Imagerie limitée au compartiment vasculaire - Image Difficile de l'os et des poumons - Résolution limitée - Atténuation peut réduire la résolution des images - Reflète très fortement en passant de tissu à gaz ou vice-versa
Contraste	- haut	- haut	-	-	-
Application	- Anatomique - Fonctionnel	- Anatomique - Fonctionnel - Moléculaire	- Fonctionnel - Métabolismes - Moléculaire	-Fonctionnel	- Anatomique - Fonctionnel
Coût	- Coût intermédiaire	- Coût intermédiaire	- Coût élevé	- Coût élevé	- A bas prix
Source & type	- rayons X (ionisant)	- Champs électriques et magnétiques (non ionisant)	- Positron (ionisant)	- Photons (ionisant)	- Les ondes sonores (non ionisant)

I.1.4. Chaîne de traitement d'image

Le traitement d'images est un domaine très vaste qui a connu, et qui connaît encore, un développement important depuis quelques dizaines d'années. Le traitement d'images est l'ensemble des méthodes et techniques opérant sur celles-ci, dans le but de rendre cette opération possible, plus simple, plus efficace et plus agréable, d'améliorer l'aspect visuel de l'image et d'en extraire des informations jugées pertinentes.

La chaîne de traitement d'image est composée de trois étapes essentielles :

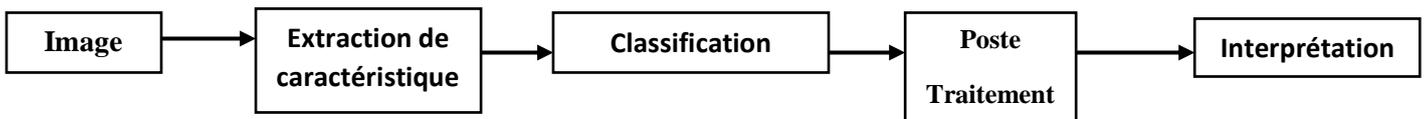


Figure I-1 : chaîne de segmentation d'image.

1. L'étape de prétraitement qui peut s'agir d'éliminer le bruit ou de rehausser le contraste pour améliorer la qualité de l'image acquise.
2. L'étape de segmentation qui sera plus détaillée dans la suite de ce travail. Cette étape peut être suivie par une étape de post traitement (étape facultative) qui consiste à raffiner la segmentation afin d'améliorer les résultats obtenus.
3. L'étape d'interprétation qui consiste à extraire l'information sémantique de l'image en identifiant les différents objets qui la constituent.

On présente le système de traitement d'image par ce schéma :

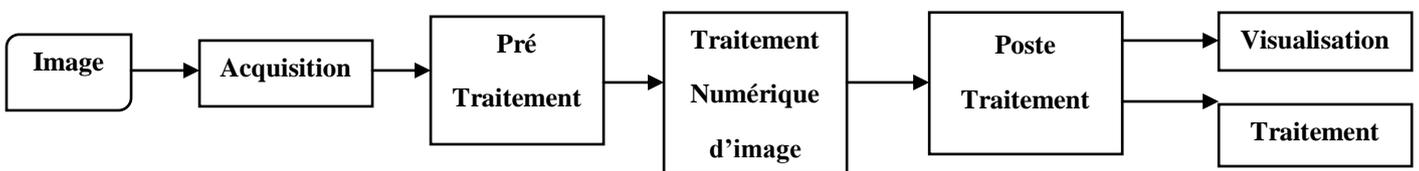


Figure I-2 : Schéma d'un système de traitement d'images.

▪ Acquisition d'une image

L'acquisition d'images constitue un des maillons essentiels de toute chaîne de conception et de production d'images. Pour pouvoir manipuler une image sur un système informatique, il est avant tout nécessaire de lui faire subir une transformation qui la rendra lisible et manipulable par ce

système. Le passage de cet objet externe (l'image d'origine) à sa représentation interne (dans l'unité de traitement) se fait grâce à une procédure de numérisation. Ces systèmes de saisie, dénommés optiques, peuvent être classés en deux catégories principales :

- Les caméras numériques.
- Les scanners.

▪ **Pré-traitement d'images**

Cette phase et le complémentaire de phase acquisition des images, leur objectif a amélioré la qualité de l'image en vue de sa segmentation. Ceci implique que les opérateurs doivent être locaux, c'est-à-dire qu'ils doivent intervenir sur un nombre limité de pixels et essentiellement sur des pixels dans le voisinage proche du pixel courant.

Les techniques de pré-traitements les plus courantes qu'on va présenter sont :

- La modification d'histogramme.
- La réduction du bruit par filtrage.

- **Post traitement** : visualisation, stockage... etc.
- **Interprétation** : vise à étiqueter les régions détectées dans l'étape de segmentation.

I.2. Anatomie cérébrale

L'encéphale est la partie du système nerveux qui loge dans la boîte crânienne et repose à la base du crâne. Son poids moyen est de 1400 à 1800 grammes [4].

I.2.1. Le cerveau

Le cerveau appartient au système nerveux central, situé dans la boîte crânienne, c'est un organe essentiel qui assure la régulation de toutes les fonctions vitales. Il est composé de deux hémisphères, droit et gauche, connectés entre eux par le corps calleux composé de substance blanche (axones gainés de myéline). Divisés en lobes, les hémisphères sont les parties les plus développées chez les mammifères [5].

I.2.2. Le cervelet

Le cervelet situé en parallèle sur deux grandes voies nerveuses (celles qui amènent les messages sensoriels vers les zones du cortex qui en font l'analyse et celles qui partent du cortex et descendent vers les muscles pour les faire contracter). Le cervelet est en partie impliqué dans la coordination des mouvements du corps [6].

I.2.3. Le tronc cérébral

Il représente un carrefour majeur du système nerveux central entre le cerveau et le cervelet. Il est responsable en particulier de la régulation de la respiration et du rythme cardiaque. C'est également un centre de passage des voies motrices et sensibles, ainsi qu'un centre de contrôle de la douleur [7].

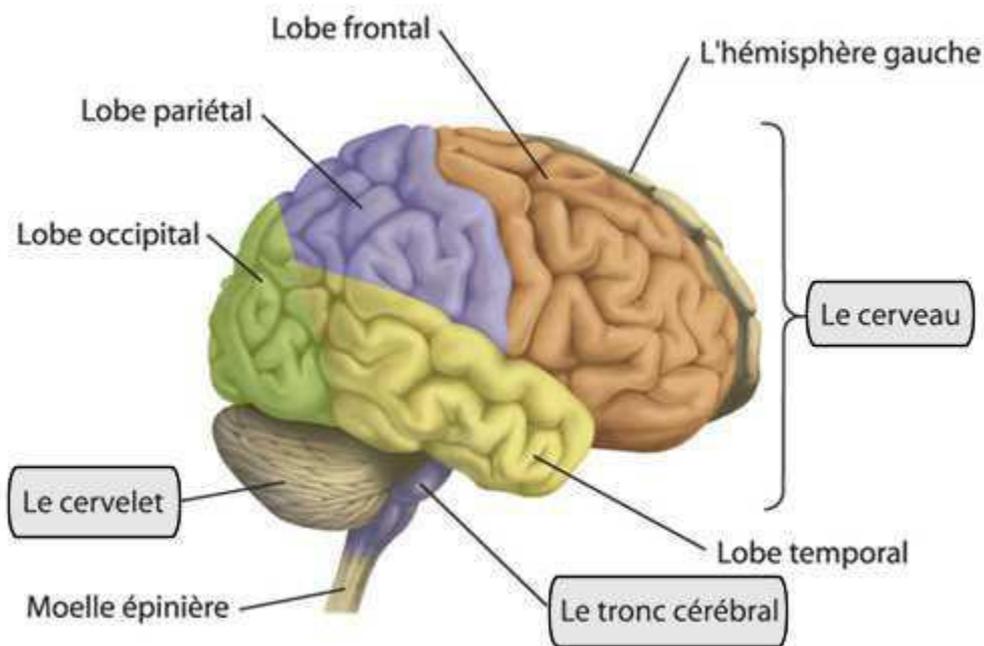


Figure I-3 : cerveau, cervelet, tronc cérébral.

I.2.4. Les composants de cerveau

Le cerveau est composé de trois matières : la matière blanche, la matière grise et le liquide céphalo-rachidien.

I.2.4.1. La matière blanche

La substance blanche est une catégorie de tissu du système nerveux central, principalement composé des axones myélinisés des neurones. Elle relie différentes aires de la substance grise où se situent les corps cellulaires des neurones. Elle constitue la partie interne du cerveau et la partie superficielle de la moelle épinière.

La substance blanche a en fait un aspect rosé à l'œil nu qui est principalement dû à la présence de capillaires sanguins. Ce n'est qu'après avoir été traité au formaldéhyde, pour la conservation, que ce tissu prend sa couleur blanche caractéristique [Wikipédia].

I.2.4.2. Matière grise

La substance grise est située dans la moelle épinière et dans l'encéphale, soit dans la profondeur du cerveau, où elle forme de petits amas, les noyaux gris, soit en surface (cortex du cervelet et des hémisphères du cerveau). Elle contient surtout les corps cellulaires des cellules nerveuses, mais aussi d'autres cellules, non nerveuses, formant un tissu interstitiel appelé névroglie, qui apporte les éléments énergétiques aux cellules nerveuses et assure leur protection.

La substance grise assure la fonction de centre nerveux : réception des messages, analyse complexe des informations, élaboration des réponses. Elle se caractérise par l'importance et la complexité de ses connexions intercellulaires. Comparée à la substance blanche, elle est ainsi en quelque sorte la partie « noble » du système nerveux.

I.2.4.3. Le liquide céphalo_rachidien (LCR)

Le liquide cérébrospinal (LCS), ou liquide céphalo-rachidien (LCR) en ancienne nomenclature, est un liquide biologique transparent dans lequel baignent le cerveau et la moelle spinale. Il est contenu dans les méninges, plus précisément entre la pie-mère (qui recouvre le système nerveux central) et l'arachnoïde (qui tapisse le versant interne de la dure-mère), c'est-à-dire dans l'espace sous-arachnoïdien. C'est également le liquide qui circule dans les quatre ventricules cérébraux, à l'intérieur du cerveau, dans le canal central de la moelle spinale puis au-delà du tube neural jusqu'au cul-de-sac dural.

Il est sécrété par les plexus choroïdes au niveau des ventricules latéraux, ventricule 3 et ventricule 4 dans les hémisphères cérébraux dérivant du tube neural encéphalique primitif.

Le liquide cébrospinal absorbe et amortit les mouvements ou les chocs qui risqueraient d'endommager le cerveau. Il est également le liquide dans lequel sont évacuées les molécules et les « déchets » provenant du cerveau et joue également un rôle de protection immunologique. Sa composition est ainsi supposée refléter l'état physiopathologique du cerveau : inflammation, infection, présence de molécules pharmacologiques, etc. [Wikipédia].

C'est un liquide corporel transparent dans lequel baignent le cerveau et le cervelet, C'est également le liquide qui circule dans les quatre ventricules cérébraux, à l'intérieur du cerveau, et dans le canal central de la moelle épinière. Il est constitué de 99 % d'eau. D'un volume moyen de 150 ml, il est absorbé par le système veineux cérébral et continuellement renouvelé [8]. Ses principaux rôles sont :

- La protection mécanique du système nerveux central contre les chocs.
- La protection contre les infections.
- Le transport des hormones entre les différentes régions éloignées du cerveau.

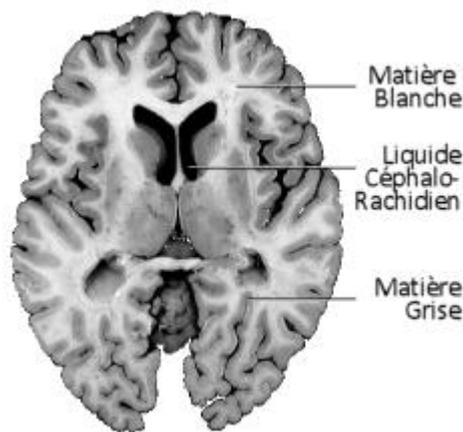


Figure I.4 : Les composants de cerveau.

I.3. Les tumeurs cérébrales

Les tumeurs cérébrales sont un problème de santé important pour tous les groupes d'âge, et les données suggèrent que leur incidence est en augmentation. Chez les nourrissons et les jeunes enfants, les tumeurs du cerveau sont la deuxième forme de cancer la plus courante. Chez les adolescents et les jeunes adultes, les tumeurs cérébrales vont du cinquième au huitième cancer le plus fréquent. Dans la population âgée, l'incidence des tumeurs cérébrales, primitives

et méta-statiques, est en augmentation. Les tumeurs cérébrales primaires ont une prévalence de 14,7 pour 100 000 aux États-Unis, et il y a environ 80 000 à 100 000 nouvelles tumeurs cérébrales Méta-statiques chaque année. Ainsi, pratiquement tous les médecins, quelle que soit leur spécialité, rencontreront des patients atteints de tumeurs cérébrales et la plupart des familles auront un parent ou un ami atteint d'une tumeur cérébrale. [9]

I.3.1. Une tumeur

Une tumeur est un ensemble anormal de cellules qui résulte d'une multiplication accrue et non contrôlée de cellules. On peut dire : Une tumeur ou lésion, néoplasme ou excroissance.

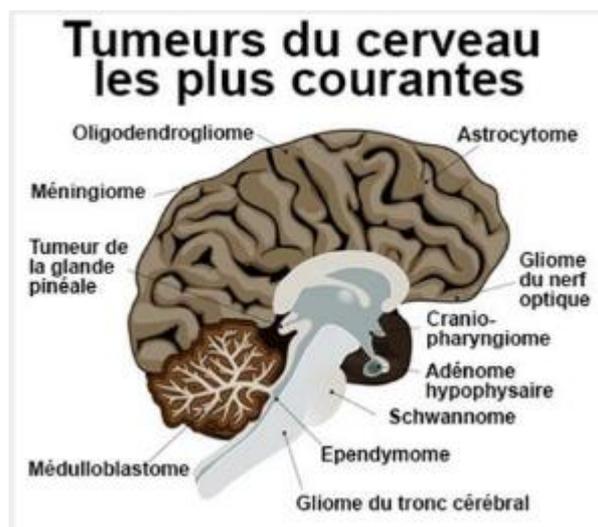


Figure I.5 : présentation d'une tumeur.

D'une manière générale, il existe trois types de tumeurs :

- **Les tumeurs bénignes** : Généralement une tumeur bénigne n'est pas un problème grave (n'est pas un cancer) à moins qu'elle n'appuie sur une structure voisine ou ne provoque d'autres symptômes. Elle se caractérise en particulier par son absence de croissance d'invasion et de métastases.
- **Les tumeurs malignes** : sont les tumeurs cancéreuses. Elles se développent plus rapidement et ont tendance à envahir d'autres zones que celles où elles sont apparues au départ. Elles peuvent développer alors de nouvelles tumeurs appelées métastases.

- **Les tumeurs intermédiaires** : dites « atypiques » ou « évolutives ». Bénéignes au début, elles peuvent se transformer en cancer dans un laps de temps variable.

I.3.2. La différence entre une tumeur bénigne et maligne

Une tumeur bénigne a des bords distincts, lisses et réguliers. Une tumeur maligne a des bords irréguliers et se développe plus rapidement qu'une tumeur bénigne.

Une tumeur maligne peut également se propager à d'autres parties de votre corps. Une tumeur bénigne peut devenir assez grosse, mais elle n'envahira pas les tissus voisins ni ne se propagera à d'autres parties de votre corps.

Certaines tumeurs bénignes peuvent devenir s'aggraver et se transformer en tumeur maligne avec le temps (par exemple, dans le côlon ou la peau). Si vous avez une tumeur bénigne, un fournisseur de soins de santé devrait la surveiller régulièrement.

I.3.3. Les causes des tumeurs du cerveau

Les causes des tumeurs du cerveau sont mal connues. On ne sait pas pourquoi, à un moment donné, les cellules se multiplient de manière incontrôlée jusqu'à former une tumeur. De nombreuses études scientifiques ont été menées ou sont en cours, pour tenter de déterminer les facteurs qui favorisent l'apparition d'une tumeur au niveau du cerveau.

On parle de facteurs de risques. Trois types de facteurs de risques sont étudiés : les risques liés à l'environnement, à l'hérédité et aux virus. Un facteur de risque n'explique pas à lui seul pourquoi une tumeur est apparue. En effet, pour deux personnes confrontées au même risque, l'une peut développer une tumeur et l'autre non. À l'inverse, une personne peut être atteinte d'une tumeur alors qu'elle n'est concernée par aucun facteur de risque. [10]

I.3.4. La gradation des tumeurs

Tableau I.2: gradation des tumeurs selon l'OMS (Organisation Mondiale de la Santé) [11] :

Grade I Tumeurs Bénignes	Tumeurs à croissance lente et normalement bien circonscrites bien qu'elles puissent envahir de grandes régions du cerveau. Selon la localisation, une ablation chirurgicale ou une biopsie peut être recommandée.
Grade II Tumeurs Bénignes	Tumeurs à croissance lente, mais, contrairement aux tumeurs de grade I, leurs limites sont imprécises. Les entités tumorales appartenant à ce groupe sont moins nombreuses que celles du grade I.
Grade III Tumeurs Malignes	Tumeurs anaplasiques Dans les tumeurs de bas grade (I et II), des foyers de cellules anaplasiques (cellules ayant perdues une partie de leurs caractères propres, donc anormales) se développent activement. Leur évolution est plus rapide que celle des tumeurs de bas grade.
Grade IV Tumeurs Malignes	Ces tumeurs peuvent contenir divers types de cellules qui se multiplient rapidement et ayant une forte tendance nécrosante spontanée. Elles ne sont pas bien définies et s'infiltrent dans le cerveau.

I.3.5. Le diagnostic de la tumeur

Le médecin traitant joue un rôle important dans le diagnostic, Le diagnostic d'une tumeur s'inscrit dans une démarche clinique précise et complexe conduisant, par la suite, à une décision thérapeutique adaptée. Pour diagnostiquer des tumeurs on a d'abord besoin de l'acquisition des images par les méthodes d'imagerie médicale (La tomodensitométrie à rayons X, IRM, TEP...) puis le traitement de ces images par les radiologues, ou par des algorithmes de traitement d'image. [11]

Le diagnostic d'une tumeur du cerveau se fait en plusieurs étapes :

- Un examen clinique et neurologique complet.
- Des examens d'imagerie.
- Un examen anatomopathologique, qui consiste à analyser un échantillon de tumeur. [9]

Les examens d'imagerie consistent à réaliser des images précises du cerveau. Ils permettent de détecter une tumeur, de la localiser de façon précise, de mesurer sa taille et d'évaluer les conséquences qu'elle a ou peut avoir sur le cerveau. Pour le diagnostic d'une tumeur du cerveau, une IRM est indispensable. Pour des raisons pratiques, l'IRM est souvent précédée d'un scanner, pour lequel il est plus facile d'obtenir un rendez-vous rapidement. [13]

I.4. Imagerie par résonance magnétique

Aujourd'hui, l'IRM est devenue un outil majeur de l'imagerie médicale moderne, permettant de réaliser des études importantes à l'échelle d'une population ou d'aider les médecins à établir un diagnostic.

L'imagerie par résonance magnétique, qui exploite les propriétés magnétiques des tissus, est certainement l'imagerie qui permet la meilleure distinction entre tissus sains et tissus pathologiques [14].

I.4.1. Définition d'IRM

L'imagerie par résonance magnétique (IRM) est une technique d'imagerie médicale permettant d'obtenir une vue 2D, 3D, d'une partie du corps, notamment du cerveau. Elle est basée sur le principe de la résonance nucléaire (RMN). Grâce aux différentes séquences, on peut observer différents tissus avec des contrastes très élevées [15]. L'imagerie par résonance magnétique (IRM) se révèle particulièrement bien adaptée à l'étude non traumatique d'organes profonds, réputés difficiles d'accès, tels que le cerveau, l'abdomen, Pelvis, etc. [16].

