

République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université de 8 Mai 1945 – Guelma -
Faculté des Mathématiques, d'Informatique et des Sciences de la matière
Département d'Informatique



Mémoire de Fin d'études Master
Filière : Informatique

Option : Science et technologie de l'information et de la communication

Thème :

**Apprentissage Profond pour identifier les cellules
cancéreuses dans des images médicales**

Encadré Par :
Ms. MADI Leila

Présenté par :
BOUGRINE Soufiane

Examineur : Dr.BORDJIBA Yamina
Président : Pr.LAFIFI Yacine

- Septembre 2024 -

Dédicace

À celui qui a été avec moi à chaque instant, celui que j'ai invoqué et qui a répondu, en qui j'ai eu confiance et qui m'a honoré, à mon Créateur, mon Seigneur, je te remercie pour toutes les bénédictions, pour la réalisation et l'achèvement. Merci, Allah.

À celui qui m'a enseigné comment je peux atteindre ce que je veux et que je peux réaliser tout ce à quoi j'aspire, à celui qui a cru en moi dans les moments de désespoir et a dit, "Tu peux le faire." À celui dont je porte le nom et j'espère le rendre fier, je dédie ceci à mon cher père.

À celle qui m'a enseigné que tout a une racine et que j'ai un Seigneur qui connaît mes intentions. À celle dont les prières m'ont accompagné à chaque pas et ont ouvert toutes les voies pour moi, à ma mère.

À ma belle famille, mes charmantes sœurs, et mon cher frère, mon soutien dans la vie. Aux amis qui ont partagé les moments et les jours, aux camarades de classe, à tous ceux qui m'ont aidé à un moment donné, et à tous ceux qui ont dit que je ne pouvais pas et sont devenus une grande motivation pour moi de continuer, de ne pas abandonner mes ambitions.

Je suis ici aujourd'hui, dédiant ma remise de diplôme à chacun d'entre
VOUS.

BOUGRINE Soufiane

Remerciements

In the Name of Allah, the Most Merciful, the Most Compassionate, all praise be to Allah, the Lord of the worlds ; and prayers and peace be upon Muhammad, His servant, and messenger. First and foremost, we express our utmost gratitude to Allah, the Ever-Magnificent.

Je tiens à exprimer ma sincère gratitude au **Dr. MADI Leila** pour sa direction éclairée, son encouragement constant et son dévouement à l'excellence académique.

Ses précieux conseils et son soutien ont grandement enrichi mon expérience de recherche et ont été essentiels à la réalisation de cette thèse.

De plus, je tiens à exprimer ma sincère gratitude à tous les participants de notre étude. Leur volonté de partager leurs expériences et leurs perspectives a été inestimable, contribuant de manière significative au succès de cette thèse. Je leur suis vraiment reconnaissant pour le temps qu'ils ont accordé et pour leurs contributions.

Encore une fois, je souhaite exprimer ma plus profonde gratitude à tous ceux qui m'ont soutenu tout au long de ce voyage, sans qui cette réalisation n'aurait pas été possible. Que Allah vous bénisse tous abondamment.

Abstract

Brain tumors are a complex and dangerous form of cancer, where early diagnosis is crucial to improving patients' chances of survival. Traditionally, doctors have relied on clinical examinations and medical imaging techniques such as MRI. However, these methods, although effective, remain subjective and dependent on human expertise, which can lead to variations in diagnoses.

The use of computer-aided diagnostic systems, notably via deep learning techniques, makes the process more objective, reproducible and accurate. These systems are based on convolutional neural networks (CNNs) capable of detecting complex patterns in medical images.

In this work, we proposed a brain tumor classification system ,we exploit transfer learning, where the ResNet50 model was used as a base model reinforced by the Convolutional Block Attention Module (CBAM) . This module enables the model to focus on the most relevant regions of the images, thus improving tumor detection. We also applied data pretreatment techniques to improve model robustness.

Keywords :Brain tumors , Deep learning , Convolutional neural networks (CNN), Transfer learning, ResNet50, CBAM attention module, Magnetic resonance imaging (MRI).

Résumé

Les tumeurs cérébrales sont une forme complexe et dangereuse de cancer, où un diagnostic précoce est crucial pour améliorer les chances de survie des patients. Traditionnellement, les médecins se basent sur des examens cliniques, des techniques d'imagerie médicale telles que l'IRM. Cependant, ces méthodes, bien que efficaces, restent subjectives et dépendantes de l'expertise humaine, ce qui peut entraîner des variations dans les diagnostics. L'utilisation des systèmes de diagnostic assistés par ordinateur, notamment via des techniques d'apprentissage profond, permet de rendre le processus plus objectif, reproductible et précis. Ces systèmes se basent sur des réseaux de neurones convolutifs (CNN) capables de détecter des motifs complexes dans les images médicales.

Dans ce travail, nous avons proposé un système de classification des tumeurs cérébrales, nous exploitons l'apprentissage par transfert, où le modèle ResNet50 a été utilisé comme modèle de base renforcé par le module d'attention CBAM (Convolutional Block Attention Module). Ce module permet au modèle de se concentrer sur les régions les plus pertinentes des images, améliorant ainsi la détection des tumeurs. Nous avons également appliqué des techniques de prétraitement des données pour améliorer la robustesse du modèle.

Mots-clés : Tumeurs cérébrales, Deep learning, Réseaux de neurones convolutifs (CNN), Apprentissage par transfert, ResNet50, Module d'attention CBAM, Imagerie par résonance magnétique (IRM).

ملخص

تُعد أورام الدماغ شكلاً معقداً وخطيراً من أشكال السرطان، حيث يُعد التشخيص المبكر أمراً بالغ الأهمية لتحسين فرص نجاة المرضى. تقليدياً، اعتمد الأطباء على الفحوصات السريرية وتقنيات التصوير الطبي مثل التصوير بالرنين المغناطيسي. ومع ذلك، تظل هذه الأساليب، على الرغم من فعاليتها، غير موضوعية وتعتمد على الخبرة البشرية، مما قد يؤدي إلى اختلافات في التشخيص. إن استخدام أنظمة التشخيص بمساعدة الحاسوب، لا سيما عبر تقنيات التعلم العميق، يجعل العملية أكثر موضوعية وقابلة للتكرار والدقة. تعتمد هذه الأنظمة على الشبكات العصبية التلافيفية (CNNs) القادرة على اكتشاف الأنماط المعقدة في الصور الطبية.