L'imagerie par résonance magnétique. Technique d'examen qui consiste à créer des images précises d'une partie du corps, grâce à des ondes (comme les ondes radio) et un champ magnétique. Les images sont reconstituées par un ordinateur et interprétées par un radiologue. Cette technique est utilisée pour le diagnostic de certaines tumeurs. Pendant l'examen, l'injection d'un produit de contraste peut être nécessaire pour améliorer la qualité de l'image. Cet examen est indolore. [24]

I.4.2. Principe de l'IRM

L'IRM est une technique d'imagerie médicale utilisant un champ magnétique puissant (gros aimant) permettant d'explorer dans les trois plans de l'espace les différents organes du corps humain.

Cette technique consiste à analyser le comportement des protons de l'eau contenus dans l'organisme lorsqu'ils sont soumis à un champ magnétique.

Le signal ainsi recueilli est traité par informatique pour donner des images. Ce sont ces images qui sont reconstruites puis analysées par le médecin radiologue.

En raison de l'utilisation d'un aimant très puissant, de nombreuses précautions s'imposent. Il existe même des contre-indications formelles à la réalisation d'examen IRM. Afin d'éliminer tout risque, il vous est demandé de répondre aux questions posées avec le plus grand soin. En cas de doute, n'hésitez pas à joindre le service. [23]

I.4.3. Types d'IRM

Il existe deux techniques pour l'acquisition de l'IRM :

I.4.3.1. IRM

En médecine, on applique souvent le RMN aux noyaux d'hydrogène, élément présent en abondance dans l'eau et les graisses des tissus biologiques. C'est la structure anatomique que l'on visualise ainsi, et on parle alors d'IRM anatomique.

En observant, sous l'effet d'un champ magnétique intense, la résonance des noyaux d'hydrogène, présents en abondance dans l'eau et les graisses des tissus biologiques, on peut visualiser la structure anatomique d'un organe. Cette méthode peut être utilisée pour le diagnostic de tumeurs cancéreuses ou pour localiser certaines informations [17].

I.4.3.2. IRM fonctionnelle

L'IRM fonctionnelle cérébrale (IRMf) permet d'observer le fonctionnement des tissus gros consommateurs d'oxygène en détectant l'oxygénation de l'hémoglobine dans le sang. Le changement d'état de l'hémoglobine entraîne une modification de ses propriétés magnétiques qui

module le signal IRM. L'IRMf permet l'observation du fonctionnement du cerveau [18].

Avec le développement de techniques ultrarapides d'acquisition et de traitement de données, il est devenu possible de réaliser des images RMN en des temps suffisamment brefs pour suivre certains aspects du métabolisme. En imagerie cérébrale on applique la résonance magnétique à l'hémoglobine dont les propriétés magnétiques diffèrent légèrement selon que cette molécule est liée ou non à l'oxygène. On accède donc à l'activité cérébrale en réalisant des images où est visualisé le contraste entre les régions riches en oxyhémoglobine et les régions du flux sanguin.

I.4.4. Les caractéristiques d'une image acquise par IRM

- Le rapport signal sur bruit
- Le contraste
- La résolution spatiale

I.4.4.1. Le rapport signal sur bruit

Le rapport signal / bruit est fonction de la moyenne de l'intensité du signal par rapport à l'importance de la variation du bruit. C'est le facteur qui conditionne le plus la qualité de l'image par résonance magnétique.

Le bruit correspond aux "parasites". Aléatoire, il provient du patient (agitation thermique des protons à l'origine d'émissions parasites) et dans une moindre mesure de la chaîne de mesure (bruit "électronique"). Il vient perturber le signal émis par les protons excités de la coupe d'intérêt. Il influencera le contraste et la résolution spatiale : le problème posé à l'opérateur est celui de la recherche du meilleur S/B avec la meilleure résolution spatiale, c'est à dire le plus petit voxel.

Le rapport signal/bruit dépend de :

- **Facteurs non modifiables** : intensité du champ, séquence d'ondes RF, caractéristiques tissulaires.
- **Facteurs modifiables** : antenne utilisée.

- **Paramètres de la séquence** : taille du voxels (résolution spatiale en 3D), nombre d'excitations, bande passante de réception. [19]

I.4.4.2. Le contraste

Le contraste dépend du type de séquence utilisé, de ses paramètres et des éventuelles préparations d'aimantation employées. Le contraste sera optimisé afin de répondre aux objectifs de la séquence : étude anatomique, caractérisation tissulaire, recherche d'un œdème, de sang, de graisse ou de liquide, étude de flux vasculaire, rehaussement... [19] Le temps d'acquisition et la qualité de l'image sont interdépendants : il faudra choisir un compromis entre ces deux paramètres lors du paramétrage d'une séquence IRM. Le contraste peut être exprimé en fonction du signal par la relation :

$$c = \frac{S_a - S_b}{S_m}$$

Où : **S_a** et **S_b** sont des radiométries de deux structures consécutives, **S_m** est la mesure du signal moyen sur une zone homogène de l'image.

La faculté de pouvoir différencier deux tissus est ainsi plus justement appréciée par la notion de contraste sur bruit : **C/B**.

I.4.4.3. La résolution spatiale

La qualité d'image est un problème permanent qui se pose à l'opérateur d'IRM. Celui-ci est constamment à la recherche du meilleur compromis possible entre un bon rapport signal sur bruit, une résolution spatiale optimale et une durée d'examen acceptable.

La résolution spatiale correspond à la "finesse" de l'image, c'est-à-dire à la taille du plus petit détail que l'on pourra détecter. Ainsi, plus les voxels de signal enregistré seront petits, plus la résolution spatiale sera élevée. L'augmentation de la résolution spatiale entraîne une baisse du rapport S/B, celui-ci étant proportionnel au volume du voxel, si le contraste et le rapport S/B sont suffisants le contrôle de la résolution spatiale par la taille du voxel est effectif. Telle que le volume du voxel est défini par la dimension de la matrice (256 x 256 ou 512 x 512 etc....), le champ de vue (10 cm, 20 cm, etc....), et l'épaisseur de coupe.

En effet, le gain en résolution spatiale peut être inefficace si l'image est trop bruitée.

Pour un contraste donné, ces trois paramètres sont liés, l'amélioration dans un domaine se faisant au détriment de l'un ou des deux autres. C'est essentiellement en jonglant avec les différents paramètres, qu'il sera possible de maintenir une qualité d'image optimale.

I.4.5. La qualité d'image acquise par l'IRM

A cause de beaucoup des ressources des artefacts en IRM les images peuvent être modifiées [25]. Nous présentons dans ce travail les artefacts suivants :

- **L'artéfact de mouvement** : pendant une séquence d'imagerie, qui est du soit aux mouvements d'un seul organe du patient ; soit aux mouvements globaux du patient.
- **Les inhomogénéités d'intensité** sont des variations de l'intensité observées pour un même tissu [20]. Les inhomogénéités dans le champ magnétique principal et dans le champ généré par les impulsions RF peuvent provoquer une inhomogénéité dans la distribution des intensités des images obtenues en IRM, qui handicape les approches purement photométriques [21].
- **Volume partiel** ; au niveau d'une interface entre deux tissus dans la direction de la sélection de coupe, l'utilisation de coupe fine permet de bien séparer les tissus. En revanche, une coupe plus épaisse contient à la fois les deux tissus : le signal résultant sera alors une moyenne du signal des tissus, d'Où une perte d'information entere de contraste. Il en résulte également perte déclaré solutions partiale, puisque le signal des petites structures va être moyenné avec celui des structures adjacentes, ce qui peut les rendre invisibles ou peu distinctes.
- **Le bruit** : Outre les artefacts, la présence de bruit dans les images acquises, dû aux processus physiques d'acquisition ou aux propriétés des éléments visualisés, complexifient encore la tâche d'analyse des images.

I.4.6. Risque éventuel

Bien que l'équipement utilisé soit bruyant, l'IRM n'est pas un examen douloureux. Le patient doit s'être préalablement dépouillé de tout objet métallique (montre, bracelet, collier, carte de crédit...). Le cas échéant, il doit également signaler le port d'une valve ou d'une pile cardiaque, d'une prothèse ou d'implants. Tous ces éléments sont en effet susceptibles de perturber le déroulement de l'examen. À noter enfin, qu'un produit de contraste peut également être injecté. Généralement bien toléré, il peut toutefois être à l'origine de manifestations allergiques sans gravité (urticaire). Les réactions plus graves sont exceptionnelles, et peuvent généralement être anticipées lors de la consultation préalable à l'examen [22].

I.4.7. Matériels utilisés en IRM

On présente un résumé de matériels utilisés en IRM dans ce schéma :

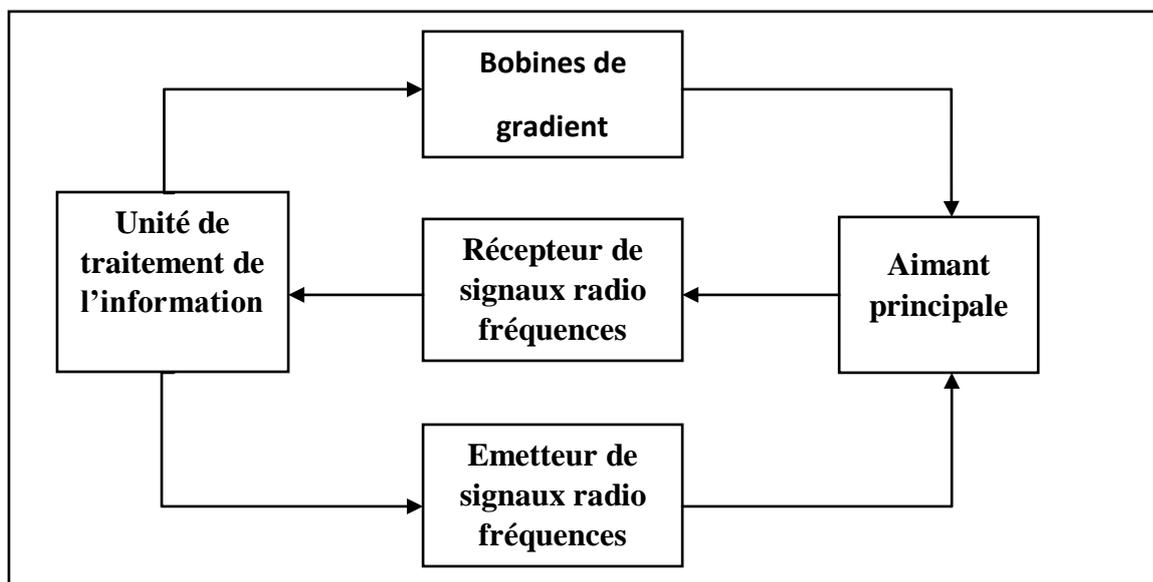


Figure I-6 : schéma fonctionnel de matériel utilisé [25].

I.4.7.1. L'aimant principal

Cet aimant va produire un champ magnétique B_0 [26]. La fréquence de résonance des protons sera proportionnelle à la valeur du champ principal B_0 . Le B_0 peut être produit par :

- **Un aimant résistif :** Un courant électrique (un électro-aimant) parcourt la bobine entourant un noyau de fer doux. Ce type de bobine produira de la chaleur par effet de Joule, un champ magnétique relativement faible.
- **Un aimant supraconducteur :** C'est le type d'aimant installé aujourd'hui. Il sera basé sur le même principe mais sera refroidi à une température proche de 0°K. Il y a ainsi création d'un aimant de champ élevé avec une consommation d'électricité plus faible.
- **Un aimant permanent :** Il peut être créé par un ensemble de corps paramagnétiques. Il faut que le B0 soit très homogène : c'est un facteur de qualité de l'appareil. Cependant il ne sera jamais entièrement uniforme : il le sera à 10^{-5} ou 10^{-6} près. L'intensité du B0 conditionne la qualité de l'image. Le rôle du B0 est de faire entrer les protons en résonance.

I.4.7.2. Les bobines de gradient

On va superposer à B0 un gradient de champ magnétique en fonction de l'espace. Ce champ magnétique est produit par des bobines parcourues par un courant continu. Il y aura des paires de bobines : il faut deux bobines pour créer un gradient de champ et une paire par direction de l'espace. Elles réalisent une variation graduelle de champ B dans l'espace permettant un codage spatial de l'image

I.4.7.3. Les antennes de champ radio fréquence

Les ondes radio fréquences sont constituées par un champ magnétique et par un champ électrique perpendiculaire entre eux. Ces antennes vont émettre un champ B1. Ce champ sinusoïdal, variable en fonction du temps, est de valeur assez faible. Leur rôle va être d'émettre les impulsions radio fréquences et permettre la réception du signal.

Conclusion

Dans ce chapitre, nous avons tout d'abord présentés des notions générales de l'image et l'imagerie médicale, après nous présentons l'anatomie cérébrale, le cerveau et leurs composants, le cervelet et le tronc cérébral et quelques informations sur les tumeurs. Enfin nous avons abordé les notions de l'imagerie par résonance magnétique et leur principe et leurs types, puis l'explication sur le matériel utilisé dans IRM.

Chapitre II

Segmentation des images

Introduction

Les méthodes de segmentation d'images par classification sont issues du domaine de classification des données multidimensionnelles. Dans ce contexte la classification permet de regrouper des pixels possédant les mêmes propriétés et les partitionner dans des classes bien définies. En effet, L'imagerie médicale joue un rôle très important dans l'observation chez l'humain. Elle permet en effet d'observer la structure et le comportement d'organes grâce à un ensemble important de techniques d'acquisition : imagerie par résonance magnétique (IRM), scanographie par rayon X, échographie, etc.

Dans ce chapitre nous présentons quelques notions théoriques et algorithmiques sur les méthodes de segmentation d'images.

II.1 Définition de segmentation

La segmentation est un traitement de bas niveau qui consiste à créer une partition de l'image observé en un certain nombre de régions selon un critère déterminé (couleur, texture, niveau de gris... etc.) qui pourront par la suite être traitées de manières différentes. Dans le domaine médical, la segmentation d'images est extrêmement compliquée. [27]

La segmentation fait référence aux notions de base de différence et de similarités perçues par le système visuel humain. En effet, c'est une étape essentielle en traitement d'images dans la mesure où elle conditionne l'interprétation qui va être faite sur ces images.

II.2 Différentes approches de segmentation

De nombreux algorithmes ont ainsi été proposés durant les dernières décennies [28]. Ils sont généralement fondés sur des principes différents. Nous proposons dans cette section de parcourir, de manière non exhaustive, différentes approches présentes dans la littérature.

II.2.1 Approche contour

Dans une image, les variations (d'intensité, de couleur, de texture...) représentent des changements de propriétés physiques ou géométriques de la scène ou de l'objet observé. Dans un grand nombre de cas, ces variations des propriétés sont des informations importantes pour les opérations de la segmentation. Un contour est une zone de transition séparant deux régions homogènes due à des discontinuités de profondeur et de réflectance de la scène

observée [29]. Elles constituent les frontières de régions correspondant à des bords ou parties d'objets de la scène [30]. De telles méthodes existent depuis longtemps et elles sont les premières méthodes à base de géométrie. Elles ont été développées au milieu des années 80 [31]. Les contours extraits ne sont généralement pas fermés ou bien continus, il est souvent nécessaire d'associer une méthode de suivi ou de fermeture de contours selon le résultat escompté.

Les méthodes dérivatives sont les plus utilisées pour détecter des transitions d'intensité par différenciation numérique, le principe des méthodes dérivatives repose sur la détection d'une transition d'un signal à partir de sa dérivée. On peut citer dans ce cas, les opérateurs du premier ordre les plus courants : Robert [32], de Prewit [33], de Sobel [34].

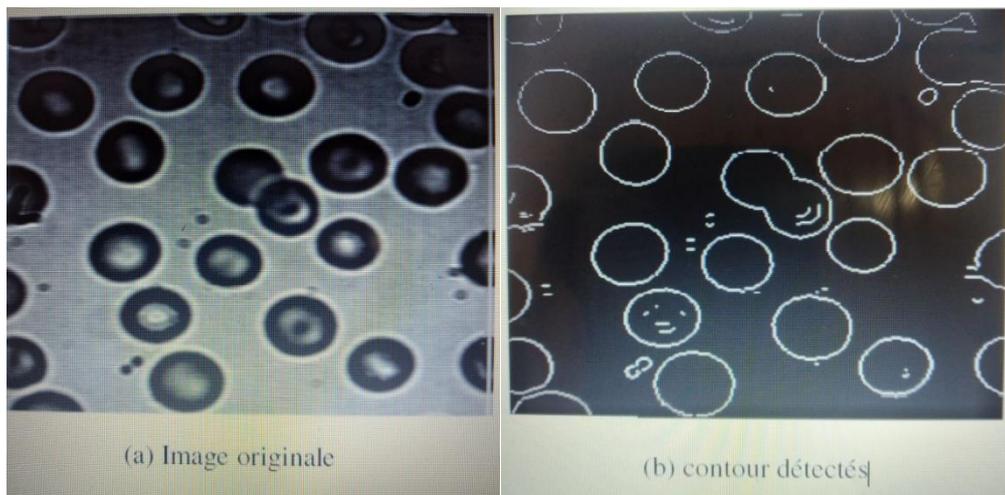


Figure II.1 : Illustration de la détection de contours par une méthode dérivative.

II.2.2. Approches régions

Les méthodes appartenant à cette famille manipulent directement des régions. Soit-elle partie d'une première partition de l'image, qui est ensuite modifiée en divisant ou regroupant des régions, et on parle alors de méthodes de décomposition/fusion (ou split and merge en anglais) ; soit elles partent de quelques régions, qui sont amenées à croître par incorporation de pixels jusqu'à ce que toute l'image soit couverte, et on parle alors de méthodes par croissance de régions. Des méthodes fondées sur la modélisation statistique conjointe de la régularité des régions et des niveaux de gris de chaque région existent également.

Les algorithmes de décomposition/fusion exploitent les caractéristiques propres de chaque région (surface, intensité lumineuse, colorimétrie, texture, etc.). On cherche des couples de régions candidates à une fusion et on les note

en fonction de l'impact que cette fusion aurait sur l'apparence générale de l'image. On fusionne alors les couples de régions les mieux notés, et on réitère jusqu'à ce que les caractéristiques de l'image remplissent une condition prédéfinie : nombre de régions, luminosité, contraste ou texture générale donnée, ou alors jusqu'à ce que les meilleures notes attribuées aux couples de régions n'atteignent plus un certain seuil (dans ce dernier cas, on parle d'un algorithme avec minimisation de fonctionnelle).

Les algorithmes de croissance de région partent d'un premier ensemble de régions, qui peuvent être calculées automatiquement (par exemple, les minima de l'image), ou fournies par un utilisateur de manière interactive. Les régions grandissent ensuite par incorporation des pixels les plus similaires suivant un critère donné, tel que la différence entre le niveau de gris du pixel considéré et le niveau de gris moyen de la région. Les algorithmes de segmentation par ligne de partage des eaux, développés dans le cadre de la morphologie mathématique, appartiennent à cette catégorie.

Les algorithmes fondés sur une modélisation statistique conjointe des régions et des niveaux de gris, notamment ceux s'appuyant sur les champs de Markov cachés, reposent sur la minimisation d'une fonction de vraisemblance (ou énergie). Cette fonction prend simultanément en compte la vraisemblance de l'appartenance du pixel à une région considérant son niveau de gris, et les régions auxquelles appartiennent les pixels voisins. Cette fonction effectue un compromis entre la fidélité à l'image initiale et la régularité des régions segmentées.

II.2.2.1 Segmentation par seuillage

Le seuillage peut aussi être déterminé en étudiant l'histogramme de l'intensité de l'image pour repérer les différents modes de la distribution de l'intensité et calculer ainsi des seuils optimaux afin de détecter les différents objets de la scène. Mais cette méthode ne fonctionne que si les objets ont une intensité homogène et de plus, qui soit différente de celle du fond. [35].

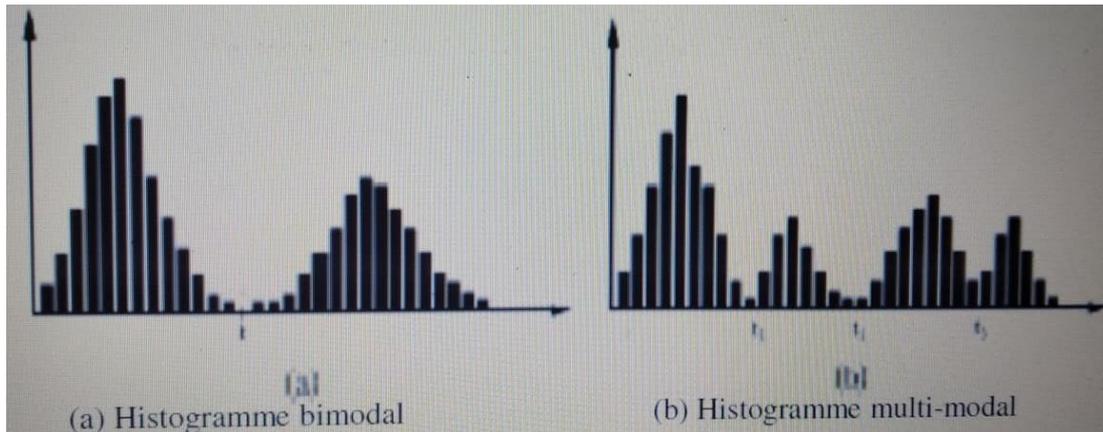


Figure II.2 : Exemples d'histogrammes.

D'une manière générale, le choix de seuils d'histogramme peut se faire de deux manières différentes. Dans la première, les seuils sont déterminés globalement pour tous les points de l'image, tandis que dans la seconde les seuils sont calculés localement à partir de petites fenêtres de focalisation [36].

II.2.2.2 Croissance de région

Ce type de segmentation permet de sélectionner un pixel ou un ensemble de pixels de l'image, appelé germe, autour duquel on fait croître une région. Les régions sont construites en ajoutant successivement à chaque germe les pixels qui lui sont connexes et qui vérifient un critère de similarité. La croissance s'arrête lorsque tous les pixels ont été traités. Les méthodes de croissance de région, également appelées approches ascendantes, est une technique qui consiste à agréger les pixels aux régions selon un double critère d'homogénéité et d'adjacence [37]. Les points voisins du germe sont successivement examinés ; s'ils vérifient le critère d'agrégation, ils sont inclus à la région et deviennent à leur tour un germe ; dans le cas contraire, ils sont rejetés. Le critère est le plus souvent un critère local basé sur les niveaux radiométriques des points. Le processus s'arrête lorsque tous les pixels adjacents aux régions ont été affectés.

II.2.2.3 Méthode de fusion-diffusion

Ce type de méthode consiste à diviser l'image, considérée comme une région initiale, en région de plus en plus petites. Le principe consiste à tester le critère d'homogénéité retenu sur l'image entière. Si le critère est valide, l'image est considérée comme segmentée ; si non, l'image est découpée en zones plus petites et la méthode est réappliquée sur chacune des zones nouvellement

obtenues. Cette méthode est effectuée par un algorithme de « division fusion » pour les images en niveau de gris, Cet algorithme est décomposé en deux étapes [38] :

1. Dans la première étape, chaque bloc (correspondant à un nœud du tétra- arbre) est analysé de sorte à savoir s'il doit être divisé en quatre sous blocs et ce jusqu'à ce que chaque sous-bloc respecte un prédicat d'homogénéité.
2. Dans une deuxième étape, une fusion de blocs adjacents possédant des caractéristiques colorimétriques identiques est réalisée.

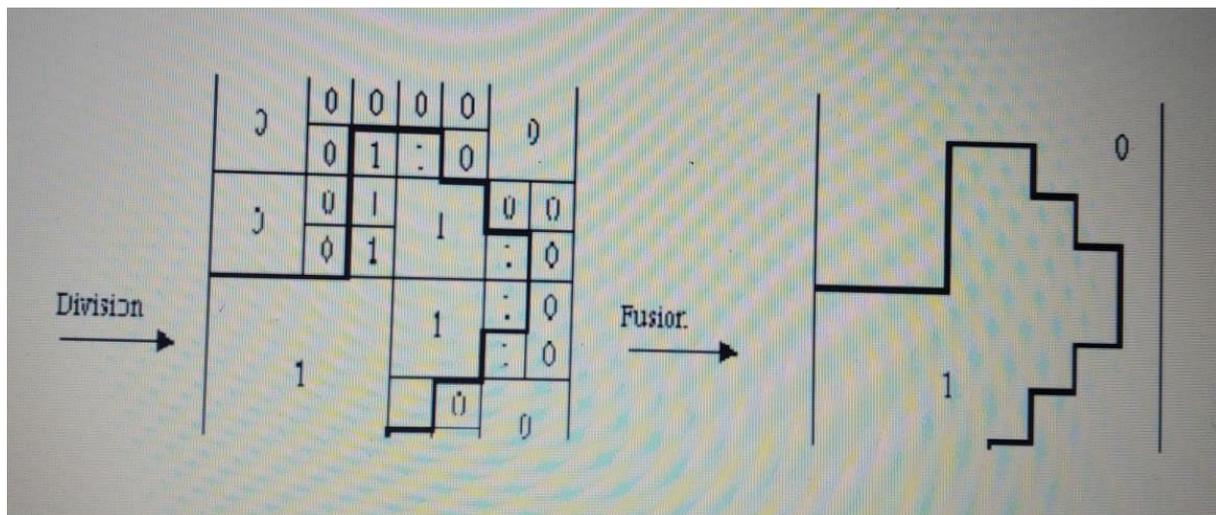


Figure II.3 : La segmentation par division-Fusion

II.3. Méthodes basées sur la Classification

Cette méthode consiste à regrouper et à classer les pixels d'une image en classe en fonction de leurs propriétés.

II.3.1. Méthodes basées sur la Classification Supervisée

L'objectif de la classification supervisée est principalement de définir des règles permettant de classer des objets dans des classes à partir de variables qualitatives ou quantitatives caractérisant ces objets. Les méthodes s'étendent souvent à des variables Y quantitatives (régression).

On dispose au départ d'un échantillon dit d'apprentissage dont le classement est connu. Cet échantillon est utilisé pour l'apprentissage des règles de classement.

Il est nécessaire d'étudier la fiabilité de ces règles pour les comparer et les appliquer, évaluer les cas de sous apprentissage ou de sur apprentissage

(complexité du modèle). On utilise souvent un deuxième échantillon indépendant, dit de validation ou de test.

On dispose de différentes stratégies d'apprentissage :

- Règle majoritaire : à tout objet, on associe la classe k telle que $P(k)$ est maximale.
- Règle du maximum de vraisemblance : à tout objet on associe k telle que $P(d/k)$ maximale.
- Règle de Bayes : à tout objet on associe k telle que $P(k/d)$ maximale. [39].

II.3.1.1. Réseaux de Neurones

Un réseau neuronal simule le système nerveux biologique. Un neurone ou un perceptron est une unité de traitement qui a les caractéristiques suivantes:

- Il possède $P + 1$ entrées que l'on notera e_i avec $i=0, \dots, P$. L'entrée e_0 est particulière : on la nomme le «biais» ou « seuil » et elle vaut toujours 1 ;
- Il possède une sortie notée S ;
- Chacune des entrées est pondérée par un poids note ω_i ;
- Une fonction d'activation, notée φ , détermine la valeur de s en fonction des valeurs présentes en entrée pondérées par leur poids :

$$S = \varphi \left(\sum_{i=0}^{i=p} \omega_i e_i \right)$$

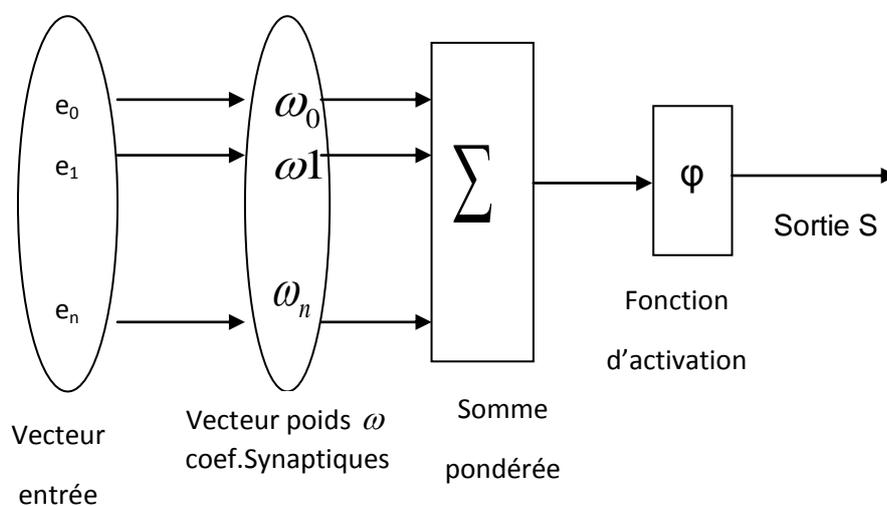


Figure II.4 : Description d'un neurone

Le fonctionnement d'un neurone est très simple : il fonctionne de manière itérative et, à chaque itération, calcule sa sortie en fonction de ses entrées. Dans une tâche de classification, cette sortie indique la classe prédite pour la donnée qui avait été placée en entrée.

La sortie du neurone ne dépend que des poids ω_i (une fois l'entrée fixée bien entendu). Donc l'apprentissage consiste ici à trouver la valeur des poids qui fait en sorte que lorsqu'une donnée est placée sur l'entrée du neurone, la sortie prédite correctement sa classe. Donc, pour un neurone, apprendre signifie "fixer la valeur de ces $P + 1$ paramètres réels".

Pour une tâche de classification, généralement on utilise un neurone dont la sortie est $S \in \{0,1\}$, ou $S \in \{1,-1\}$.

La fonction d'activation peut être simplement la fonction linéaire $\phi(v) = v$. Dans ce cas, la sortie du neurone est la somme pondérée des entrées. On parle alors de neurone «linéaire». La sortie prend alors une valeur dans \mathbb{R} . Cette fonction peut prendre d'autres formes.

Un réseau de neurones formel, couramment appelé réseau de neurones, est un calcul (ou algorithme), généralement réalisé à l'aide d'un ordinateur, dont le résultat reproduit ou prévoit aussi fidèlement que possible, le comportement de n'importe quel processus en fonction des facteurs qui déterminent ce comportement. On entend par « processus » tout système, naturel ou artificiel, et par « facteur » toute grandeur qui est susceptible d'avoir une influence sur le processus. Ainsi, un réseau de neurones est capable de prévoir l'évolution de la température d'un four en fonction de l'intensité du courant électrique dans la résistance chauffante, la vitesse d'un véhicule en fonction du déplacement de la pédale d'accélérateur et de la pente de la route, le cours d'une valeur mobilière en fonction des taux de change du dollar et du yen ainsi que de ses cours passés, la dureté d'un matériau en fonction de sa composition chimique et de sa température d'élaboration, la solvabilité d'une entreprise. [40].

Un perceptron multicouche (PMC) est constitué des éléments suivants:

- Une couche d'entrées (notée C_0): P entrées si les données sont décrites par P attributs (il faut que les attributs soient numériques).
- Une couche de sortie (notée C_q): qui peut contenir plusieurs neurones.
- Une ou plusieurs couches intermédiaires situées entre la couche d'entrée et la couche de sortie. Chaque couche est nommée C_i , chacune est composée d'un certain nombre de perceptrons (le nombre de perceptrons de la couche C_i est noté $|C_i|$, chaque perceptron de la couche C_i possède $|C_{i-1}| + 1$ entrées, chacune de ces entrées correspondant à la sortie d'une

unité de la couche précédente C_{i-1} et une entrée auxiliaire le biais qui vaut 1.

- Chaque connexion entre deux unités est caractérisée par un poids réel.

Dans un PMC, les connexions sont orientées de la couche d'entrée vers la couche de sortie.

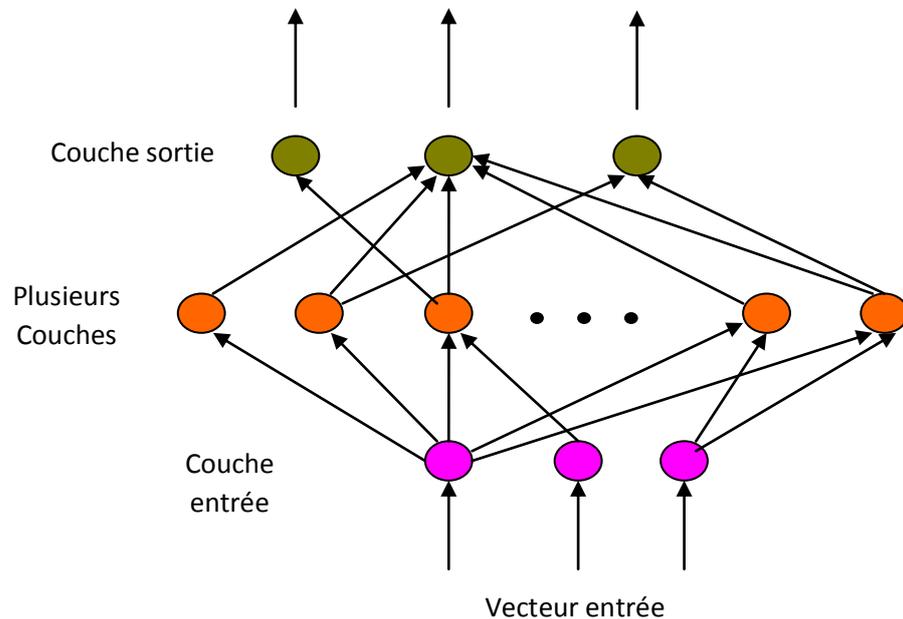


Figure II.5: Architecture PMC

Etapes Principales de construction d'un PMCM

- Construction du réseau **PMC**:
 - Représentation des entrées : le nombre de nœuds en entrée correspond à la dimension des données du problème (attributs ou leurs codages). Les attributs doivent être numériques.
 - Nombre de nœuds par couche : Le nombre de nœuds par couche est au moins égal à deux et au plus égal au nombre de nœuds en entrée.
 - Nombre de nœuds en sortie : fonction du nombre de classes associées à l'application.
- Apprentissage du réseau utilisant les données disponibles pour obtenir un ensemble de poids qui font que la plupart des instances de l'ensemble d'apprentissage sont correctement classées.
- Elagage du réseau : Supprimer les arcs et les nœuds qui n'affectent pas le taux d'erreur du réseau.
- Interprétation des résultats

II.3.1.2. L'algorithme K-plus proches voisins [K-Nearest Neighbor] (KNN)

C'est un type d'algorithme supervisé qui peut être utilisé à la fois pour la classification et les problèmes prédictifs de régression. Cependant, il est principalement utilisé pour les problèmes prédictifs de classification dans l'industrie.

L'algorithme K-plus proches voisins (KNN) utilise la "similarité des caractéristiques" pour prédire les valeurs des nouveaux points de données, ce qui signifie en outre que le nouveau point de données se verra attribuer une valeur en fonction de correspondance avec les points de l'ensemble d'apprentissage.

Nous pouvons comprendre son fonctionnement à l'aide des étapes suivantes ;

Étape 1 – Pour implémenter un algorithme, nous avons besoin d'un ensemble de données. Ainsi, lors de la première étape de KNN, nous devons charger la formation ainsi que les données de test.

Étape 2 – Ensuite, nous devons choisir la valeur de K, c'est-à-dire les points de données les plus proches. K peut être n'importe quel nombre entier.

Étape 3 – Pour chaque point des données de test, procédez comme suit :

- Calculer la distance entre les données de test et chaque ligne de données d'entraînement à l'aide de l'une des méthodes à savoir : distance euclidienne, de Manhattan ou de Hamming. La méthode la plus couramment utilisée pour calculer la distance est euclidienne.
- Maintenant, en fonction de la valeur de distance, triez-les par ordre croissant.
- Ensuite, il choisira les K premières lignes du tableau trié.
- Maintenant, il attribuera une classe au point de test en fonction de la classe la plus fréquente de ces lignes.

Étape 4 – Fin

Exemple :

Voici un exemple pour comprendre le concept de K et le fonctionnement de l'algorithme KNN. Supposons que nous ayons un ensemble de données qui peut être tracé comme suit :

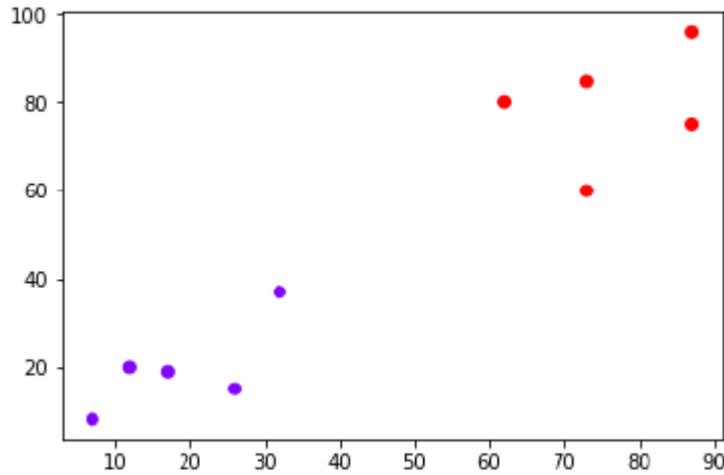


Figure II.6 : concept de KNN.

Maintenant, nous devons classer le nouveau point de données (le point noir (60,60)) en classe bleue ou rouge. Nous supposons que $K = 3$, c'est-à-dire qu'il trouverait les trois points de données les plus proches. Le point noir appartient à la classe des rouges dans ce cas (figure II.7).

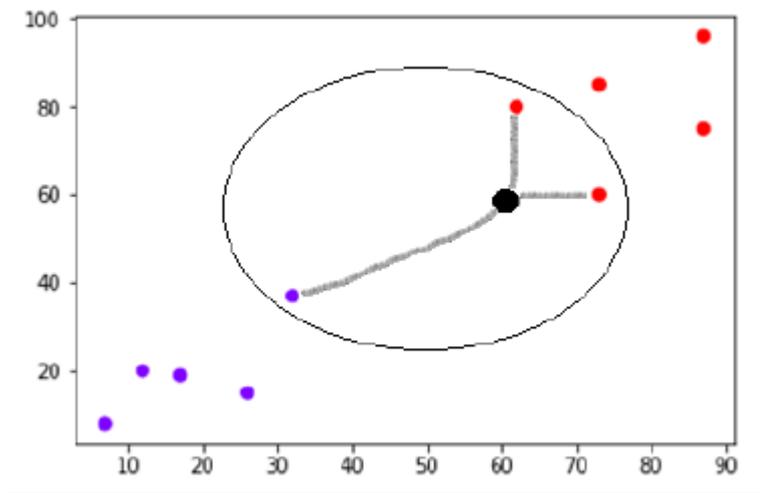


Figure II.7 : KNN algorithme.

II.3.2. Méthodes basées sur la Classification non Supervisée

La classification non supervisée est une méthode visant à identifier la structure en groupes homogènes interprétables de l'ensemble des lignes (ou des colonnes) d'un tableau de données. Elle est réalisée sur une seule dimension du tableau et chaque classe est alors composée d'éléments se ressemblant et différents de ceux des autres classes, une extension de ce procédé a été proposée dans le but de réaliser une classification jointe des lignes et des colonnes d'un tableau. L'objectif de cette analyse est l'identification d'une structure sous-

jacente en blocs homogènes qui existe entre ces deux ensembles ; on parle de classification croisée. Depuis, la classification croisée a été développée sous plusieurs variantes et son intérêt s'est considérablement accru ces dernières années avec l'arrivée de nombreuses applications comme l'analyse de données marketing. [43].