اقترحنا في هذا العمل نظاماً لتصنيف أورام الدماغ، نستغل فيه التعلم التحويلي، حيث تم استخدام نموذج ResNet50 كنموذج أساسي معزز بوحدة الانتباه الكتلي التلافيفي (CBAM). تمكن هذه الوحدة النموذج من التركيز على المناطق الأكثر صلة في الصور، وبالتالي تحسين اكتشاف الورم. طبقنا أيضاً تقنيات المعالجة المسبقة للبيانات لتحسين متانة النموذج.

الكلمات الرئيسية: أورام الدماغ، التعلم العميق، الشبكات العصبية التلافيفية (CNN)، التعلم التحويلي، ResNet50، وحدة الانتباه CBAM، التصوير بالرنين المغناطيسي (MRI).

Table des matières

Introduction Générale	6
1 L’Imagerie Médicale et Détection d’Objet	9
Introduction	9
2 L’imagerie médicale	9
2.1 Techniques d’Imagerie Médicale	10
2.2 Limitations de l’imagerie médicale traditionnelle	12
3 Les Tumeurs Cérébrales et Les Cellules Cancéreuses	13
3.2 Les Cellules Cancéreuses	13
3.3 Types de tumeurs	14
4 La détection d’objet dans l’imagerie médicale	15
4.1 La détection d’objet	16
4.2 Localisation et classification d’objets d’intérêt spécifiques dans une image	16
4.3 Approches de détection d’objet	16
5 Intelligence Artificielle et Imagerie Médicale	18
6 Aperçu sur Datasets	18
6.1 Aperçu sur Datasets	18
6.2 OASIS (Open Access Series of Imaging Studies)	19
6.3 TCGA-GBM (The Cancer Genome Atlas - Glioblastoma Multiforme)	19
6.4 TCGA-LGG (The Cancer Genome Atlas - Lower Grade Glioma)	19
6.5 Le REMBRANDT (Repository of Molecular Brain Neoplasia Data)	20
Conclusion	20
2 Apprentissage Profond	22
Introduction	22
2 Apprentissage Profond	23
3 Réseaux neuronaux artificiels (RNA)	23
3.1 Structure d’un RNA	24
3.2 Fonctionnement de base d’un RNA	25
4 Réseaux de neurones convolutifs (CNN)	26
4.1 Les Couches de CNN	26
4.2 Quelques architectures de base de CNN	28
5 L’apprentissage par transfert	29
5.1 ResNet50	30
5.2 Types de connexions résiduelles	31

6 Les travaux Conex	32
6.1 Approches hybrides pour la classification des tumeurs cérébrales à l'aide de données d'IRM	32
6.2 Approches d'apprentissage profond pour la classification des tumeurs cérébrales à l'aide de données d'IRM	36
Conclusion	37
3 Conception	39
Introduction	39
2 Objectif	40
3 Architecture de système	40
3.1 La phase prétraitement des données	40
3.2 Phase de Construction et d'Entraînement du Modèle	43
3.3 Architecture de Model	47
Conclusion	51
4 Implementation et Realisation	53
Introduction	53
2 Environnement de Développement	53
2.1 Outils et Langages Utilisés	53
2.2 Configuration Matérielle	55
2.3 Environnements Logiciels	55
2.4 Outils d'évaluation des performances	56
3 Collecte de données	57
4 Implementation	59
5 Entraînement et Évaluation des Performances	63
5.1 Entraînement	63
5.2 Résultats et discussions	65
6 L'interface du système	71
7 Discussion finale	73
Conclusion et Perspectives	74
Conclusion Générale	75

Table des figures

1.1	La radiologie	10
1.2	Le scannere	10
1.3	L'échographie	11
1.4	L'IRM	11
1.5	Le scannere	12
1.6	La scintigraphie	12
1.7	La Détection d'Objets dans l'Imagerie Médicale	15
2.1	Apprentissage Profond	23
2.2	Structure Réseaux de neurones artificiels	25
2.3	Réseaux de neurones artificiels	25
2.4	Couche Convolutive (CONV)	27
2.5	Couche POOL	27
2.6	Les Couches de CNN	28
2.7	Processus d'apprentissage par transfert	30
2.8	Architecture de ResNet50	31
3.1	Image original	41
3.2	Image Cropé	41
3.3	Image avec CLAHE	41
3.4	Image original	42
3.5	Image après le Rotation	42
3.6	Image original	43
3.7	Image après decolage de la largeur	43
3.8	Image original	43
3.9	Image après un flip horizontal et vertical	43
3.10	L'architecture de bloc CBAM	45
3.11	Architecture du modèle pré-entraîné ResNet50 utilisé dans notre système.	47
3.12	l'architecture général de système	48
4.1	Nombre d'Échantillons par Classe dans l'Ensemble d'Entraînement	58
4.2	Nombre d'Échantillons par Classe dans l'Ensemble de test	59
4.3	Photo encadrée en utilisant les coordonnées extrêmes	60
4.4	Nombre d'Échantillons par Classe dans l'Ensemble d'Entraînement	60
4.5	Application de CLAHE sur les images	61

4.6	Le redimensionnement et les étiquetages et la normalisation des données .	62
4.7	Préparation des Données d'Entraînement et de Test pour le Modèle , One-Hot Encoding	63
4.8	Génération et Augmentation des Données d'Entraînement avec ImageData-Generator	63
4.9	Difinition des callbacks	64
4.10	Compilation du Modèle	64
4.11	Entraînement du Modèle sur 25 Époques Surveillance par Callbacks	64
4.12	graphique de accuracy et loss de l'entraînement et de la validation	65
4.13	la matrice de confusion de dataset de test	66
4.14	le graph ROC de dataset de test	66
4.15	rapport de classification	67
4.16	graphique de accuracy et loss de l'entraînement et de la validation	68
4.17	la matrice de confusion	68
4.18	le graph ROC	69
4.19	rapport de classification	69
4.20	page index	71
4.21	select image	71
4.22	Result Glioma	72
4.23	result NO Tumeur	72
4.24	result Meningioma	73

Liste des tableaux

2.1	Architectures de réseaux de neurones et leurs descriptions	28
2.2	Résumé pour Approches hybrides pour la classification des tumeurs cérébrales.	35
2.3	Approches d'apprentissage profond pour la classification des tumeurs cérébrales à l'aide de données d'IRM.	37
4.1	la distribution des images	58
4.2	Paramètres du Modèle	65
4.3	Métriques de Performance du Modèle	67
4.4	Résumé des métriques d'évaluation du modèle	69
4.5	Comparaison des performances des expérimentations avec et sans CBAM .	70

Introduction Générale

Le cancer est une maladie répandue et potentiellement mortelle qui touche des millions de personnes dans le monde. Le cancer est une maladie complexe caractérisée par la croissance anarchique de cellules anormales qui échappent aux mécanismes normaux de régulation de la croissance cellulaire. Ces cellules peuvent former des masses appelées tumeurs qui peuvent être bénignes (non cancéreuses) ou malignes (cancéreuses) .