II.3.2.1 Classification par K-Means (K-Moyens)

L'algorithme des k-means est utilisé pour classer les objets à partir de leurs propriétés en k Parties (ou clusters) en supposant que les attributs forment un espace vectoriel. L'objectif est de minimiser la variance intra-cluster V :

$$V = \sum_{i=1}^k \sum_{x_j \in S_i} |x_j - U_i|$$

Où S_i , $i = 1, 2, \dots, k$ sont les k clusters et U_i est le point moyen des points $x_j \in S_i$

L'algorithme de K-means se résume dans les étapes suivantes :

- **Étape 1** : Partitionnement des données en k sous-ensembles (non vides).
- **Étape 2** : Calcul des centres des groupes de la partition courante.
- **Étape 3** : Les données sont affectées au groupe dont leur centre est plus proche.
- **Étape 4** : Retour à l'étape 2.
- **Étape 5** : Arrêt lorsque les groupes sont constants.

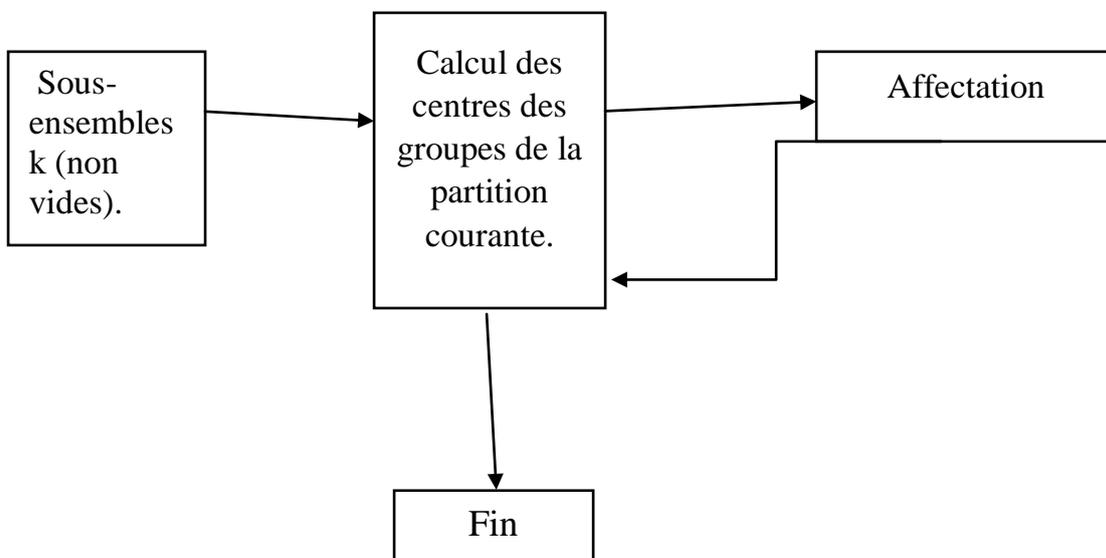


Figure II.8 : schéma résumé sur k-means_algorithm.

Cet algorithme est parmi les algorithmes de segmentation les plus populaires parce qu'il est extrêmement rapide en pratique. En termes de performance, cet algorithme ne garantit pas un optimum global. La qualité de la solution dépend grandement des ensembles initiaux et peut, en pratique, on atteint l'optimum local et non pas l'optimum global. Comme l'algorithme est très rapide, une méthode courante est de le lancer plusieurs fois et de retourner la meilleure partition [39] en plus l'intervention d'un expert est généralement nécessaire pour fixer le nombre K de classes.

-Algorithme des k-means :

Etape1 : Choisir aléatoirement les C centres initiaux $v_1, v_2, v_3 \dots v_c$ à partir des points $\{x_1, x_2, \dots x_N\}$

Etape2: Assigner x_i $i=1 \dots N$ à la classe k ; $k=1 \dots c$ si :

$$\|x_i - v_k\| \leq \|x_i - v_p\| ; p = 1, \dots, c \text{ avec } i \neq p$$

Etape 3 : Mettre à jour les centres des classes par la formule (2).

Etape4 : Répéter de l'étape 2 et 3 jusqu'à atteindre la convergence : $v_k^* = v_k$; $k=1, \dots, c$.

Avantages

- Apprentissage non supervisé : la méthode des k-moyennes et ses variantes ne nécessitent aucune information sur les données. La segmentation peut être utile, pour découvrir une structure cachée qui permettra d'améliorer les résultats de méthodes d'apprentissage supervisé (classification, estimation, prédiction).
- Applicable à tous type de données : en choisissant une bonne notion de distance, la méthode peut s'appliquer à tout type de données (mêmes textuelles).

Inconvénients

- Problème du choix de la distance : les performances de la méthode (la qualité des groupes constitués) sont dépendantes du choix d'une bonne mesure de similarité ce qui est une tâche délicate surtout lorsque les données sont de types différents.
- Le choix des bons paramètres : la méthode est sensible au choix des bons paramètres, en particulier, le choix du nombre k de groupes à constituer. Un mauvais choix de k produit de mauvais résultats. Ce choix peut être fait en combinant différentes méthodes, mais la complexité de l'algorithme augmente.

- L'interprétation des résultats : il est difficile d'interpréter les résultats produits, en d'autres termes, d'attribuer une signification aux groupes constitués. [44].

II.3.2.2. Classification par C_moyenne floue (FCM) :

L'algorithme des C-moyennes floues, effectue une optimisation itérative en évaluant de façon approximative les minimums d'une fonction d'erreur. Il existe toute une famille de fonctions d'erreur associées à cet algorithme qui se distinguent par des valeurs différentes prises par un paramètre réglable, m , appelé indice de flou "fuzzy index" et qui détermine le degré de flou de la partition obtenue. Les FCM sont un cas particulier d'algorithmes basés sur la minimisation d'un critère ou d'une fonction objective. Dans ce cas, les x_j ne sont plus assignés à une unique classe, mais à plusieurs par l'intermédiaire de degrés d'appartenance U_{ji} du vecteur x_j à la classe i . Le but des algorithmes est alors non seulement de calculer les centres de classe mais aussi l'ensemble des degrés d'appartenance des vecteurs aux classes.

Fonctionnement de l'algorithme FCM

L'algorithme fuzzy c-means est un algorithme qui repose sur le même principe que le K_Means mais qui a recours à la logique des ensembles flous (utilisation de probabilités). L'algorithme suivant montre les étapes de FCM :

1. Initialiser les centres
2. Fixer le paramètre m (coefficient flou).
3. Calcul de la partition floue initiale U (la matrice d'appartenance)
4. Répéter :
 - calcul des nouveaux centres
 - calcul de la nouvelle partition floue Tant que le critère d'arrêt n'est pas vérifié.

L'algorithme FCM a été largement utilisé pour la segmentation des images de cerveau, quels que soient la modalité et le type d'acquisition (mono ou multi-5. Spectrale). De nombreux travaux ont notamment été effectués en imagerie par résonance magnétique. En IRM fonctionnelle. La méthode FCM est utilisée pour segmenter les régions activées (simulation et aires motrices) du cerveau. Les voxels étant représentés par leur niveau de gris, les auteurs ont montré non seulement que l'algorithme avait des performances comparables à l'analyse de corrélation standard (avec l'avantage de ne nécessiter aucune connaissance a priori sur le paradigme), mais aussi que le FCM détectait des zones effectivement activées qui restaient silencieuses avec la corrélation. Et

dans notre travail dans le domaine de la télédétection (clustering des images satellitaires). [45].

II.3.2.3. Classification par K_moyenne possibiliste (PCM)

Cette approche combine les avantages des deux approches précédentes : dans la même classe l'appartenance représente la typicalité, cependant en dehors d'une classe l'appartenance représente plutôt le partage. L'algorithme PCMM consiste à trouver une partition de telle manière que le degré d'appartenance d'un objet à la classe gagnante représente un degré de typicalité renforcé par un élargissement de l'extension, alors que les classes restantes le considèrent comme étant un degré de partage en affaiblissant l'extension. L'approche proposée, appelée C-moyennes floues-possibilistes (Fuzzy-Possibilistic C-Means : FPCM). [46]

L'utilisation de l'algorithme de classification possibiliste en imagerie médicale d'une manière générale, et pour la caractérisation des tissus cérébraux en particulier est peu fréquente. Schenone et Masulli ont proposé de combiner une approche par réseaux de neurones avec l'algorithme PCM pour segmenter les tissus cérébraux et des entités pathologiques (méningiomes). Dans l'algorithme PCM et initialisé par l'algorithme FCM dans un processus de fusion de données pour segmenter les tissus sains des images IRM. Barra et Boire ont également appliqué le PCM en IRM sur des vecteurs formes xx_{jj} où chaque voxel est caractérisée par l'ensemble de ses coefficients d'ondelettes. Dans la coopération entre les algorithmes de classification dans une architecture multi-agents est introduite. Cette dernière est basée sur la coopération entre les deux algorithmes FCM et PCM avec l'ajout de deux types de rejets : le rejet d'ambiguïté qui concerne les individus situés entre les frontières des différentes classes et le rejet d'aberrance concernant les individus dont les similarités par rapport à toutes les classes sont très éloignées. L'approche proposée est testée pour segmenter des images IRM saines et d'autres pathologiques. Les résultats obtenus sont très prometteurs [47].

II.4. Méthodes intelligentes

En générale les méthodes intelligentes sont basées sur la vision par ordinateur et l'Intelligence Artificielle. Dans ce travail nous nous intéressons bien évidemment et plus précisément sur la relation de l'IA avec l'Imagerie Médicale.

II.4.1. La Médecine et l'Intelligence Artificielle

L'intelligence artificielle (IA) est un processus d'imitation de l'intelligence humaine qui repose sur la création et l'application d'algorithmes exécutés dans un environnement informatique dynamique. Son but est de permettre à des ordinateurs de penser et d'agir comme des êtres humains. [55]. L'IA augmente la capacité d'analyse et de prise de décision, elle améliore la façon de diagnostiquer, de détecter, de traiter, de pronostiquer une maladie et d'accompagner un patient.

Les systèmes d'IA ont la capacité d'apprendre, conduisant à la découverte de nouveaux phénomènes et à la création de connaissances médicales. Un système informatique doté avec l'intelligence artificielle peut être utilisé pour analyser de grandes quantités de données, Et quand le cas d'un patient est complexe, rare ou que la personne qui pose le diagnostic est simplement inexpérimentée, un système expert peut aider à établir des diagnostics probables basés sur les données du patient.

Maintenant, les systèmes informatiques intelligents peuvent même prédire les étapes chirurgicales, automatiser de différentes tâches pendant la chirurgie, identifier les complications et avertir les chirurgiens des défis à venir [48]. À l'aide de L'intelligence artificielle les chirurgiens peuvent envoyer des flux en direct de point de vue du site opératoire à des experts du monde entier pour des conseils en temps réel. L'IA permet d'avoir l'avis d'un expert éloigné et peut participer à améliorer la prise en charge médicale et non médicale via la télémédecine.

II.4.2. L'Imagerie Médicale et l'Intelligence Artificielle

La technologie et les algorithmes d'intelligence artificielle est le futur de la santé et est déjà en train de la révolutionner dans le domaine médical. Certains voient dans les applications médicales de l'IA la possibilité de remplacer le médecin. L'IA est en train de changer la radiologie diagnostique, et l'anatomopathologie. Dans ce domaine, ses performances ont été comparées à celles de médecins. Il a ainsi été démontré que dans certains cas l'IA était capable d'être plus rapide, plus précise et plus fiable que l'humain. Mais, ses résultats doivent être validés par un humain car les performances des radiologues sont très bonnes et celles des radiologues associés à l'IA sont excellentes, Ce qui renforce l'idée de complémentarité.

II.4.3. Solutions d'Intelligence Artificielle pour le diagnostic des tumeurs

L'utilisation de l'intelligence artificielle dans plusieurs études d'imagerie médicale a démontré que les techniques d'apprentissage automatique telles que les réseaux de neurones convolutifs ont le potentiel de décrire les phénotypes tumoraux avec un pouvoir prédictif supérieur aux mesures cliniques. Les algorithmes d'IA sont automatiquement capables d'identifier les régions et d'extraire leurs caractéristiques qui diffèrent de celles des tissus sains et d'extraire leurs caractéristiques.

II.5. L'apprentissage approfondie

L'apprentissage approfondi ou Le deep Learning peut être considéré comme un sous-ensemble de l'apprentissage automatique. C'est un domaine basé sur l'apprentissage et l'amélioration par lui-même en examinant les algorithmes informatiques. Alors que l'apprentissage automatique utilise des concepts plus simples, le deep Learning fonctionne avec des réseaux de neurones artificiels, conçus pour imiter la façon dont les humains pensent et apprennent. Jusqu'à récemment, les réseaux de neurones étaient limités par la puissance de calcul et étaient donc limités en complexité. Cependant, les progrès de l'analyse du Big Data ont permis des réseaux de neurones plus vastes et sophistiqués, permettant aux ordinateurs d'observer, d'apprendre et de réagir à des situations complexes plus rapidement que les humains.

L'apprentissage est une phase du développement d'un réseau de neurones durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré. On distingue parmi les types d'apprentissage [49] :

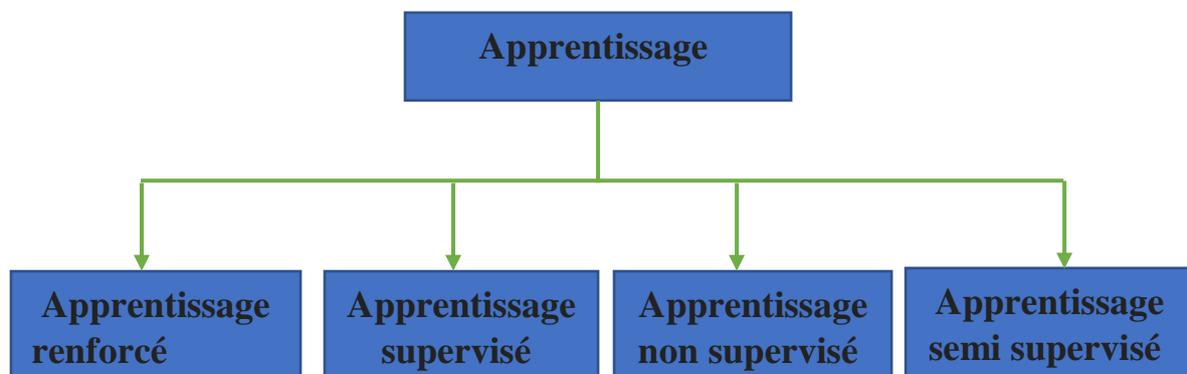


Figure II.9.les types d'apprentissage.

II.5.1. L'apprentissage supervisé

Il fournit ce qu'on appelle des données annotées pour former le modèle, c'est-à-dire qu'il a déjà été associé à une balise ou à une classe cible et nous voulons que l'algorithme soit capable, une fois formé, de prédire cette cible sur de nouvelles données non annotées.

II.5.2. L'apprentissage non supervisé

Dans ce cas, il applique le même algorithme d'apprentissage pour trouver des similitudes et des différences au sein de ces seules données, et pour agréger celles qui partagent des caractéristiques communes avec les données d'entrée, puisque les données d'entrée ne sont pas annotées.

II.5.3. L'apprentissage semi supervisé

Ce sont des méthodes très intéressantes qui tirent parti des deux mondes (supervisé et non supervisé), mais elles comportent leur lot de difficultés : certaines données sont annotées et d'autres non.

II.5.4. L'apprentissage renforcé

Il se base sur un cycle d'expérience / récompense et améliore les performances à chaque itération. Une analogie souvent citée est celle du cycle de dopamine : une "bonne" expérience augmente la dopamine et donc augmente la probabilité que l'agent répète l'expérience. [11]

II.5.5. Types de modèles d'apprentissage en profondeur

L'apprentissage en profondeur offre une grande précision dans les ensembles de données encombrés. Les algorithmes d'apprentissage en profondeur ont été utilisés avec un réseau de neurones profonds (DNN), un réseau de neurones convolutifs (CNN) et un réseau neuronal récurrent (RNN) et une mémoire à long terme (LSTM) dans ces applications.

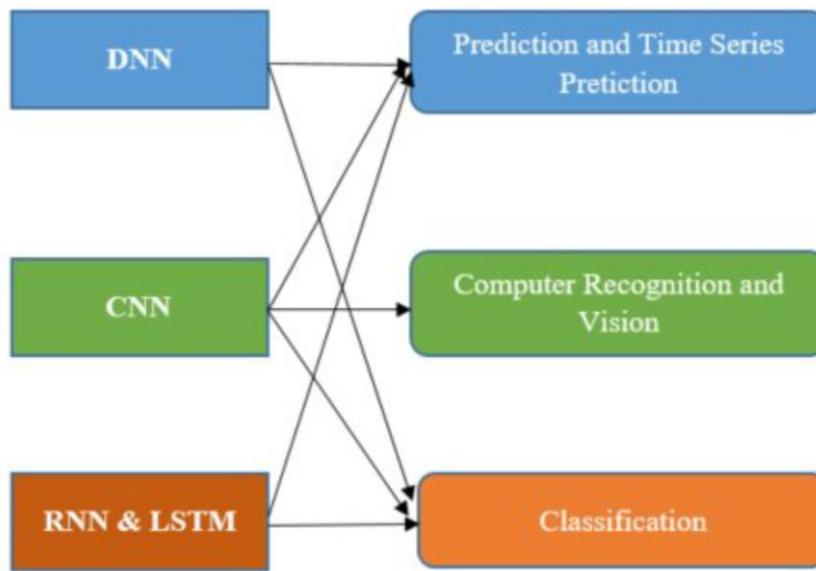


Figure II.10. Les algorithmes d'apprentissage en profondeur.

II.5.6. Réseau de neurones convolutifs (CNN)

Les réseaux de neurones convolutifs ont une méthodologie similaire à celle des méthodes traditionnelles d'apprentissage supervisé : ils reçoivent des images en entrée, détectent les features automatiquement de chacune d'entre elles, puis entraînent un classifieur dessus. Donc les CNN réalisent eux-mêmes tout le boulot fastidieux d'extraction et description de features. Aujourd'hui, les réseaux de neurones convolutifs, aussi appelés CNN ou ConvNet pour Convolutional Neural Network, sont toujours les modèles les plus performants pour la classification d'images.

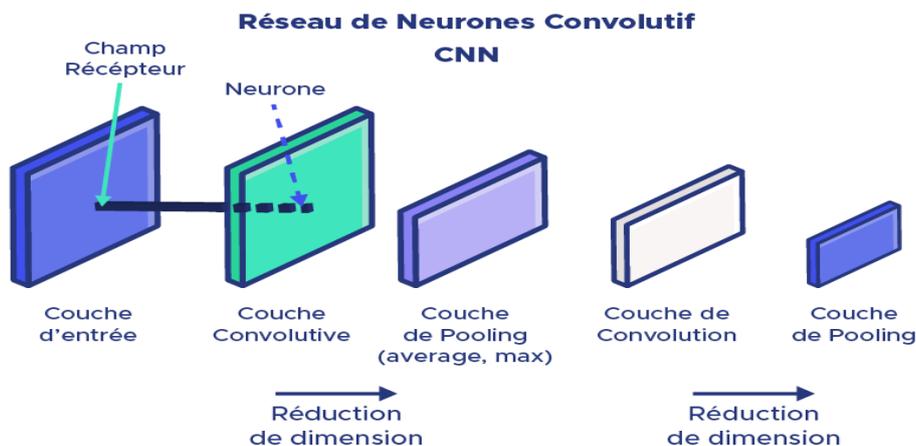


Figure II.11. Représentation d'un Réseau de neurones Convolutif.

Les réseaux de neurones convolutifs désignent une sous-catégorie de réseaux de neurones. Cependant, les CNN sont spécialement conçus pour traiter des images en entrée. Leur architecture est alors plus spécifique : elle est composée de deux blocs principaux.

- 1- Le premier bloc fait la particularité de ce type de réseaux de neurones, puisqu'il fonctionne comme un extracteur de features. Pour cela, en appliquant des opérations de filtrage par convolution. La première couche filtre l'image avec plusieurs noyaux de convolution, et renvoie des "feature maps", qui sont ensuite normalisées (avec une fonction d'activation) et/ou redimensionnées. Ce procédé peut être réitéré plusieurs fois : on filtre les features maps obtenues avec de nouveaux noyaux, ce qui nous donne de nouvelles features maps à normaliser et redimensionner, et qu'on peut filtrer à nouveau, et ainsi de suite. Finalement, les valeurs des dernières feature maps sont concaténées dans un vecteur. Ce vecteur définit la sortie du premier bloc, et l'entrée du second.
- 2- Le second bloc : Les valeurs du vecteur en entrée sont transformées (avec plusieurs combinaisons linéaires et fonctions d'activation) pour renvoyer un nouveau vecteur en sortie. Ce dernier vecteur contient autant d'éléments qu'il y a de classes : l'élément i représente la probabilité que l'image appartienne à la classe i . Comme pour les réseaux de neurones ordinaires, les paramètres des couches sont déterminés par rétropropagation du gradient : l'entropie croisée est minimisée lors de la phase d'entraînement. Mais dans le cas des CNN, ces paramètres désignent en particulier les features des images.

Voir maintenant les différents types de couches d'un CNN. Il existe quatre types de couches pour un réseau de neurones convolutif : la couche de **convolution**, la couche de **pooling**, la couche de **correction ReLU** et la couche **fully-connected**. [52]

- **Couche convolution**

La convolution, d'un point de vue simpliste, est le fait d'appliquer un filtre mathématique à une image. D'un point de vue plus technique, il s'agit de faire glisser une matrice par-dessus une image, et pour chaque pixel, utiliser la somme de la multiplication de ce pixel par la valeur de la

matrice. Cette technique nous permet de trouver des parties de l'image qui pourraient nous être intéressantes.

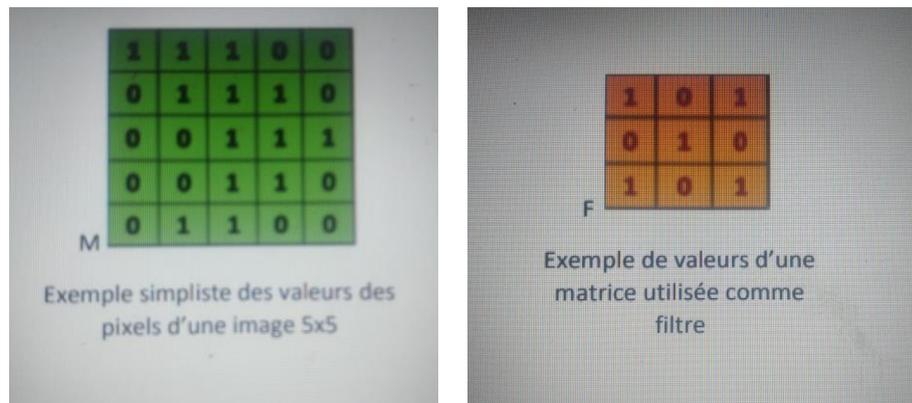


Figure II.12. exemple d'une couche convolution.

- **Le pooling**

Le pooling est un sous-échantillon de l'image permettant à la couche suivante des régions spatiales plus grandes de réduire la dimension (nombre de variable à traiter) mais aussi de rendre le réseau moins sensible aux traductions éventuelles de l'image sans affecter la profondeur. [53]

Ainsi, il regarde les réponses des filtres à différents emplacements, permet de pointer les emplacements des filtres qui donnent les mêmes réponses.

- **Correction ReLU**

ReLU est une fonction qui doit être appliquée à chaque pixel d'une image après convolution, et remplace chaque valeur négative par un 0. Si cette fonction n'est pas appliquée, la fonction créée sera linéaire et le problème XOR persiste puisque dans la couche de convolution, aucune fonction d'activation n'est appliquée. ReLU est très utilisée dans les réseaux de neurones à convolution car il s'agit d'une fonction rapide à calculer : $f(y) = \max(0, y)$ Sa performance est donc meilleure que d'autres fonctions où des opérations coûteuses doivent être effectuées. [54]

- **Fully-connected**

Les neurones d'une couche entièrement connectée sont connectés avec chacun des neurones du volume correspondant à son entrant, c'est-à-dire

avec chacun des neurones de la couche précédente. Si on décompose notre volume d'entrants sous forme vectorielle, on a exactement un réseau de neurones PCM [11].

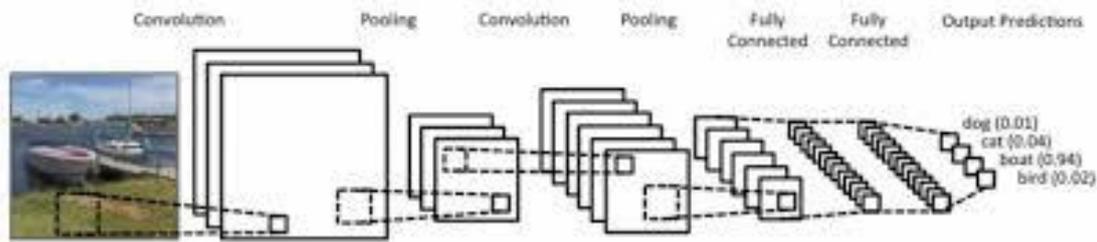


Figure II.13. : Exemple de réseau convolutif deux étapes CONV-POL suivies par deux couches FC.

II.5.7. Modèle VGG16

C'est un modèle de réseau neuronal convolutif proposé par K.Simonyan et A.Zisserman de l'Université d'Oxford dans l'article << Very Deep Convolutional Networks for Large-Scale Image Recognition >>. Le modèle atteint une précision de test de 92.7% dans le top 5 dans ImageNet, qui est un ensemble de données de plus de 14 millions d'images appartenant à 1000 classes. C'était l'un des célèbres modèles soumis à ILSVRC-2014. Il apporte une amélioration par rapport à AlexNet en remplaçant les grands filtres de la taille d'un noyau (11 et 5 dans la première et la deuxième couche convolutive, respectivement) par plusieurs filtres de la taille d'un noyau 3x3 l'un après l'autre. VGG16 a été formé pendant des semaines et utilisait des GPU NVIDIA Titan Black. Figure II.9 et II.10 représente l'architecture du modèle VGG16.

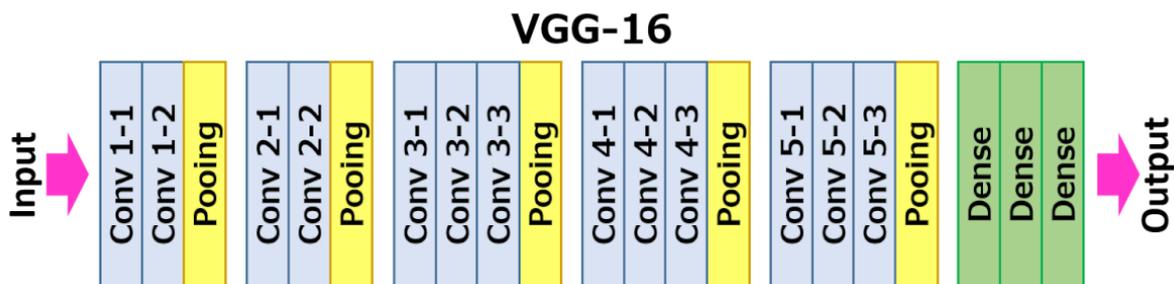


Figure II.14. : Le réseau de neurones VGG16.

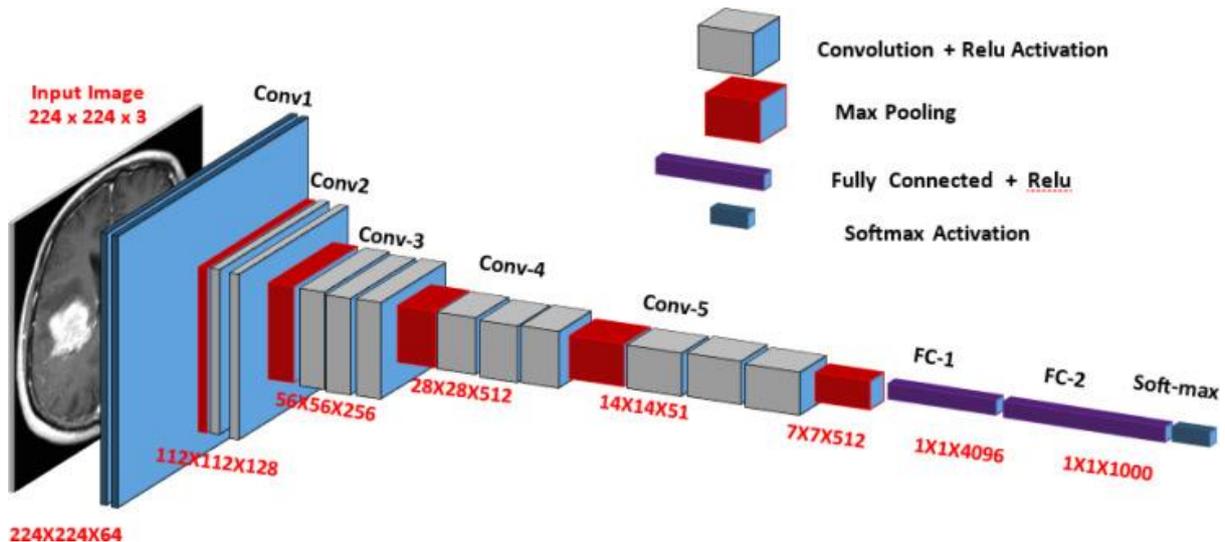


Figure II.15. L'architecture du modèle VGG16.

II.6. Evaluation de la segmentation des images

L'évaluation de la segmentation est généralement basée sur des critères qualitatifs et quantitatifs. Le problème d'évaluation est crucial car il permet au chercheur de :

- Comparaison de la méthode de segmentation avec d'autres méthodes.
- Choisir l'algorithme le plus approprié à son application et ajuster ses paramètres. En fonction du problème à résoudre car le paramétrage des algorithmes est un problème bien connu des chercheurs. Pour comparer objectivement les méthodes de segmentation, il est préférable d'utiliser des images de synthèse dont les caractéristiques et les zones sont parfaitement localisées, dont le principal inconvénient est qu'elles ne reproduisent pas toutes les situations possibles et ne tiennent pas suffisamment compte de la réalité.

II.6.1. Evaluation non-supervisée

L'évaluation non-supervisée des résultats de segmentation, consiste exécuter des mesures d'homogénéité des régions ou de contrastes entre régions.

Ont classifié ces critères en deux grandes catégories :

- **Les critères de Contraste :** s'intéressent à la variabilité inter-régions.

- **Les critères d'adéquation au modèle** : s'intéressent à l'uniformité en intensité ou en couleur à l'intérieur des régions.

II.6.2. Évaluation supervisée

L'évaluation supervisée consiste à comparer le résultat fourni par un algorithme avec une segmentation de référence (vérité-terrain). Cette dernière peut être une image de synthèse ou une image construite à la main par un expert du domaine de l'application dans le cas d'images naturelles, en traçant des contours (par des médecins, des géographes, etc.), en utilisant des outils informatiques de dessin.[51]

Dans le domaine de la segmentation de l'IRM cérébrale, un grand nombre d'images avec des fragments de référence sont fournies par de nombreux laboratoires. Ensuite, les critères basés sur un hachage de référence sont généralement préférés. Deux types d'images peuvent être utilisés :

a) Des images simulées (fantômes)

Ces images reposent sur la simulation du phénomène physique d'acquisition d'une image IRM à partir d'un modèle anatomique réaliste. On peut alors comparer le résultat d'un algorithme de segmentation au modèle anatomique sous-jacent. Les bases de données simulées BrainWeb 1 mises à disposition par le McConnell Brain Imaging Center sont des bases de données très courantes pour l'évaluation d'un algorithme de segmentation d'IRM cérébrales. [50]

b) Des images réelles [50]

Ces images sont segmentées manuellement par un expert. Ce type d'évaluation a l'avantage de se baser sur des « vraies » acquisitions, représentant la réalité. La segmentation manuelle est cependant une tâche fastidieuse et sujette à une variabilité inter-expert et intra-expert non négligeable.

Le résultat de la segmentation de ces images, réelles ou fantômes, est comparé à la segmentation de référence via une mesure de similarité. Plusieurs mesures ont été proposées. Parmi celles-ci, le coefficient de Dice (Dice, 1945). Le coefficient de Dice ou encore appelé Kappa Index permet de mesurer la ressemblance entre deux régions. Numériquement, en notant VP_k le nombre de vrais positifs pour la classe k , FP_k le nombre de faux positifs et FN_k le nombre de faux négatifs, le coefficient de Dice d_k pour la classe k est donné par :

$$d_k = \frac{2VP_k}{2VP_k + FN_k + FP_k}$$

Le coefficient de Dice est entre 0 et 1. Lorsqu'il est égal à 1, la segmentation est parfaite.

Une autre métrique utilisée appelée précision « accuracy », cette métrique est utilisée pour justifier les performances du modèle qui peuvent être définies comme suit :

Précision=Nombre d'images correctement prédites/Nombre total d'images testées×100 %

Conclusion

Dans cet état de l'art, nous avons présenté un certain nombre de méthodes dédiées à la classification et à la segmentation des structures cérébrales saines ou pathologiques classées selon leurs approches. La diversité des techniques est représentative de la complexité du problème de segmentation d'images IRM cérébrales, Ces difficultés sont liées, d'une part, aux objets traités et d'autre part, au particularisme des images IRM. Et nous avons présenté l'apport de l'intelligence artificielle à l'imagerie médicale en particulier les méthodes de l'apprentissage profond. Nous avons marqué leurs modèles utilisés et nous avons détaillé le modèle VGG16 car c'est le modèle que nous intéresse et qui sera implémenté.

Chapitre III

Contribution et résultats

III.1. Introduction

Ce chapitre présente l'environnement de travail la platform utilisée google colab et le langage de Programmation Python avec ces Bibliothèques et l'utilisation du ces bibliothèques dans la platform colab.

Ensuite, on détaille une implémentation du modèle VGG16 dans la plateforme colab, ainsi que le résultats obtenus.

III.2. Environnement de travail



III.2.1. Python

Python est le langage de programmation open source le plus utilisé dans le domaine de l'intelligence artificielle et surtout en Deep Learning vu qu'il contient un nombre important de bibliothèques performantes et utiles pour la vision artificielle et l'utilisation des réseaux de neurones. Il est conçu pour optimiser la productivité des programmeurs en orant des outils de haut niveau et une syntaxe simple à utiliser Depuis quelques années. Notre projet est fait avec le langage de Python dans le cloud et on n'aura pas besoin d'installer un logiciel ou un IDE dans notre ordinateur. Nous citons Les bibliothèques Python suivants :



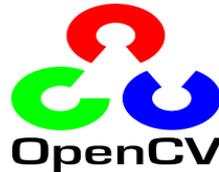
Figure III.1. Les bibliothèques Python.

- **NumPy**



C'est l'une des bibliothèques les plus essentielles lorsqu'on travaille sur des projets en Python. Elle permet des manipulations de tableaux de grande dimension comme les opérations arithmétiques, les conversions, etc. [11]

- **OpenCV**



OpenCV est une bibliothèque graphique libre, initialement développée par Intel, C'est l'une des meilleures bibliothèques de vision par ordinateur et de traitement d'image en temps réel utilisée pour faire des tonnes de choses sur des images et des vidéos.

- **Matplotlib**



Matplotlib est une bibliothèque destinée à tracer et visualiser des données sous formes de graphiques, c'est-à-dire donner une représentation visuelle des résultats. Nous l'utilisons quand nous devons tracer quelques images pour faire des comparaisons. Elle peut être combinée avec les bibliothèques python de calcul scientifique NumPy et SciPy.

- **Pandas**



Cette bibliothèque est utilisée lorsqu'on a besoin d'organiser nos données et de les rendre plus faciles à utiliser et à comprendre. Elle les organise dans des structures appelées « trames de données » et celles-ci aideront avec une représentation claire des données ainsi que leur personnalisation.[11]

Elle propose en particulier des structures de données et des opérations de manipulation de tableaux numériques et de séries temporelles. Pandas est un logiciel libre sous licence BSD. [Wikipédia]



- **SciPy**

C'est une bibliothèque Python open source pour le calcul scientifique et technique. Les principaux packages incluent des modules pour : l'optimisation, l'algèbre linéaire, l'intégration, la FFT, le traitement du signal et de l'image [11]. Scipy utilise les tableaux et matrices du module NumPy.



- **Scikit-learn**

Une bibliothèque Python open source qui propose de nombreuses classifications, des algorithmes de régression et de regroupement comme les support vector machines (SVM), le random forests, le gradient boosting, les k-means et DBSCAN. Elle est utilisée lorsque la création et la formons des modèles de Machine Learning.[11]



- **Tensorflow**

TensorFlow est une bibliothèque d'apprentissage en profondeur open source développée par Google. Il s'agit d'une boîte à outils permettant de résoudre des problèmes mathématiques extrêmement complexes avec aisance.



- **Keras**

Keras est un Framework Python pour l'apprentissage en profondeur. Son USP est la réutilisation du code pour le CPU et le GPU. Keras est compatible avec les versions de Python 2.7 / 3.6 et autres.il a été initialement développé dans le cadre de l'effort de recherche du projet ONEIROS (système

d'exploitation de robot intelligent neuro-électronique à composition open-ended). Keras peut désormais être exécuté sur TensorFlow.

III.2.2. Google Colab

Google Colab ou « Colaboratory » est un service cloud gratuit hébergé par Google pour encourager la recherche sur le Deep Learning et l'Intelligence Artificielle, où souvent l'obstacle à l'apprentissage et au succès est l'exigence d'une puissance de calcul considérable.

Cette plateforme permet alors d'entraîner des modèles de Deep Learning directement dans le cloud. Sans donc avoir besoin d'installer quoi que ce soit sur l'ordinateur à l'exception d'un navigateur [55].

Colab permet :

- De développer des applications en Deep Learning en utilisant des bibliothèques Python populaires telles que Keras, TensorFlow, PyTorch et OpenCV.
- D'utiliser un environnement de développement (Jupyter Notebook) qui ne nécessite aucune configuration.
- L'accès à un processeur graphique GPU et au TPU (Tensor Processor Unit) gratuitement.

Google Colaboratory fonctionne sur les serveurs Google donc nous n'avons rien installé. De plus, les documents Colab (Jupyter Notebook) sont enregistrés directement sur le compte Google Drive.

Avantages de Colab : [11]

En plus d'être facile à utiliser, le Colab est assez flexible dans sa configuration et fait beaucoup de travail pour nous :

- Prise en charge de Python 2.7 et Python 3.6
- Accélération GPU gratuite
- Bibliothèques préinstallées : les principales bibliothèques de Python comme TensorFlow, Scikit-learn, entre autres, sont préinstallées et prêtes à être importées.
- Prend en charge les commandes Bash (langage de commande).

III.2.2.1. Google Colab

Suivre le lien <https://colab.research.google.com/> et se connecter avec le compte Gmail.



Figure III.2. La page d'accueil de la plateforme Colab.