Le cancer englobe un large éventail de maladies, chacune avec ses propres caractéristiques et comportements biologiques. Parmi les types de cancer les plus courants, on trouve le cancer du sein, le cancer du poumon, le cancer du cerveau. La détection précoce et le diagnostic des anomalies jouent un rôle important dans l'amélioration de l'état des patientes et l'augmentation des taux de survie. Le diagnostic du cancer repose souvent sur une combinaison de techniques d'imagerie médicale, de tests de laboratoire, de biopsies et d'examens cliniques. L'imagerie médicale, en particulier, joue un rôle crucial dans la détection précoce et la surveillance de la progression de la maladie. Des technologies avancées telles que l'imagerie par résonance magnétique (IRM), la tomographie par émission de positons (TEP) et la mammographie permettent de visualiser les structures internes du corps et d'identifier les anomalies suspectes. L'IRM, en particulier, fournit des images détaillées des structures internes du corps sans l'utilisation de rayonnements ionisants, ce qui en fait un outil précieux pour l'évaluation des tumeurs et des tissus anormaux.

Contexte et Problématique

Les techniques d'apprentissage automatique constituent un outil puissant pour le développement de classificateurs performants. En s'appuyant sur des ensembles de données d'apprentissage étiquetés, ces techniques permettent aux algorithmes d'apprendre des modèles et des relations complexes au sein des données. Cette connaissance acquise est ensuite exploitée pour effectuer des prédictions ou des classifications sur de nouveaux échantillons, même ceux qui n'ont jamais été vus auparavant.

Dans ce contexte, le Deep Learning, une branche de l'intelligence artificielle (IA), a émergé comme une solution prometteuse pour automatiser et améliorer la détection des cellules cancéreuses dans les images médicales. Le Deep Learning, en particulier les réseaux de neurones convolutifs (Convolutional Neural Networks - CNN), a démontré une capacité remarquable à extraire des caractéristiques complexes et à classifier des images avec une précision comparable, voire supérieure, à celle des experts humains. Les modèles de Deep Learning peuvent identifier des anomalies subtiles qui pourraient être difficiles à détecter à l'œil nu.

Objectifs et Contributions

L'objectif principal de notre travail est de concevoir un modèle avec une architecture d'apprentissage en profondeur, et d'implémenter un modèle de réseau de neurones convolutionnel pour la classification de cancer du cerveau en utilisant l'apprentissage par transfert . La mise en œuvre des modèles de CNN implique plusieurs étapes, incluant le prétraitement des images. L'évaluation de ces modèles est réalisée à l'aide de métriques comme l'accuracy, la le score F, le rappel (recall).

Chapiter 1

L'Imagerie Médicale et Détection d'Objet

L’Imagerie Médicale et Détection d’Objet

Introduction

L’imagerie médicale, avec ses avancées impressionnantes, se profile comme un outil indispensable dans la détection précoce et la caractérisation des tumeurs cérébrales ainsi que des cellules cancéreuses. En intégrant des technologies de pointe telles que l’intelligence artificielle, notamment dans le domaine de l’apprentissage automatique, l’imagerie médicale évolue vers une précision et une efficacité sans précédent. Cette convergence de la médecine et de la technologie promet non seulement d’améliorer les diagnostics, mais aussi d’ouvrir de nouvelles voies pour des traitements plus ciblés et personnalisés. Dans cette introduction, nous explorerons le rôle crucial de l’imagerie médicale dans la lutte contre les maladies du cerveau, en mettant en lumière les possibilités révolutionnaires offertes par l’intégration de l’intelligence artificielle dans ce domaine vital de la santé.

2 L’imagerie médicale

L’imagerie médicale est une branche de la médecine qui utilise diverses techniques d’imagerie pour visualiser l’intérieur du corps humain à des fins de diagnostic, de suivi et de traitement des maladies. Ces techniques permettent de créer des images détaillées des organes, des tissus, des vaisseaux sanguins et des structures anatomiques, fournissant ainsi aux professionnels de la santé des informations précieuses sur l’état de santé d’un patient. Les modalités d’imagerie médicale les plus couramment utilisées comprennent la radiographie, la tomographie par ordinateur (CT), l’imagerie par résonance magnétique (IRM), l’échographie, la mammographie, la scintigraphie et la tomographie par émission de positons (PET) [11]. L’imagerie médicale joue un rôle essentiel dans le diagnostic précoce des maladies, la planification des traitements, le suivi de l’évolution des pathologies, et la recherche médicale [54].

2.1 Techniques d'Imagerie Médicale

- **La radiologie** : La radiographie [18] utilise les rayons X, qui traversent les tissus avec une absorption variable selon leur densité. Une source de rayons X est placée devant le corps et un détecteur à l'arrière. Les photons traversent le corps et sont différemment absorbés par les tissus, permettant ainsi de distinguer les os des muscles sur l'image obtenue.

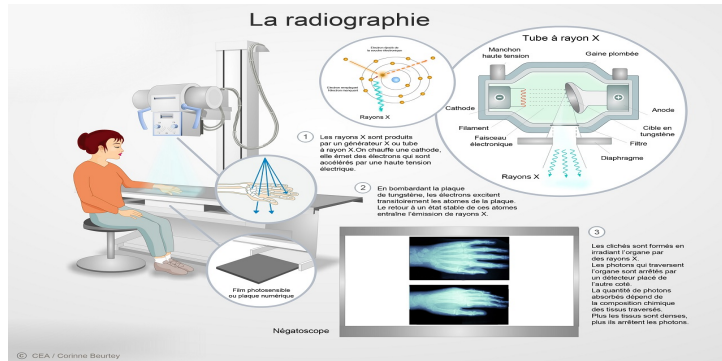


FIGURE 1.1 – La radiologie [17]

- **Le scanner (tomodensitométrie)** : Le scanner, ou tomodensitométrie [69], utilise des rayons X pour produire des images détaillées du corps en coupe transversale. Un faisceau de rayons X tourne autour du patient, tandis qu'un détecteur capte les rayons traversant les tissus. Ces données sont ensuite traitées par un ordinateur pour créer des images précises des structures internes, permettant de distinguer clairement les différentes densités des tissus.