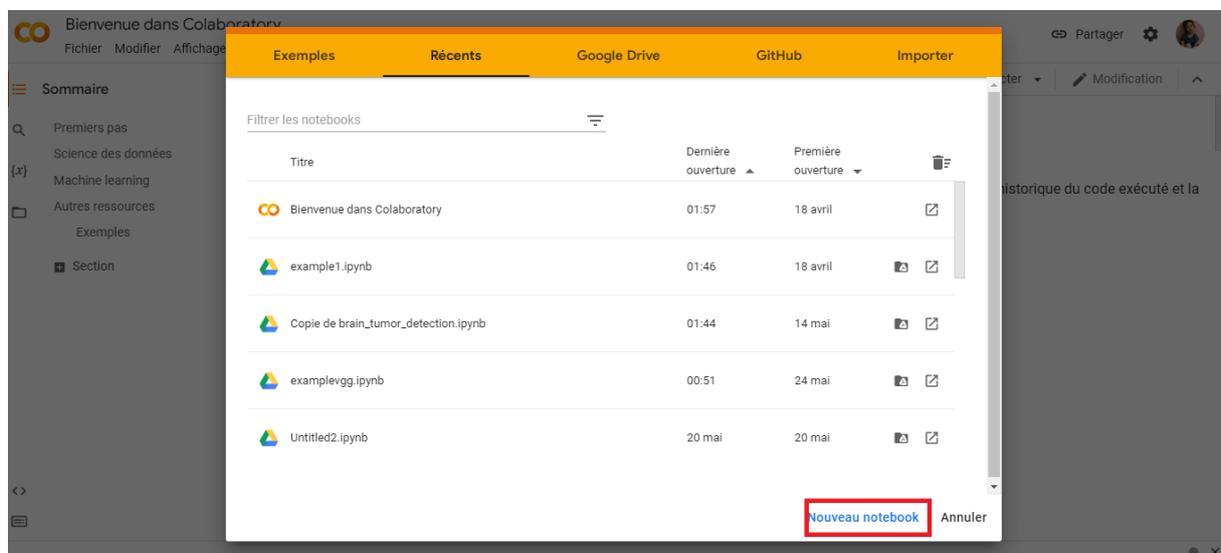


Figure III.3. Illustration d'ouvrir un notebook.

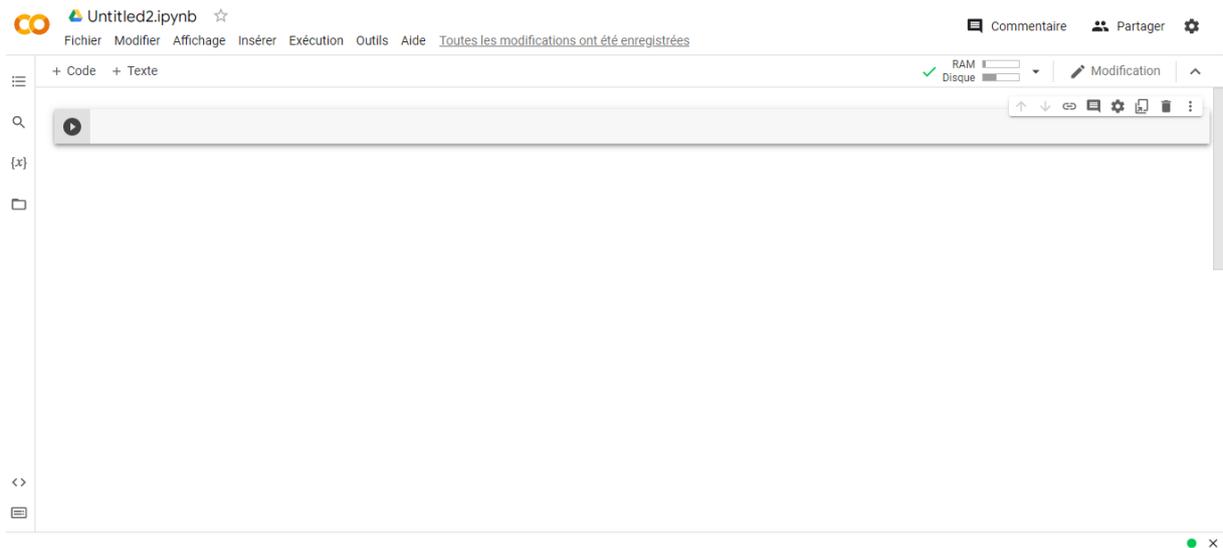


Figure III.4. Illustration du nouveau notebook ouvert dans la plateforme colab.

III.2.2.2. Google drive

- Pour monter le drive dans le dossier « gdrive », on exécute la commande :

```
From google.colab import drive  
drive.mount('/content/gdrive')
```



Figure III.5. Illustration : comment utiliser drive.

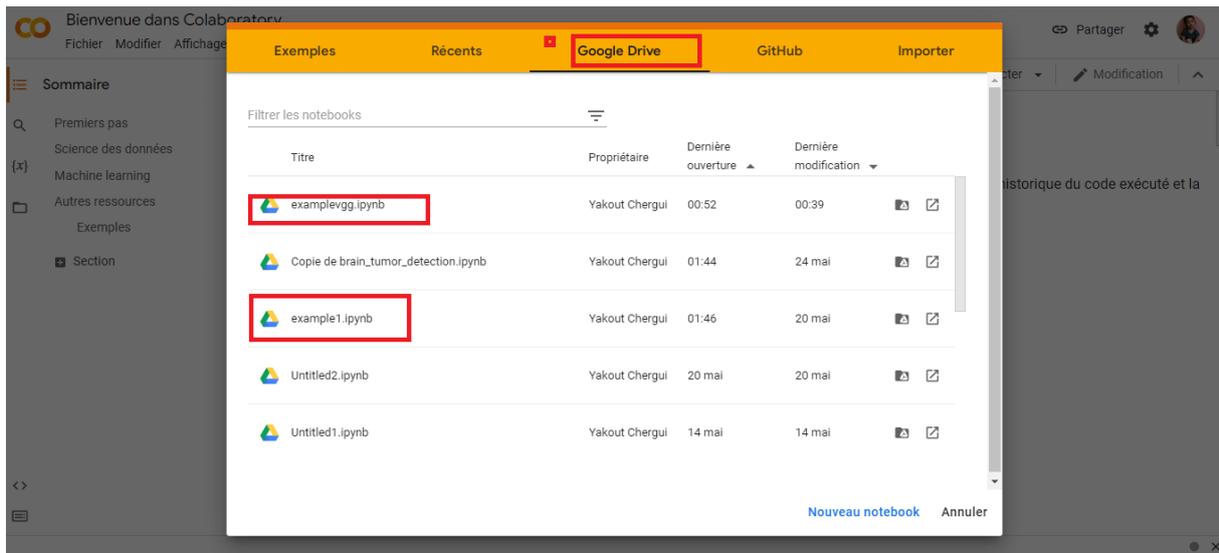


Figure III.6. Illustration : comment ouvrir un notebook dans drive.

- Importer les bibliothèques Python (Python Libraires)



Figure III.7. Illustration : comment importer une bibliothèque dans colab.

- Choisir l'accélérateur matériel

Dans la barre des options choisir "exécution"(Runtime) puis "modifier le type d'exécution" (Change Runtime Type) et mettre l'option accélérateur matériel (Hardware Accelerator) en mode GPU ou TPU (Tensor Processing Unit). bien expliquer dans ces figures suivantes :

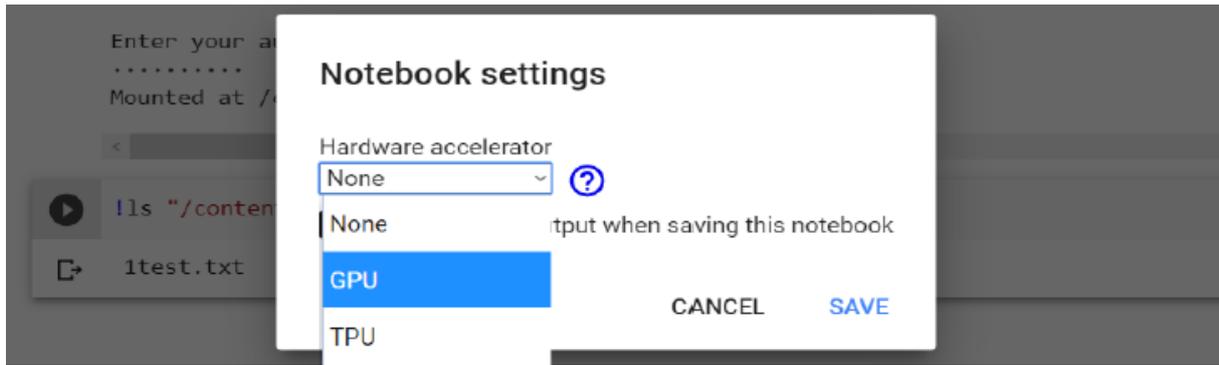


Figure III.8. Illustration : choisir l'accélération matérielle.

- Entraîner sur GPU

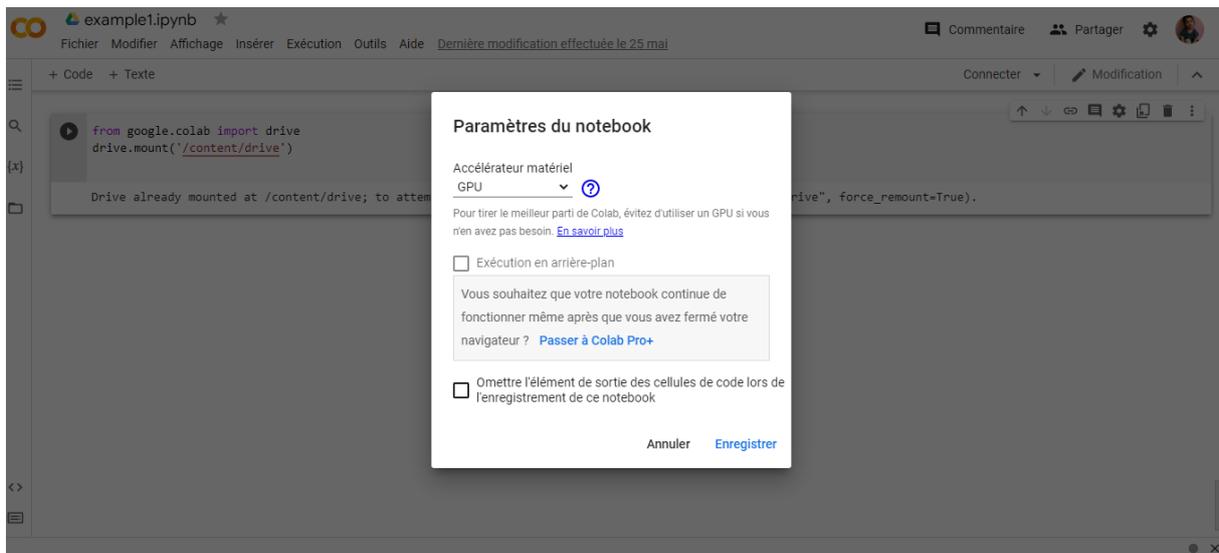


Figure III.9. L'accélération matérielle GPU.

- Entraîner sur TPU

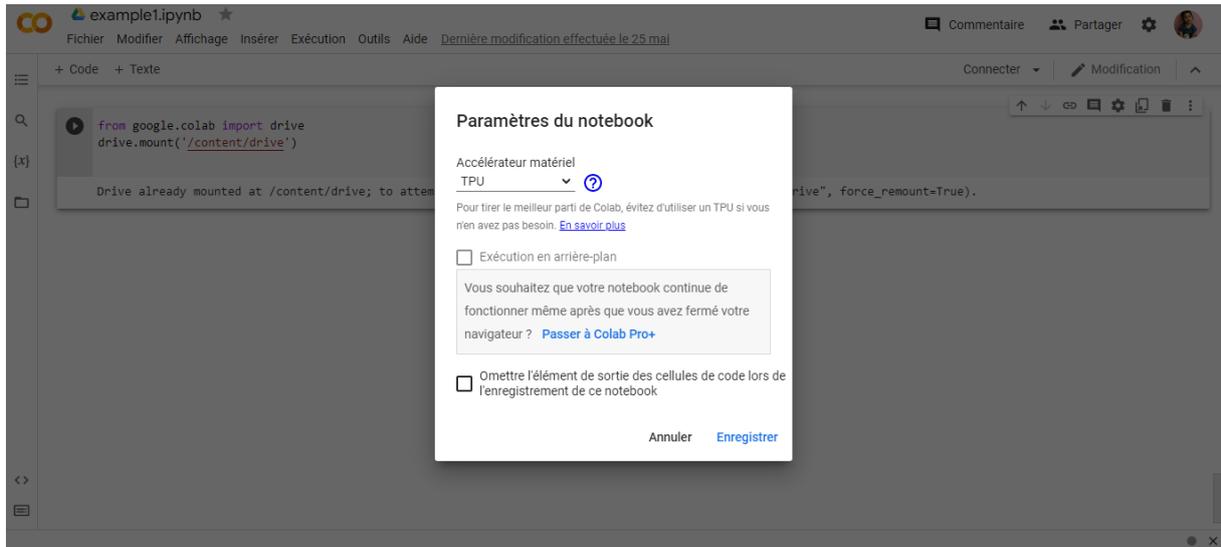


Figure III.10. L'accélération matérielle TPU.

III.3. Modèle VGG16

III.3.1. Implémentation

Afin d'implémenter notre modèle VGG16 pour avoir classifié les images IRM en images saines et malades, on va suivre les étapes d'implémentation suivantes [11]:

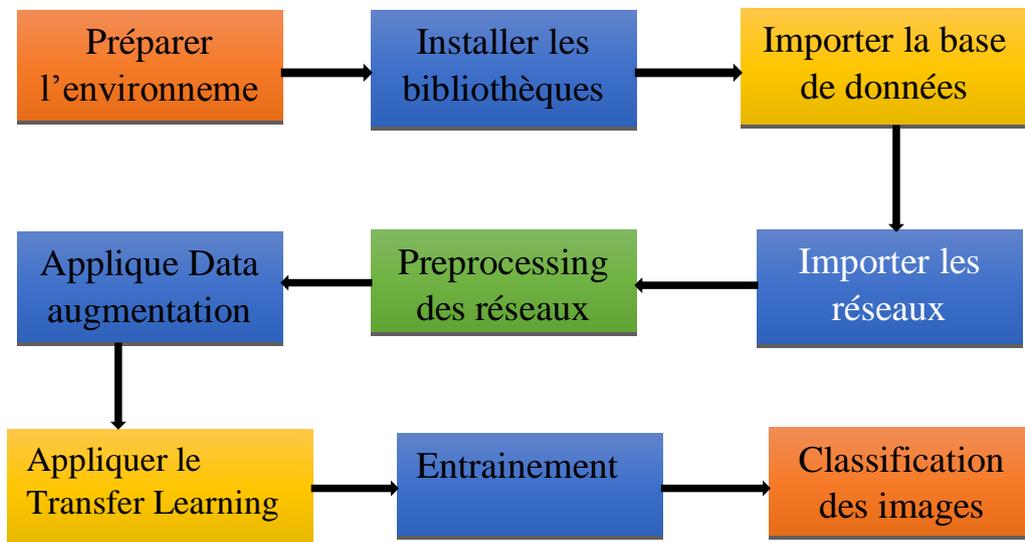


Figure III.11. Les étapes d'implémentation de réseau de Classification.

Dans notre implémentation nous avons basé sur des implémentations des réseaux de neurones en utilisant python issues de « kaggle »¹, Kaggle est un site représentant une communauté de l'intelligence artificielle et des sciences de données. Kaggle regroupe tous les codes et les données dont on a besoin pour faire le travail de science des données, et contient plus de 50 000 ensembles de données publics et 400 000 blocs-notes publics pour conquérir n'importe quelle analyse en un rien de temps.

- **Préparer l'environnement**

On va utiliser le modèle VGG16 pour segmenter les images IRM

```

# Import necessary modules
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import VGG16
from tensorflow.keras.layers import Input
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import AveragePooling2D
from tensorflow.keras.layers import Dropout
from tensorflow.keras.layers import Flatten
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.utils import to_categorical

from sklearn.preprocessing import LabelBinarizer
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report, confusion_matrix

from imutils import paths
import numpy as np
    
```

Figure III.12. Code de préparation de réseau et de l'environnement

- **Installer les bibliothèques**

```

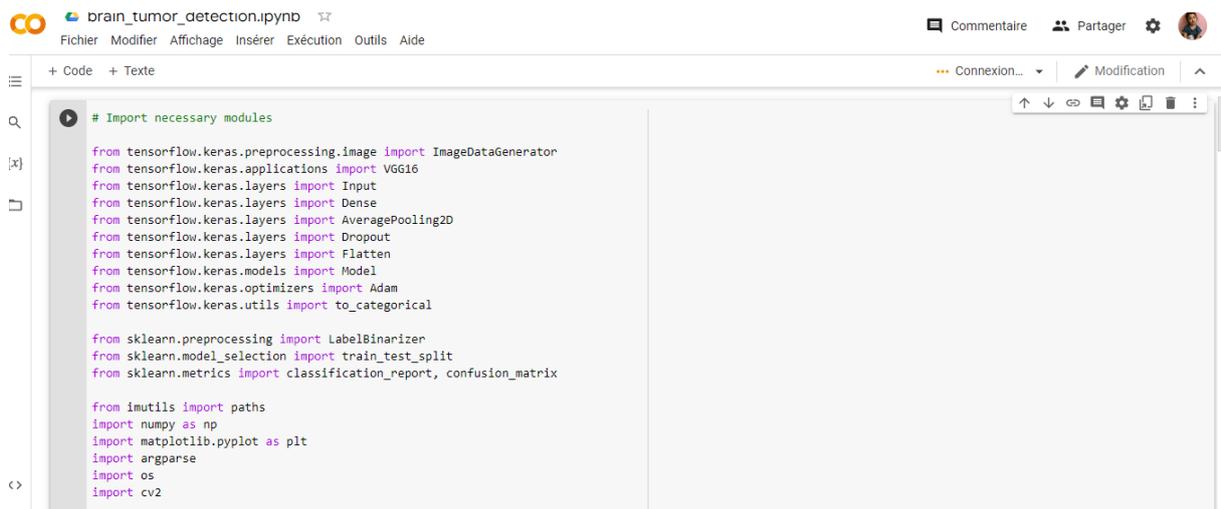
!pip install numpy
!apt-get install matplotlib
!pip install matplotlib
!pip install scipy
!pip install sklearn
import shutil

Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages (1.18.4)
Reading package lists... Done
Building dependency tree
Reading state information... Done
E: Unable to locate package matplotlib
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: matplotlib in /usr/local/lib/python3.7/dist-packages (3.2.2)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib) (1.4.2)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.7/dist-packages (from matplotlib) (0.11.0)
Requirement already satisfied: numpy>=1.11 in /usr/local/lib/python3.7/dist-packages (from matplotlib) (1.18.4)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib) (3.0.9)
Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib) (2.8.2)
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.7/dist-packages (from kiwisolver=1.0.1->matplotlib) (4.2.0)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.1->matplotlib) (1.15.0)
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: scipy in /usr/local/lib/python3.7/dist-packages (1.4.1)
Requirement already satisfied: numpy>=1.13.3 in /usr/local/lib/python3.7/dist-packages (from scipy) (1.18.4)
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: sklearn in /usr/local/lib/python3.7/dist-packages (0.0)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.7/dist-packages (from sklearn) (1.0.2)
✓ 13 s terminée à 08:11
    
```

Figure III.13. Code d'installer les bibliothèques.

¹ <https://www.kaggle.com/>

- **Importer les bibliothèques**



```
# Import necessary modules
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import VGG16
from tensorflow.keras.layers import Input
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import AveragePooling2D
from tensorflow.keras.layers import Dropout
from tensorflow.keras.layers import Flatten
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.utils import to_categorical

from sklearn.preprocessing import LabelBinarizer
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report, confusion_matrix

from imutils import paths
import numpy as np
import matplotlib.pyplot as plt
import argparse
import os
import cv2
```

Figure III.14. Importer les bibliothèques utilisées.