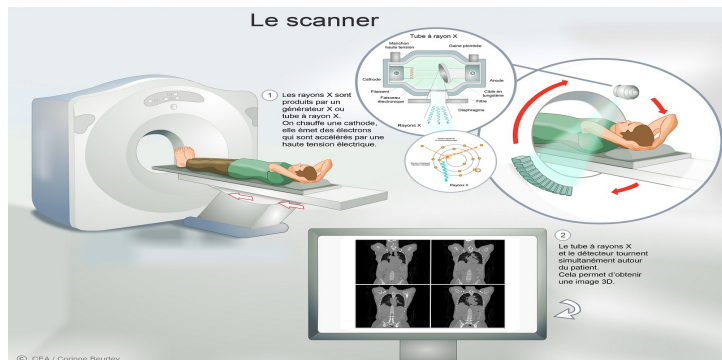


FIGURE 1.2 – Le scanner [17]

- **L'échographie** : L'échographie [53] utilise des ondes sonores à haute fréquence pour produire des images des structures internes du corps. Un transducteur émet des ondes sonores qui traversent les tissus et sont réfléchies en fonction de leur densité. Ces échos sont captés par le transducteur et convertis en images en temps réel par un ordinateur, permettant ainsi de visualiser des organes, des tissus mous et des flux sanguins.

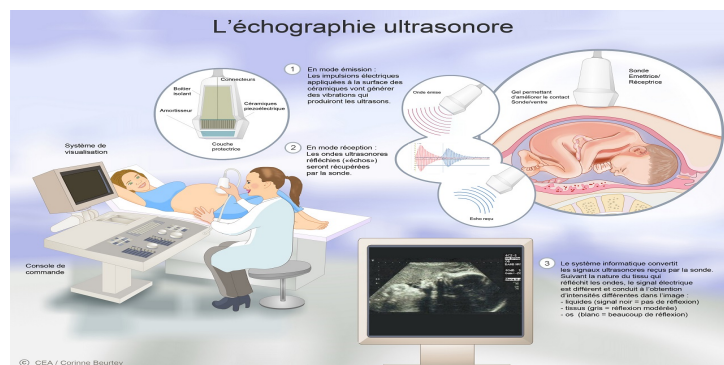


FIGURE 1.3 – L'échographie[17]

- **L'IRM** : (imagerie par résonance magnétique) [47] utilise un champ magnétique puissant et des ondes radio pour produire des images détaillées des structures internes du corps. Le champ magnétique aligne les protons des atomes d'hydrogène dans le corps, et les ondes radio les excitent. Lorsque les protons reviennent à leur état initial, ils émettent des signaux qui sont captés et transformés en images par un ordinateur. L'IRM est particulièrement utile pour visualiser les tissus mous, comme le cerveau, les muscles et les ligaments.



FIGURE 1.4 – L'IRM[17]

- **La spectroscopie par résonance magnétique (SRM)** La spectroscopie par résonance magnétique (SRM) [26] est une méthode d'analyse non invasive qui utilise des champs magnétiques et des ondes radio pour examiner les molécules dans le corps humain. En détectant les signaux émis par les noyaux atomiques soumis à ces champs, la SRM génère des images détaillées des tissus corporels et permet d'identifier différentes substances, facilitant ainsi le diagnostic médical.

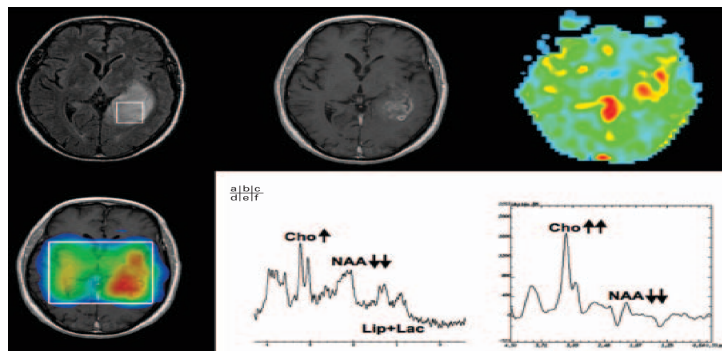


FIGURE 1.5 – Le scanner [27]

- **La scintigraphie** : La scintigraphie [82], pratique médicale en médecine nucléaire, requiert une injection légèrement radioactive par voie intraveineuse. Elle explore les organes et les os afin de repérer d'éventuelles anomalies fonctionnelles ou structurales. Le produit radioactif se fixe sur la zone d'intérêt, émettant des rayonnements gamma captés par une caméra pour produire une image scintillante sur un écran d'ordinateur.

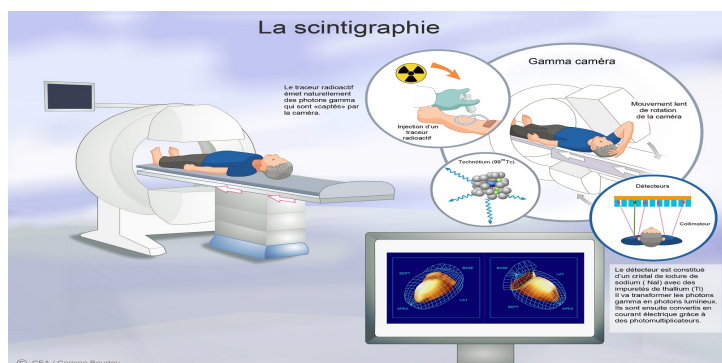


FIGURE 1.6 – La scintigraphie [17]

2.2 Limitations de l'imagerie médicale traditionnelle

Certains processus d'imagerie traditionnels peuvent avoir une précision limitée, ce qui rend difficile la visualisation de petites structures ou de détails fins souvent et il peut également être difficile de distinguer les différentes structures anatomiques, Il a constaté une grande difficulté dans l'imagerie des tissus mous tels que les muscles, les tendons et les organes internes, mais l'exposition aux rayonnements pendant la radiographie et les tomodensitogrammes expose les patients à des doses nocives de rayonnement, ce qui entraîne des complications. De nombreuses méthodes d'imagerie traditionnelles se concentrent uniquement sur la structure anatomique et ne fournissent pas suffisamment d'informations sur les fonctions des tissus ou des organes, mais sont très coûteuses[52], telles que l'IRM et la TEP, coûteuses à effectuer et nécessitent un équipement spécialisé et un personnel formé.

3 Les Tumeurs Cérébrales et Les Cellules Cancéreuses

Les tumeurs cérébrales sont des masses ou des croissances anormales de cellules qui se développent dans le cerveau ou le système nerveux central. Elles peuvent être bénignes, se caractérisant par une croissance lente et bien délimitée, ou malignes, présentant une croissance rapide et invasive [65]. Les tumeurs bénignes, bien que non cancéreuses, peuvent encore causer des problèmes graves en raison de leur emplacement. Les tumeurs malignes, ou cancers du cerveau, sont plus agressives, envahissant les tissus environnants et pouvant se propager à d'autres parties du corps [86].

Les tumeurs cérébrales et les cellules cancéreuses sont toutes deux dangereuses en raison de leur capacité à perturber les fonctions normales du corps humain. Les tumeurs cérébrales, en particulier, peuvent affecter des fonctions critiques telles que le mouvement, la sensation, et les capacités cognitives. La détection précoce et le traitement des tumeurs cérébrales et des cellules cancéreuses sont cruciaux pour améliorer les chances de survie et la qualité de vie des patients [8]. Les avancées technologiques en imagerie médicale et en biologie moléculaire ont permis des progrès significatifs dans le diagnostic et le traitement de ces maladies, offrant ainsi de nouvelles perspectives pour les patients atteints de tumeurs cérébrales et de cancers.

3.1 Les Tumeurs Cérébrales

Les causes des tumeurs peuvent être variées et incluent des facteurs génétiques, environnementaux, et des infections virales. Parmi les facteurs de risque connus :

- Génétique : Prédispositions héréditaires à certains types de cancers [45].
- Exposition aux Toxines : Tabagisme, radiation, produits chimiques industriels [85].
- Infections Virales : Certains virus comme le HPV (virus du papillome humain) peuvent causer des tumeurs [21].
- Mode de Vie : Mauvaise alimentation, manque d'activité physique, consommation excessive d'alcool [61].
- Âge : Le risque de développer des tumeurs augmente avec l'âge [28].

Le diagnostic des tumeurs repose sur diverses méthodes, y compris l'imagerie médicale (radiographie, tomodensitométrie, IRM, échographie), les biopsies (prélèvement et analyse de tissus) et les tests sanguins pour la recherche de marqueurs tumoraux spécifiques [76].

3.2 Les Cellules Cancéreuses

Les cellules cancéreuses sont des cellules anormales qui se divisent et croissent de manière incontrôlée, échappant aux mécanismes de régulation de la division cellulaire normale. Contrairement aux cellules saines, les cellules cancéreuses peuvent éviter l'apoptose (la mort cellulaire programmée), se multiplier de manière excessive et envahir les tissus voisins [38]. Elles possèdent la capacité de pénétrer dans le sang ou le système lymphatique, se disséminant ainsi à d'autres parties du corps pour former de nouvelles tumeurs, un processus connu sous le nom de métastase [35]. Les cellules cancéreuses résultent souvent de mutations génétiques qui activent des oncogènes (gènes favorisant la croissance tumorale) ou désactivent des gènes suppresseurs de tumeurs (gènes empêchant la croissance tumorale) [83]. Ces altérations génétiques leur confèrent des caractéristiques distinctes telles qu'une

prolifération rapide, une insensibilité aux signaux de croissance et de suppression de croissance, une capacité d'angiogenèse (formation de nouveaux vaisseaux sanguins pour nourrir la tumeur) et une potentialité métastatique, rendant le cancer une maladie complexe et difficile à traiter [37].

3.3 Types de tumeurs

Les tumeurs cérébrales sont des masses anormales de cellules dans le cerveau [66], tandis que les cellules cancéreuses sont des cellules anormales qui se divisent de manière incontrôlée [38]. Les tumeurs cérébrales et les cellules cancéreuses peuvent être de même type, impliquant des cellules malignes [57].

3.3.1 Selon la Nature (Bénignes vs Malignes)

1. **Tumeurs bénignes**
 - Non cancéreuses
 - Croissance lente
 - Bien délimitées et souvent encapsulées
 - Ne se propagent pas (pas de métastases)
2. **Tumeurs malignes**
 - Cancéreuses
 - Croissance rapide et incontrôlée
 - Bien délimitées et souvent encapsulées
 - Mal délimitées et non encapsulées
 - Invasives et peuvent se propager (métastases)
3. **Tumeurs précancéreuses**
 - Contiennent des cellules anormales qui pourraient devenir cancéreuses
 - Pas encore invasives

3.3.2 Selon l'Origine Cellulaire

1. **Carcinomes** : *l'origine* est les cellules épithéliales et les types spécifiques :
 - *Carcinome basocellulaire* : Peau
 - *Carcinome épidermoïde* : Peau et muqueuses
 - *Adénocarcinome* : Glandes et tissus glandulaires
2. **Sarcomes** *l'origine* est les tissus conjonctifset les types spécifiques :
 - *Ostéosarcome* : Os
 - *Liposarcome* : Graisse
 - *Rhabdomyosarcome* : Glandes et tissus glandulaires
3. **Lymphomes et leucémies** *l'origine* est le Système lymphatique et cellules sanguines les types spécifiques :
 - *Lymphome de Hodgkin* : Système lymphatique
 - *Lymphome non hodgkinien* : Système lymphatique
 - *Leucémie lymphoïde* : Cellules sanguines

4. **Gliomes** *l'origine* est les Cellules gliales du système nerveux central les types spécifiques :
 - *Glioblastome* : Cerveau
 - *Astrocytome* : Cerveau et moelle épinière
 - *Oligodendrogliome* : Cerveau

3.3.3 Selon la Localisation

1. **Tumeurs cérébrales** comme les gliomes et les méningiomes, proviennent des tissus du cerveau ou du système nerveux central.
2. **Tumeurs osseuses** telles que l'ostéosarcome et le chondrosarcome, ont pour origine les tissus osseux.
3. **Tumeurs des tissus mous** comme le liposarcome et le fibrosarcome, proviennent des tissus conjonctifs et musculaires.
4. **Tumeurs des organes solides** comme l'adénocarcinome (poumon, sein, prostate), l'hépatome (foie), et le néphroblastome (rein), proviennent des organes internes.

4 La détection d'objet dans l'imagerie médicale

L'évolution de la détection d'objets a été marquée par une transition de méthodes traditionnelles basées sur le traitement d'images vers des approches modernes utilisant l'apprentissage profond. Cette transition a permis des avancées significatives en termes de précision et de généralisation. Les réseaux de neurones convolutifs ont joué un rôle central dans cette évolution, permettant une détection plus rapide et plus précise des objets dans les images.