- **Base de données d'apprentissage**

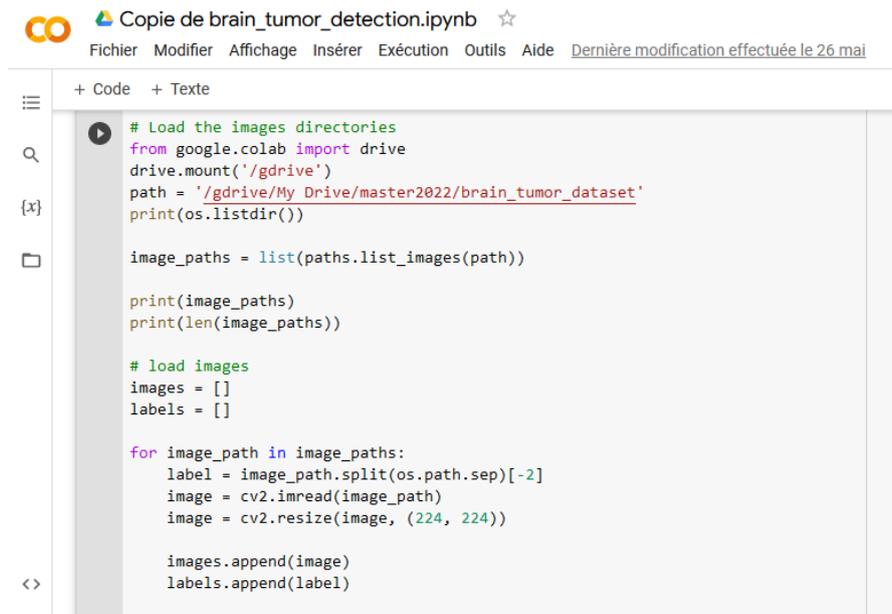
Afin de réaliser ce projet de détection des anomalies cérébrales, il fallait construire une base de données contenant des images IRM. Notre base de données a été obtenue à partir de la plateforme Open Source «kaggle/dataset»². Elle est composée d'environ 253 images IRM au format JPG présentant en 2 dossiers :

- Dossier 1 : présence de la tumeur nommée « Yes » avec 155 images
- Dossier 2 : absence de la tumeur nommée « No » avec 98 images.

On a utilisé une technique de traitement d'image permettant de générer plus de données/d'images à partir des données originales pour résoudre le problème de manque des images IRM car ces images sont très peu pour entraîner nos réseaux de neurones. Cette technique est appelée « Data augmentation », et elle sera expliquée bien plus loin dans ce chapitre.

² <https://www.kaggle.com/datasets/jjprotube/brain-mri-images-for-brain-tumor-detection>

- Importer la base de données



```
CO Copie de brain_tumor_detection.ipynb ☆
Fichier Modifier Affichage Insérer Exécution Outils Aide Dernière modification effectuée le 26 mai

+ Code + Texte

# Load the images directories
from google.colab import drive
drive.mount('/gdrive')
path = '/gdrive/My Drive/master2022/brain_tumor_dataset'
print(os.listdir())

image_paths = list(paths.list_images(path))

print(image_paths)
print(len(image_paths))

# load images
images = []
labels = []

for image_path in image_paths:
    label = image_path.split(os.path.sep)[-2]
    image = cv2.imread(image_path)
    image = cv2.resize(image, (224, 224))

    images.append(image)
    labels.append(label)
```

Figure III.15. Code d'extraction du fichier compressé.

- L'affichage des éléments de base de données

```
#affichage de 12 exemples d'images saines
w=20
h=20
fig=plt.figure(figsize=(20,20))
columns=4
rows=3
for i in range(1, columns*rows +1):
    fig.add_subplot(rows,columns,i)
    plt.imshow(images[i+220])
plt.show()
```

Figure III.16. Code d'afficher les images saines.

```
#affichage de 12 exemples d'images malades
w=20
h=20
fig=plt.figure(figsize=(20,20))
columns=4
rows=3
for i in range(1, columns*rows +1):
    fig.add_subplot(rows,columns,i)
    plt.imshow(images[i*5])
plt.show()
```

Figure III.17. Code d'afficher les images malades.

- Diviser le données en données d'apprentissage et de testes

```
#Split the dataset
(train_X, test_X, train_Y, test_Y) = train_test_split(images, labels, test_size= 0.10, random_state= 42, stratify= labels)
```

Figure III.18. Code de Division des données apprentissage/testes

- Augmentation d'images (Data Augmentation)

C'est une technique appliquée à des données d'entraînement existantes pour créer artificiellement de nouvelles données d'entraînement. Les techniques d'augmentation peuvent créer des variations des images qui peuvent améliorer la capacité des modèles à généraliser ce qu'ils ont appris à de nouvelles images telle que les nouvelles images à partir à la même classe que les images originales.

Il existe plusieurs types d'augmentation des données, on a proposé quelques-unes, ils sont résumés dans la figure suivante :

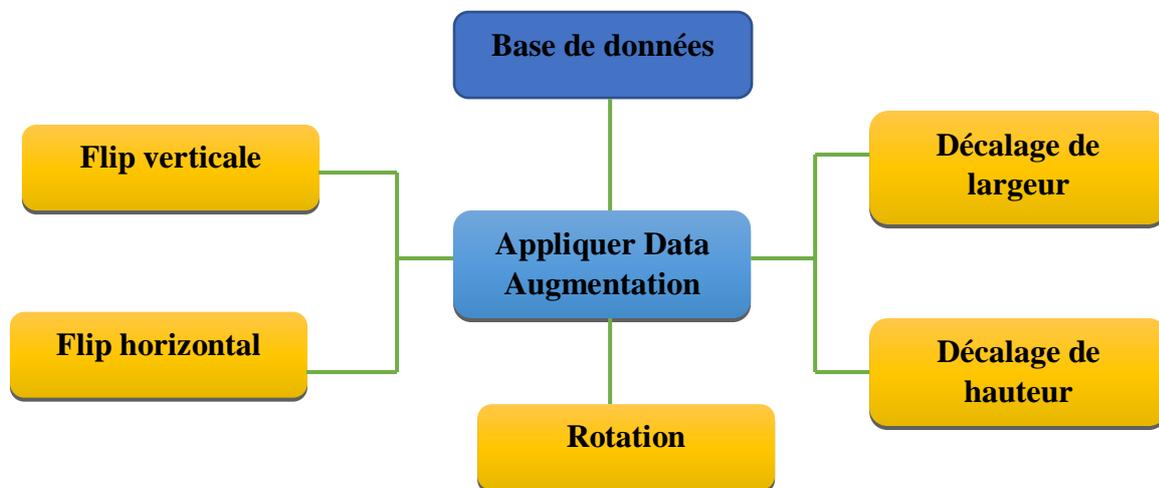


Figure III.19 : quelques types d'augmentation de données.

On a utilisé la technique de rotation pour l'augmentation des données automatiquement à l'aide de la bibliothèque de Deep Learning « Keras » par la classe ImageDataGenerator.

```
# Build the Image Data Generator
train_generator = ImageDataGenerator(fill_mode= 'nearest', rotation_range= 15)
```

Figure III.20 : l'application de l'augmentation des données.

- Construction et compilation du modèle

```

# Build the Image Data Generator
train_generator = ImageDataGenerator(fill_mode= 'nearest', rotation_range= 15)

# Build the model
base_model = VGG16(weights= 'imagenet', input_tensor= Input(shape = (224, 224, 3)), include_top= False)
base_input = base_model.input
base_output = base_model.output
base_output = AveragePooling2D(pool_size=(4, 4))(base_output)
base_output = Flatten(name="flatten")(base_output)
base_output = Dense(64, activation="relu")(base_output)
base_output = Dropout(0.5)(base_output)
base_output = Dense(2, activation="softmax")(base_output)

# Freeze the layers
for layer in base_model.layers:
    layer.trainable = False

# Compile the model
model = Model(inputs = base_input, outputs = base_output)
model.compile(optimizer= Adam(learning_rate= 1e-3), metrics= ['accuracy'], loss= 'binary_crossentropy')

# Let's see the architecture summary of our model
model.summary()

```

Figure III.21. Construction et compilation du modèle

- Paramétrage et transfert d'apprentissage

```

#store the following hyperparameters
batch_size = 8
train_steps = len(train_X) // batch_size
validation_steps = len(test_X) // batch_size
epochs = 10

```

Figure III.22. Code du paramétrage

Une époque d'apprentissage (epochs =10 dans notre cas) signifie que l'algorithme d'apprentissage a effectué un passage dans l'ensemble de données d'apprentissage, où les exemples ont été séparés en groupes de « taille de lot » (batch size) sélectionnés aléatoirement [58].

Le modèle est utilisé avec beaucoup de paramètres mais la majorité est réglé initialement en utilisant le transfert d'apprentissage :

```

Total params: 14,747,650
Trainable params: 32,962
Non-trainable params: 14,714,688

```

- **L'apprentissage du modèle**

```

+ Code + Texte
epochs = 10

#let's train our model
# Fit the model
history = model.fit_generator(train_generator.flow(train_X, train_Y, batch_size= batch_size),
                             steps_per_epoch= train_steps,
                             validation_data = (test_X, test_Y),
                             validation_steps= validation_steps,
                             epochs= epochs)
    
```

Figure III.23. Code d'apprentissage de modèle

- **Test du modèle**

```

# Evaluate the model
predictions = model.predict(test_X, batch_size= batch_size)
predictions = np.argmax(predictions, axis= 1)
actuals = np.argmax(test_Y, axis= 1)
    
```

Figure III.24. Code d'évaluation du modèle

- **Affiche de résultats d'évaluation de la classification**

```

+ Code + Texte
actuals = np.argmax(test_Y, axis= 1)

# Print Classification report and Confusion matrix
print(classification_report(actuals, predictions, target_names= label_binarizer.classes_))

cm = confusion_matrix(actuals, predictions)
print(cm)

# Final accuracy of our model
total = sum(sum(cm))
accuracy = (cm[0, 0] + cm[1, 1]) / total
print("Accuracy: {:.4f}".format(accuracy))
    
```

Figure III.25. Code d'affichage des résultats (précision).

```

+ Code + Texte
# Plot the losses and accuracies
N = epochs
plt.style.use("ggplot")
plt.figure()
plt.plot(np.arange(0, N), history.history["loss"], label= "train_loss")
plt.plot(np.arange(0, N), history.history["val_loss"], label= "val_loss")

plt.plot(np.arange(0, N), history.history["accuracy"], label= "train_acc")
plt.plot(np.arange(0, N), history.history["val_accuracy"], label= "val_acc")

plt.title("Training Loss and Accuracy on Brain Dataset")
plt.xlabel("Epoch")
plt.ylabel("Loss / Accuracy")
plt.legend(loc= "lower left")
plt.savefig("plot.jpg")
    
```

Figure III.26. Code des tracés des courbes (précision et perte).

III.3.2. Résultats

- Résultats de téléchargement des images avec affichage de quels exemples d'images saines et malades

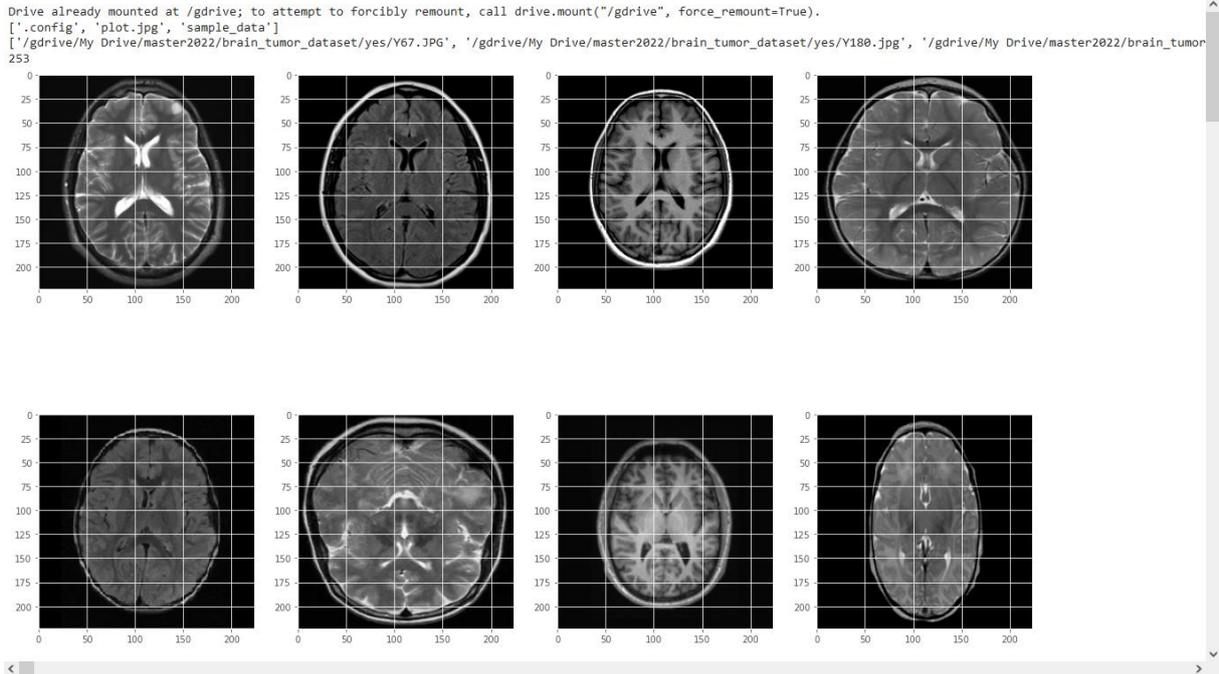


Figure III.27. Les images saines.

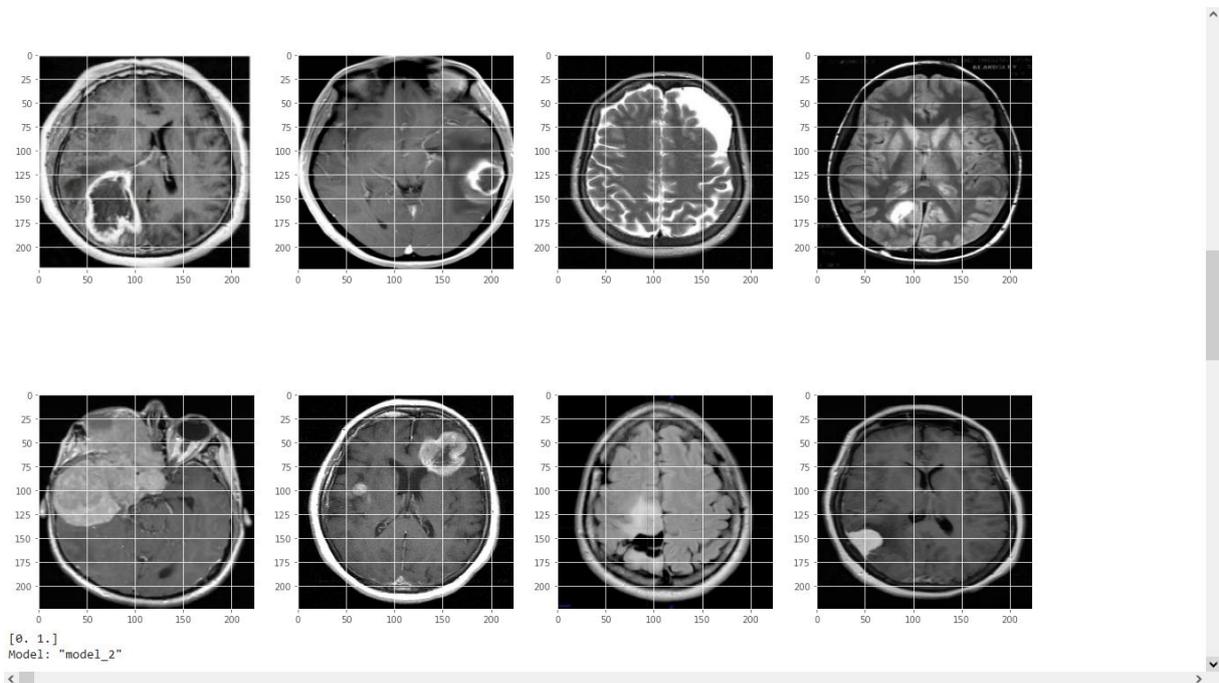


Figure III.28. Les images maladies.

- Résultats de construction et compilation du modèle donnés sous forme de son architecture et ses paramètres

```
[0. 1.]
Model: "model_2"
```

Layer (type)	Output Shape	Param #
input_3 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
average_pooling2d_2 (AveragePooling2D)	(None, 1, 1, 512)	0
flatten (Flatten)	(None, 512)	0
dense_4 (Dense)	(None, 64)	32832
dropout_2 (Dropout)	(None, 64)	0
dense_5 (Dense)	(None, 2)	130

```
-----
Total params: 14,747,650
Trainable params: 32,962
Non-trainable params: 14,714,688
```

Figure III.29. Visualisation du modèle.

- Résultats des itérations d'apprentissage

```

/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:127: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use
Epoch 1/10
28/28 [=====] - 111s 4s/step - loss: 0.7079 - accuracy: 0.5890 - val_loss: 0.6252 - val_accuracy: 0.6154
Epoch 2/10
28/28 [=====] - 109s 4s/step - loss: 0.6542 - accuracy: 0.6164 - val_loss: 0.5882 - val_accuracy: 0.6154
Epoch 3/10
28/28 [=====] - 109s 4s/step - loss: 0.6179 - accuracy: 0.6256 - val_loss: 0.5490 - val_accuracy: 0.7692
Epoch 4/10
28/28 [=====] - 109s 4s/step - loss: 0.5879 - accuracy: 0.6849 - val_loss: 0.5091 - val_accuracy: 0.8846
Epoch 5/10
28/28 [=====] - 110s 4s/step - loss: 0.5961 - accuracy: 0.6849 - val_loss: 0.4849 - val_accuracy: 0.8846
Epoch 6/10
28/28 [=====] - 109s 4s/step - loss: 0.5566 - accuracy: 0.7352 - val_loss: 0.4642 - val_accuracy: 0.8846
Epoch 7/10
28/28 [=====] - 109s 4s/step - loss: 0.5524 - accuracy: 0.7443 - val_loss: 0.4313 - val_accuracy: 0.9231
Epoch 8/10
28/28 [=====] - 109s 4s/step - loss: 0.5206 - accuracy: 0.7717 - val_loss: 0.4106 - val_accuracy: 0.9231
Epoch 9/10
28/28 [=====] - 109s 4s/step - loss: 0.4868 - accuracy: 0.7945 - val_loss: 0.3841 - val_accuracy: 0.8846
Epoch 10/10
28/28 [=====] - 109s 4s/step - loss: 0.4880 - accuracy: 0.7900 - val_loss: 0.3758 - val_accuracy: 0.8846

```

Figure III.30. Visualisation de l'apprentissage.

- Evaluation du modèle

	precision	recall	f1-score	support
no	1.00	0.70	0.82	10
yes	0.84	1.00	0.91	16
accuracy			0.88	26
macro avg	0.92	0.85	0.87	26
weighted avg	0.90	0.88	0.88	26

```

[[ 7  3]
 [ 0 16]]
Accuracy: 0.8846

```

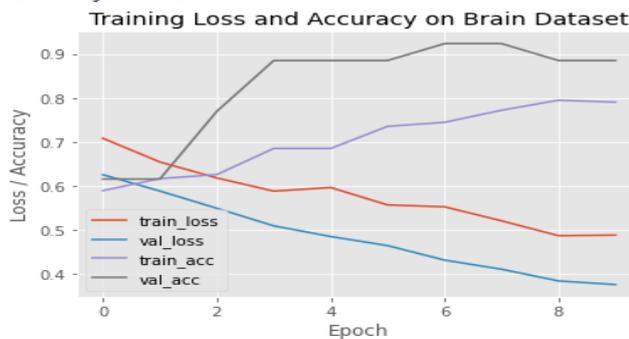


Figure III.31. La courbe de perte(loss) et la courbe de précision(accuracy)de réseau VGG16.