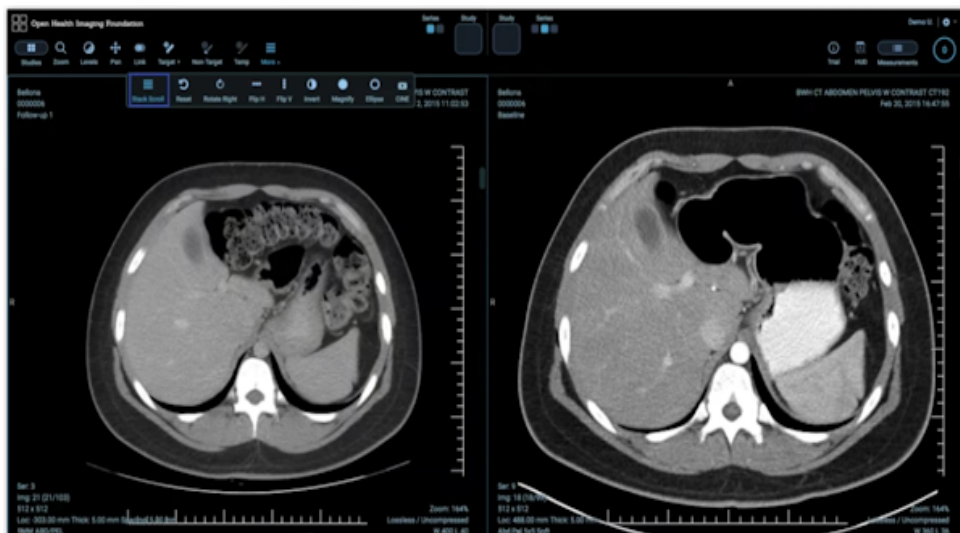


FIGURE 1.7 – La Détection d'Objets dans l'Imagerie Médicale[32]

4.1 La détection d'objet

La détection d'objets consiste à localiser et à classer des objets dans des images ou des vidéos [90]. Elle vise à identifier des objets spécifiques et à fournir leurs délimitations, ce qui est essentiel pour des tâches telles que le suivi d'objets et la compréhension de scènes [55]. Cette technologie est utilisée dans divers domaines, notamment la vidéosurveillance pour identifier et suivre des individus ou des objets [73], l'agriculture pour surveiller les cultures et détecter les parasites [46], le commerce de détail pour analyser les comportements [56] et les images médicales pour détecter des anomalies [54]. La détection d'objets repose sur des éléments clés tels que les réseaux de propositions régionales (RPN) pour générer des propositions d'objets potentiels, les réseaux d'extraction de caractéristiques pour analyser les représentations et les classificateurs pour identifier les catégories d'objets [75]. Les algorithmes de détection d'objets varient en termes de vitesse, de précision et de résolution pour répondre aux exigences spécifiques des différentes applications [88].

4.2 Localisation et classification d'objets d'intérêt spécifiques dans une image

Dans le contexte de l'imagerie médicale, la localisation d'objets se réfère à la capacité d'identifier et de délimiter avec précision les objets d'intérêt dans une image [54]. Cette étape est cruciale car elle fournit des informations sur la position exacte et l'étendue spatiale des structures ou des anomalies dans l'image [89]. Pour ce faire, des techniques de segmentation et de délimitation sont souvent utilisées, telles que les boîtes englobantes ou les masques sémantiques [77]. Par exemple, dans le cas de la détection de tumeurs, la localisation précise de la tumeur dans une image radiographique ou IRM peut fournir des informations importantes pour le diagnostic et la planification du traitement [24].

D'autre part, la classification consiste à attribuer des étiquettes de catégorie à chaque objet détecté. Dans le domaine de l'imagerie médicale, cela signifie souvent différencier les structures anatomiques normales des anomalies ou des pathologies [79]. Par exemple, dans une image de tomographie assistée par ordinateur (CT) du cerveau, la classification peut être utilisée pour identifier et étiqueter automatiquement les différentes régions cérébrales, telles que le cortex, les ventricules, ou les structures vasculaires, ce qui facilite l'analyse quantitative et la génération de rapports médicaux [16]. En combinant la localisation et la classification, la détection d'objets en imagerie médicale permet aux cliniciens d'identifier et de caractériser les structures anatomiques pertinentes ou les anomalies pathologiques avec une précision accrue. Cela peut contribuer à des diagnostics plus rapides et plus précis, à une planification de traitement personnalisée et à un suivi de l'évolution de la maladie [84].

4.3 Approches de détection d'objet

Dans le domaine de la détection d'objets, il existe deux approches principales : les méthodes traditionnelles basées sur les caractéristiques et les méthodes modernes basées sur l'apprentissage profond. .

4.3.1 Méthodes basées sur les caractéristiques (features)

Ces méthodes se concentrent sur l’identification de caractéristiques discriminantes, comme les contours, les textures, et les formes des objets. Une fois ces caractéristiques extraites, elles sont ensuite classifiées à l’aide de modèles statistiques ou d’algorithmes de machine learning.

- **SIFT (Scale-Invariant Feature Transform)**

SIFT est une méthode permettant de détecter des points d’intérêt dans une image, peu importe l’échelle ou l’angle de vue. L’idée est de trouver des points caractéristiques dans l’image qui sont robustes aux changements d’échelle et de rotation. Ces points sont ensuite utilisés pour correspondre à des objets dans d’autres images [58].

- **HOG (Histogram of Oriented Gradients)**

L’une des méthodes les plus célèbres pour la détection d’objets, en particulier pour la détection de piétons. HOG divise une image en petites cellules et, pour chaque cellule, un histogramme des orientations des gradients est calculé. Ces histogrammes sont ensuite combinés pour former un vecteur de caractéristiques décrivant l’image. Ce vecteur est ensuite utilisé comme entrée pour un classificateur comme une SVM (Support Vector Machine)[19].

- **SURF (Speeded Up Robust Features)**

SURF est une amélioration de SIFT qui se concentre sur la rapidité d’exécution tout en conservant la robustesse de la détection de caractéristiques d’image. Il utilise un filtre approximatif basé sur les boîtes pour remplacer les convolutions complexes de SIFT.[6]

4.3.2 Méthodes basées sur l’apprentissage profond (modernes)

Les méthodes de détection d’objets basées sur l’apprentissage profond ont révolutionné la vision par ordinateur en permettant des détections plus précises et rapides sans nécessiter d’extraction manuelle des caractéristiques. Des modèles comme R-CNN et ses améliorations, Fast R-CNN et Faster R-CNN, utilisent des réseaux de neurones convolutifs (CNN) pour extraire des caractéristiques d’image et générer des propositions de régions, améliorant ainsi la rapidité et l’efficacité des détections [29] [75]. D’autres modèles, comme YOLO et SSD, ont introduit une approche unifiée qui permet une détection en temps réel en prédisant directement les boîtes englobantes et les classes d’objets sur l’ensemble de l’image[74]. Récemment, des architectures basées sur les transformers, comme DETR, ont émergé, modélisant les relations spatiales entre objets et simplifiant le pipeline de détection . Ces approches modernes permettent des performances inégalées dans des applications variées comme la surveillance et la conduite autonome[12].

La localisation et la classification d’objets spécifiques dans une image sont des tâches cruciales en vision par ordinateur, et les méthodes basées sur l’apprentissage profond moderne ont révolutionné la manière dont ces tâches sont abordées [50]. Ces méthodes exploitent la puissance des réseaux de neurones profonds pour extraire des informations riches et significatives des images, permettant ainsi une détection précise et efficace des objets d’intérêt [30]. Dans cette introduction, nous explorerons quelques-unes des approches les plus innovantes et les plus efficaces dans ce domaine, qui ont transformé notre capacité à analyser et à comprendre le contenu visuel de manière automatisée [75].

5 Intelligence Artificielle et Imagerie Médicale

Dans le domaine de l’imagerie médicale, l’intelligence artificielle (IA) joue un rôle essentiel en proposant des solutions innovantes pour analyser et interpréter les données d’imagerie [54]. Elle repose sur des algorithmes sophistiqués capables d’extraire des informations précieuses à partir d’images radiologiques, d’IRM, de scanners CT et d’autres modalités d’imagerie. Ces systèmes exploitent des techniques avancées telles que l’apprentissage automatique et le Deep Learning pour détecter les anomalies, segmenter les structures anatomiques et assister les radiologues et les cliniciens dans leurs diagnostics [33].

Les applications de l’IA en imagerie médicale sont nombreuses, allant de la détection précoce des maladies à l’évaluation de la progression des pathologies, en passant par l’aide à la planification chirurgicale et au suivi des réponses aux traitements [79]. Par exemple, pour la détection de tumeurs, les systèmes d’IA peuvent automatiquement repérer les masses anormales dans les images radiographiques, permettant ainsi un diagnostic rapide et une intervention précoce. De même, en segmentant les organes et les tissus, les systèmes d’IA aident à quantifier les changements anatomiques et à évaluer l’évolution des maladies au fil du temps [54].

Bien que l’utilisation de l’IA en imagerie médicale offre de nombreux avantages, comme une analyse rapide et précise des images et une réduction de la charge de travail des radiologues, elle présente également des défis, notamment la nécessité de données d’entraînement de haute qualité et la validation clinique des systèmes avant leur déploiement clinique à grande échelle [33].

D’autre part, le Deep Learning en imagerie médicale représente une approche spécifique de l’intelligence artificielle, utilisant des réseaux de neurones profonds pour analyser et interpréter les images médicales [54]. Ces réseaux, tels que les réseaux de neurones convolutionnels (CNN), sont largement utilisés pour des tâches telles que la détection d’anomalies, la segmentation d’organes et la classification de pathologies [79].

Le Deep Learning en imagerie médicale offre plusieurs avantages par rapport aux approches traditionnelles, notamment une capacité à apprendre automatiquement les caractéristiques des données, une flexibilité pour traiter des données complexes et une performance supérieure dans de nombreuses tâches d’analyse d’images. Cependant, son utilisation nécessite une expertise considérable en apprentissage automatique et en analyse d’images, ainsi que des ressources informatiques importantes pour l’entraînement et le déploiement des modèles. De plus, il est crucial de garantir la fiabilité et la robustesse des modèles, surtout dans des environnements cliniques où des erreurs pourraient avoir des conséquences graves sur les patients [33].

6 Aperçu sur Datasets

6.1 Aperçu sur Datasets

Le **dataset BRATS (Brain Tumor Segmentation Challenge)** [64] est une référence dans la recherche sur l’analyse et la segmentation des tumeurs cérébrales à partir d’images IRM. Il est largement utilisé pour développer et évaluer des algorithmes d’intelli-

gence artificielle dans la détection et la segmentation des tumeurs cérébrales, en particulier les gliomes. Le dataset fournit des images IRM multimodales (T1, T1c, T2, FLAIR) accompagnées de segmentations manuelles réalisées par des experts médicaux. Les annotations incluent la segmentation des trois principales régions tumorales :

- **Le noyau tumoral (core)** : région centrale de la tumeur.
- **L’œdème pérítumoral** : zone d’inflammation autour de la tumeur.
- **La partie rehaussée** : régions de la tumeur visible après injection de produit de contraste.

Le dataset contient des cas avec différents grades de tumeurs, ce qui permet de développer des algorithmes capables de segmenter aussi bien des gliomes à haut grade que des gliomes à bas grade.

6.2 OASIS (Open Access Series of Imaging Studies)

Le dataset OASIS (Open Access Series of Imaging Studies)[60] est une ressource publique utilisée dans le domaine de la neuroimagerie pour l’étude de diverses affections cérébrales, notamment la maladie d’Alzheimer. Il a été conçu pour permettre aux chercheurs de mieux comprendre la structure du cerveau humain au cours du vieillissement, et pour soutenir la recherche sur les troubles neurodégénératifs. Il comprend des images IRM de participants d’âges variés, allant des jeunes adultes en bonne santé à des personnes âgées, dont certaines atteintes de la maladie d’Alzheimer. Le dataset aussi propose des images IRM en haute résolution, capturant des informations structurelles précises sur le cerveau. En plus des images, OASIS fournit des données cliniques telles que l’évaluation cognitive, le diagnostic de la maladie d’Alzheimer, et des informations démographiques.

6.3 TCGA-GBM (The Cancer Genome Atlas - Glioblastoma Multiforme)

Le dataset TCGA-GBM (The Cancer Genome Atlas - Glioblastoma Multiforme)[10] est un jeu de données riche et multidimensionnel issu du projet TCGA (The Cancer Genome Atlas), un programme collaboratif à grande échelle visant à cartographier les altérations génomiques des cancers humains. Le TCGA-GBM se concentre spécifiquement sur le glioblastome multiforme (GBM), un type de tumeur cérébrale maligne très agressive. Ce dataset fournit des données détaillées et variées pour les chercheurs travaillant à comprendre les mécanismes moléculaires et génomiques du GBM, ainsi que pour le développement de nouvelles stratégies thérapeutiques.

6.4 TCGA-LGG (The Cancer Genome Atlas - Lower Grade Glioma)

Le dataset TCGA-LGG (The Cancer Genome Atlas - Lower Grade Glioma) [13] est une ressource publique précieuse pour l’étude des gliomes de bas grade (LGG), qui sont des tumeurs cérébrales moins agressives que le glioblastome multiforme (GBM). Ce dataset fait partie du projet Cancer Genome Atlas (TCGA) et a été conçu pour fournir des informations complètes et multidimensionnelles sur les altérations moléculaires et génomiques des gliomes de bas grade. Il vise à aider les chercheurs à comprendre les

mécanismes sous-jacents à ces types de tumeurs et à améliorer les stratégies diagnostiques et thérapeutiques.