Nous remarquons que la précision du modèle augmente et se stabilise à une valeur proche de 0.90, ce qui est un bon résultat pour nos données

III.4. Comparaison entre les différents accélérateurs matériels

On a fait cette comparaison en termes de temps d'exécution entre les trois accélérateurs CPU, GPU et TPU.

- **Exécution avec le CPU**

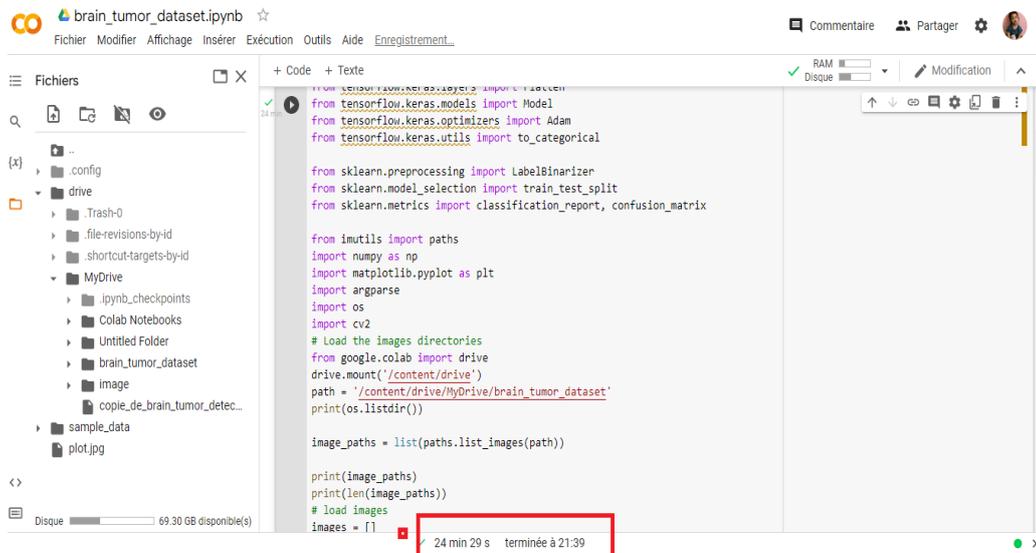


Figure III.32. Le temps d'exécution avec le CPU.

- **Exécution avec le GPU**



Figure III.33. Le temps d'exécution avec le GPU.

- **Exécution avec le TPU**

```

from tensorflow.keras.layers import Flatten
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.utils import to_categorical

from sklearn.preprocessing import LabelBinarizer
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report, confusion_matrix

from imutils import paths
import numpy as np
import matplotlib.pyplot as plt
import argparse
import os
import cv2

# Load the images directories
from google.colab import drive
drive.mount('/content/drive')
path = '/content/drive/MyDrive/brain_tumor_dataset'
print(os.listdir())

image_paths = list(paths.list_images(path))

print(image_paths)
print(len(image_paths))
# load images

```

24 min 11 s terminée à 20:52

Figure III.34. Le temps d'exécution avec le TPU.

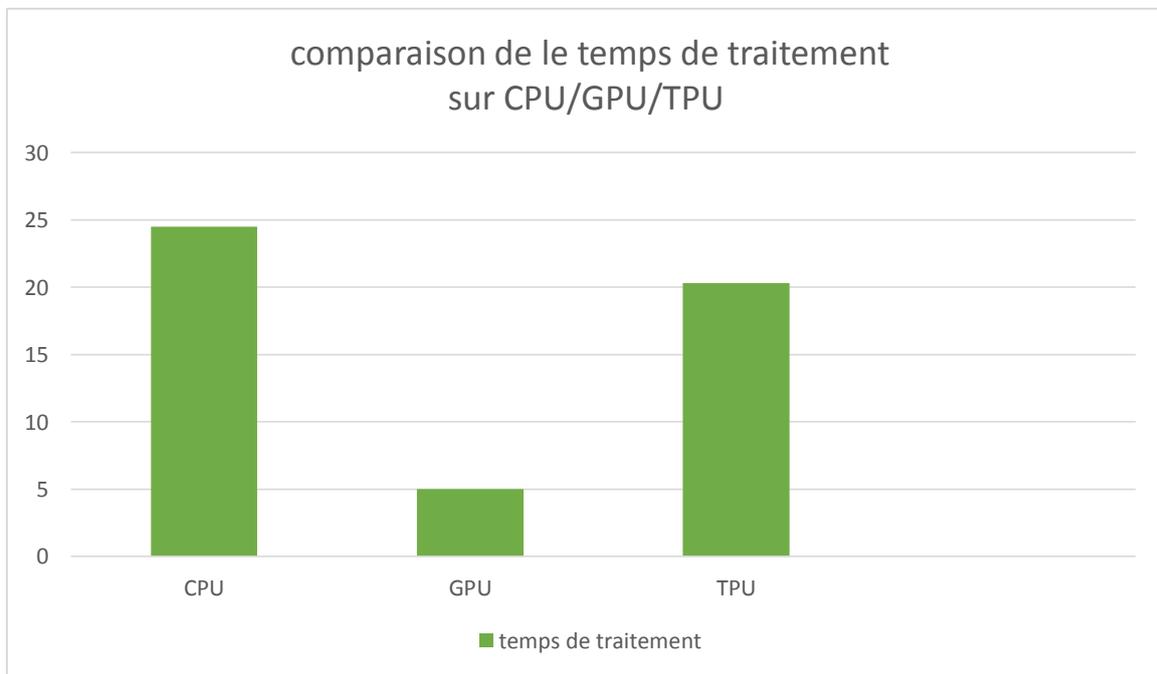


Figure III.35. L'histogramme de la Comparaison entre les différents accélérateurs matériels.

Nous remarquons que le temps d'exécution de GPU est plus court 5 fois que CPU et 4 fois que TPU.

Conclusion

Dans ce chapitre, nous avons présenté la plateforme de travail «google colab » et le langage de programmation « Python » avec certaines bibliothèques « OpenCV, Keras, ... ». Nous avons remarqué que l'environnement google colab est complet et très bénéfique, il nous permet l'implémentation de tous nos codes en offrant de très bons ressources matériels et logicielles qui nous permettent de faire nos implémentations dans de très bonnes conditions. Il nous offre aussi les différents accélérateurs matériels (CPU/GPU/TPU), ce qui nous encourage aussi à implémenter plus de modèles de deep Learning qui sont gourmands en termes de temps d'exécution.

En suite, Nous avons fait l'implémentation du modèle préféré des CNN (modèle de deep Learning) qui est le modèle VGG16 pour classier des images IRM cérébrales en images saines et malades. Nous avons suivies les étapes déjà faite en littérature [12] et en exploitant quelques ressources de codes et de données existantes « kaggle ». Nous avons vu comment prétraiter les images, construire et compiler le modèle et l'application de transfert de l'apprentissage. Ainsi que l'évaluation du modèle. Nous avons conclu que le modèle peut classier nos données avec précision proche de 90%.

En fin, Nous avons fait une comparaison d'exécution en utilisant trois accélérateurs offertes par le cloud (CPU/GPU/TPU) et nous concluons que l'accélérateur GPU est plus rapides ce qui prouve l'utilité de l'implémentation des modèles de deep Learning dans l'environnement Cloud.

Conclusion générale

Aujourd'hui, l'imagerie par résonance magnétique est, un outil puissant permettant l'observation in vivo de l'anatomie cérébrale. Utilisée en routine clinique, la multiplicité des pondérations d'acquisition permet aux médecins d'accéder à une information riche, abondante et particulièrement adaptée au diagnostic des tumeurs cérébrales.

Ce travail de recherche vise à proposer une méthode de classification des données fournies par les images IRM pour la détection des tumeurs cérébrales dans le cadre de l'aide au diagnostic de ces dernières.

Dans une première étape, nous nous sommes attachées à présenter le contexte de notre étude. Nous avons tout d'abord parlé de quelques notions de l'image, dans lequel nous allons voir la définition de l'image et les types, le traitement de la chaîne d'image. Puis, nous avons présenté l'anatomie cérébrale, nous allons voir le cerveau et ses composants, le cervelet et le troc cérébral. Enfin, nous avons expliqué le principe des images à résonance magnétique et ses types et ses caractéristiques, ainsi nous allons voir la qualité d'image acquise par IRM et le matériel utilisé en IRM.

Dans une seconde étape, nous avons produit un état de l'art des principales méthodes de segmentation/classification des images IRM cérébrales. Premièrement, nous avons expliqué les différentes méthodes pour segmenter une image.

Deuxièmes, nous avons mentionné les méthodes basées sur la classification supervisée (réseaux de neurones, KNN) et les méthodes basées sur la classification non supervisée (K-moyens, FCM, PCM) et nous avons présenté l'utilité de l'intelligence artificielle en imagerie médicale et notre méthode d'application (modèle VGG16 de CNN).

En fin, nous avons détaillé une implémentation du modèle VGG16 dans un environnement cloud en utilisant la plateforme Colab. Nous avons vu comment prétraiter les images, construire et compiler le modèle et l'application de transfert de l'apprentissage. Ainsi que l'évaluation du modèle. Nous avons conclu que le modèle peut classifier nos données avec précision proche de 90%. En fin, Nous avons fait une comparaison d'exécution en utilisant trois accélérateurs offertes par le cloud (CPU/GPU/TPU) et nous concluons que l'accélérateur GPU est plus rapides ce qui prouve l'utilité de l'implémentation des modèles de deep Learning dans l'environnement Cloud. Pour les résultats, cette dernière méthode donne des Résultats proche à la réalité, mais les résultats ne sont pas toujours Excellents avec toutes les images.

Conclusion générale

Nous avons rencontré des problèmes logiciels liés à l'installation des bibliothèques python et matériels liés aux ressources demandées pour les modèles de deep Learning, ceci nous a poussé à penser au cloud et la plateforme google colab. Ces problèmes résolus par le cloud Nous a facturé un temps de travail et de réflexion qui nous a empêché de bien maîtriser le modèle de deep Learning et d'installer d'autres modèles, pour faire des comparaisons, ainsi d'implémenter une technique permettant de déterminer les régions tumeur des images saines. Nous incitons nos amis des années prochaines de terminer ce travail.

En conclusion, le traitement d'images est un domaine très vaste qui a connu, et qui connaît encore, un développement important depuis quelques dizaines d'années, nous avons essayé de creuser plus profondément dans ce domaine et nous avons voulu appliquer de nombreuses méthodes de segmentation, mais nous avons rencontré quelques obstacles. Nous espérons que les étudiants qui nous ont succédé feront plus de recherches dans ce domaine.

Bibliographie

- [1] A. Capri, "Caractérisation des objets dans une image en vue d'une aide à l'interprétation et d'une compression adaptée au contenu : application aux images échographiques", Thèse de doctorat en science à l'université D'orléans, 2007.
- [2] RC. Gonzalez, RE. Woods, "*Digital Image Processing*", Reading, MA: Addison-Wesley, 1993.
- [3] O. ASSAS, "classification floue des images", Thèse de doctorat en science à l'université de Batna, 2013.
- [4] B. Abdelmadjid, "analyse et segmentation d'images multi spectrales : application au images MSG", Thèse de doctorat en électronique à l'université Mouloude Mamri Tizi-Ouzou, 2013.
- [5] Futura sciences : magazine française, cerveau, <http://www.futurasciences.com/magazines/sante/infos/dico/d/biologie-cerveau-3125>, 2014.
- [6] L. Willmann, "Étude de l'évolution des tumeurs cérébrales en IR". Cahiers d'IRM, Masson, 1989.
- [7] H. Khotanlou, "Segmentation 3D de Tumeurs et de Structures Internes du Cerveau en IR", Thèse de Doctorat en science à L'école Nationale Supérieure de Télécommunications, 2008.
- [8] Y. Morin, "Larousse médicale", Edition Larousse, 2006.
- [9] Steven A. Goldman : "Présentation des tumeurs cérébrales ". Article 2018.
- [10] E-cancer.fr : "Tumeurs-du-cerveau". Article 2010.
- [11] KHERMAZA ELYES et BOUTIARA ABDELOUAHAB "Traitement des images IRM pour la détection des tumeurs cérébrales par les algorithmes de Deep Learning CNN, Faster RCNN, Mask R-CNN et Transfer Learning sous environnement Cloud "Mémoire de Master Université SAAD DAHLAB de BLIDA_2019/2020.
- [13] R. bentata : "segmentation d'image tomographique par émission de positon. Mémoire de magister Université " Es-Sénia "" . Oran, 2011.
- [14] A. Capelle, "Segmentation d'images IRM multi-échos tridimensionnelles pour la détection des tumeurs cérébrales par la théorie de l'évidence", Thèse de doctorat en science à L'université de Poitiers, 2002.
- [15] http://www.tumeur.tronc.cerebral.com/traitements__8_biopsie_.html
- [16] Pierre-Jean Nacher, Magnetic Resonance Imaging: From Spin Physics to Medical Diagnosis, Quantum Scape, 1-35 2008 Birkhäuser Verlag Basel, 2007.
- [17] www.cea.fr 2003, direction de la communication bâtiments siège 91191 Gif.
- [18] <http://www.cnrs.fr/imagerie-biomedicale>,
- [19] Hadjer LAGUEL, Déploiements sur une plateforme de visualisation, d'un algorithme coopératif pour la segmentation d'images IRM basé sur les systèmes multi-agents, Université des Sciences et de la Technologie Houari Boumediene.
- [20] Wigner, W. & SZEKELY, B., A review on image segmentation Techniques with remount Summing perspective. ISPRS TC VII Symposium, Vienna, Au stair, Vol. XXXVIII, Port 7A, 2010. BOULARAS ASMA, université Mohamed Boudiaf-département d'informatique-2016.
- [21] J.C. Muller, L. Abdennour, D. Dormont, L. Puybasset Unité de neuro-anesthésie-réanimation, département d'anesthésie-réanimation et département de neuroradiologie, hôpital de la Pitié-Salpêtrière, 75013 Paris, France e-mail : louis.puybasset@psl.ap-hop-paris.fr.. BOULARAS ASMA, université Mohamed Boudiaf-département d'informatique-2016.
- [22] www.sfrnet.org: Société française de radiologie. BOULARAS ASMA, université Mohamed Boudiaf-département d'informatique-2016.

- [23] [Http://www.scanner33.com/IRM_principe_et_technique.asp](http://www.scanner33.com/IRM_principe_et_technique.asp).
- [24] [Https://www.e-cancer.fr/Dictionnaire](https://www.e-cancer.fr/Dictionnaire).
- [25] BOULARAS ASMA, université Mohamed Boudiaf-département d'informatique-2016.
- [26] https://www.larousse.fr/encyclopedie/medical/substance_grise
- [27] Azouza Naouel de UNIVERSITE AKLI MOAND OULHADJE-BOUIRA (2017/2018).
- [28] J. Cocquerez and S. Philips, « Analyse d'Images : Filtrage et Segmentation », Masson,1995.
- [29] T. Pavlidis, " Why progress in machine vision is so slow", Pattern Recognition Letters, Vol.13, pp. 221-225, 1992.
- [30] H. Rami, " Contribution au développement de descripteurs LBP pour le suivi d'objet en mouvement", Thèse de doctorat en électronique à l'université Rabat, 2016.
- [31] Ch. Serief, "Extraction automatique de points d'intérêt à base de la transformée en contourlets non sous échantillonnée pour le recalage des images", Thèse de doctorat en science à l'université Mentouri Constantine, 2009.
- [32] L.G. Roberts, "Machine Perception of Three-Dimensional Solids", MIT Press: Cambridge. 159–197, 1965.
- [33] J.M.S Prewitt, «Picture Processing and Psychophysics», chapter Object Enhancement and Extraction, Academic Press: New York, pp. 75–149, 1970.
- [34]I. Sobel, "Neighbourhood coding of binary images for fast contour following and general array binary processing", Computer Graphics and Image Processing, Vol.8, pp. 127–135, 1978.
- [35] https://fr.wikipedia.org/wiki/Segmentation_d%27image?fbclid=IwAR3h3x0-R1wii0BcFFfzx2ta5EKKkINB_3dP5H15pSP8IJWYno6oC0qVBeA
- [36] S. Ouadfel, " Contributions à la Segmentation d'images basées sur la résolution collective par colonies de fourmis artificielles "Thèse de doctorat en informatique à l'université Hadj Lakhdar de Batna, 2006.
- [37] https://www.researchgate.net/publication/38320558_Croissance_de_region_variationnelle_et_contraintes_geometriques_tridimensionnelles_pour_la_segmentation_d'image?fbclid=IwAR3h3x0-R1wii0BcFFfzx2ta5EKKkINB_3dP5H15pSP8IJWYno6oC0qVBeA
- [38] C. Meurie, "Segmentation d'images couleur par classification pixellaire et hiérarchie de partitions ", Thèse de doctorat en informatique à l'université De Cean, 2005.
- [39] <https://math.univ-angers.fr/~labatte/enseignement%20UFR/master%20MIM/classificationsupervisee.pdf>
- [40]https://hal.archives-ouvertes.fr/hal-02088207/document?fbclid=IwAR0uhvjf5yp3e5A4GoDWGJrjCu4_Pk0gXQ_yephcK_Yhfq18EfVBXA1G2Z4
- [41] <http://www-lisic.univ-littoral.fr/~verel/TEACHING/08-09/sac-M1/cRdNV9.pdf> :
- [42] Kaggle: "<https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumordetection>" 2019.
- [43] : Mme. Azouza Naouel de UNIVERSITE AKLI MOAND OULHADJE-BOUIRA (2017/2018),BOULARAS Asma UNIVERSITE MOHAMED BOUDIAF - M'SILA (2015 /2016).
- [44] Aurdal. « Analysis of multi-image magnetic resonance acquisitions for segmentation and quantification of cerebral pathologies », thèse doctorat, École Nationale des Télécommunications, 1997.
- [45] Mme. Azouza Naouel de UNIVERSITE AKLI MOAND OULHADJE-BOUIRA (2017/2018),BOULARAS Asma UNIVERSITE MOHAMED BOUDIAF - M'SILA (2015 /2016).

- [46] https://pmb.univ-saida.dz/butecopac/doc_num.php?explnum_id=140
- [47] https://www.researchgate.net/publication/340593627_Une_Nouvelle_Approche_De_Classification_Automatique_Non_Supervisee_Par_CMeans_Fusion_Des_Algorithmes_flou_et_possibiliste
- [48] <https://www.m-soigner.com/pratiques/sant%C3%A9-num%C3%A9rique/747-enjeux-defis-et-apports-de-l-intelligence-artificielle-dans-la-medecine-contemporaine.html>.
- [49] <https://openclassrooms.com/fr/courses/4011851-initiez-vous-au-machine-learning/4020611-identifiez-les-differents-types-dapprentissage-automatiques>.
- [50] M.hameurlaine these doctorat “Classification collaborative multi-stratégies de données multi-sources : Application à la segmentation sub-voxelique d’Images IRM » 2017.
- [51] Belmerabet Sarra & Bardjak Nawal.Mémoire de MASTER ‘Segmentation d’image’ Université Larbi Ben M'hidi Oum El Bouaghi.2016/2017
- [52] http://staff.univbatna2.dz/sites/default/files/merzougui_ghalia/files/support_de_cours_deep_learning-chapitre3-cnn.pdf
- [53] http://staff.univbatna2.dz/sites/default/files/merzougui_ghalia/files/support_de_cours_deep_learning-chapitre3-cnn.pdf
- [54] http://staff.univbatna2.dz/sites/default/files/merzougui_ghalia/files/support_de_cours_deep_learning-chapitre3-cnn.pdf
- [55] SouvikMandal - How to use Google Colab – GeeksforGeeks - A computer science portal for geeks
- [56] <https://deeptai.org/machine-learning-glossary-and-terms/accuracy-error>
- [57] Jason Brownlee - How to Choose Loss Functions When Training Deep Learning Neural Networks – modifié le 24 Avril 2020 in Deep Learning Performance
- [58] <https://deeptai.org/machine-learning-glossary-and-terms/epoch>