6.5 Le REMBRANDT (Repository of Molecular Brain Neoplasia Data)

Le REMBRANDT [59] est un dataset public destiné à la recherche sur les tumeurs cérébrales, principalement les gliomes, qui sont les tumeurs les plus courantes du système nerveux central. REMBRANDT a été créé dans le cadre d’une collaboration entre le National Cancer Institute (NCI) et le National Institutes of Health (NIH), avec pour objectif d’intégrer des données cliniques et génomiques afin de favoriser la recherche sur le cancer du cerveau et d’améliorer la compréhension des mécanismes moléculaires sous-jacents aux tumeurs cérébrales malignes.

Conclusion

Dans la conclusion met en lumière l’importance croissante de l’imagerie médicale dans le domaine de la détection d’objets, en particulier pour l’identification des tumeurs et des cellules cancéreuses. Après avoir passé en revue les techniques d’imagerie médicale et leurs limites, ainsi que la diversité des tumeurs, le chapitre souligne l’intérêt des méthodes de détection d’objet pour localiser et classifier les anomalies. Enfin, l’intégration de l’intelligence artificielle dans ce contexte promet d’améliorer la précision diagnostique, ouvrant ainsi la voie à de nouvelles avancées dans le traitement des maladies.

Chapter 2

Apprentissage Profond

Apprentissage Profond

Introduction

L'imagerie médicale est l'un des piliers de la médecine moderne, permettant aux médecins de voir à l'intérieur du corps humain sans recourir à des interventions chirurgicales. Depuis la découverte des rayons X par Wilhelm Röntgen en 1895, le domaine de l'imagerie médicale a considérablement progressé [78]. Ses techniques telles que l'échographie, la tomographie informatisée (CT) et l'imagerie par résonance magnétique (IRM) se sont multipliées. Cela a grandement aidé à diagnostiquer les maladies avec facilité et a gagné du temps pour accélérer le traitement, fournissant des images précises et détaillées des structures internes du corps [11].

L'apprentissage profond, une branche de l'intelligence artificielle inspirée par le travail du cerveau humain, apparaît comme une technologie prometteuse dans l'analyse et le traitement de données volumineuses et complexes [31]. En particulier, les réseaux de neurones convolutifs (CNN) ont démontré une capacité exceptionnelle à reconnaître les motifs et les caractéristiques des images, dépassant souvent les méthodes traditionnelles dans divers domaines, y compris l'imagerie médicale [54].

2 Apprentissage Profond

L'apprentissage profond est une branche de l'intelligence artificielle qui utilise des réseaux de neurones artificiels composés de plusieurs couches pour modéliser et apprendre des données complexes. Dans le domaine de la détection d'objets, l'apprentissage profond est largement utilisé pour extraire des caractéristiques significatives à partir d'images et pour classifier et localiser différents objets dans ces images. Les réseaux de neurones convolutifs (CNN) sont particulièrement efficaces pour cette tâche, car ils sont capables d'apprendre des représentations hiérarchiques des objets à différentes échelles et niveaux de complexité.

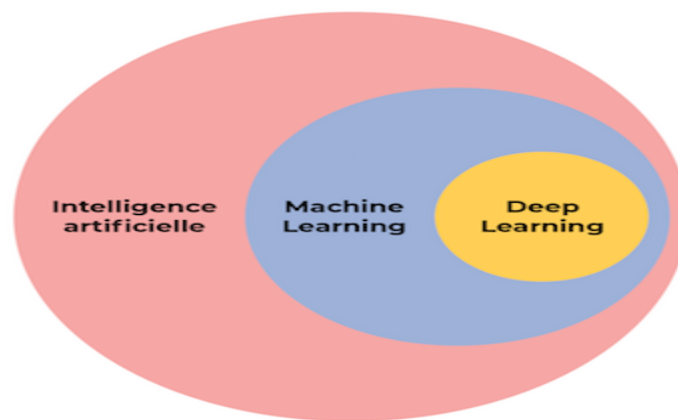


FIGURE 2.1 – Apprentissage Profond [2]

3 Réseaux neuronaux artificiels (RNA)

Les premiers réseaux neuronaux artificiels (RNA) capables d'apprentissage ont été mis au point par Rosenblatt en 1959. Les premiers perceptrons étaient limités à la résolution de problèmes linéaires simples. Les RNA, inspirés des réseaux de neurones biologiques, sont des méthodologies informatiques effectuant des analyses multifactorielles. En biologie, les réseaux neuronaux sont constitués de nombreux neurones interconnectés, chaque neurone pouvant émettre un signal électrochimique et interagir avec d'autres neurones via des dendrites. Les signaux sont transmis à travers un réseau de structures ramifiées vers la périphérie. Les interactions entre les axones et les dendrites par l'intermédiaire des synapses facilitent l'interprétation des signaux. La réception d'un signal entraîne une réponse régulée par les synapses, un processus crucial dans l'apprentissage.

Les RNA contiennent des couches de nœuds de données qui interagissent par des connexions pondérées pour traiter les signaux. La pondération des connexions est ajustée lors de séances d'apprentissage avec des données entrantes. Les architectures des RNA actuels sont plus simples que les réseaux biologiques, avec moins de nœuds et d'interconnexions.

Les cliniciens utilisent des données cliniques pour évaluer la condition des patients et prendre des décisions diagnostiques et thérapeutiques. De manière similaire, un RNA passe par une phase d'apprentissage où les données sont réparties en trois sous-groupes :

ensemble d'apprentissage, ensemble de validation (ou de test) et ensemble de vérification. L'ensemble d'apprentissage ajuste les poids des connexions pour que les résultats de sortie se rapprochent des résultats réels. Le processus d'apprentissage se poursuit jusqu'à ce que l'erreur soit minimisée. L'ensemble de validation décide quand arrêter l'apprentissage pour éviter le surapprentissage. L'ensemble de vérification, indépendant, évalue les performances du RNA après l'apprentissage.

Les RNA, avec la puissance de calcul moderne, peuvent traiter des phénomènes non linéaires sans nécessiter de relations linéaires simples. Ils peuvent corrélérer divers facteurs pronostiques, détecter des interactions cachées entre variables, prévoir des évolutions, classer des patients en groupes de risque et approximer des fonctions. Des études ont souvent montré que les RNA surpassent les méthodes traditionnelles, comme la régression logistique, en termes de sensibilité et de spécificité.

3.1 Structure d'un RNA

Un RNA est composé de plusieurs couches de nœuds (ou neurones) interconnectés. Ces couches comprennent :

1. **Couche d'entrée** : Cette couche reçoit les données brutes en entrée. Chaque neurone de cette couche représente une caractéristique spécifique des données d'entrée. Par exemple, dans une image, chaque neurone peut représenter la valeur d'un pixel [31].
2. **Couches cachées** : Ces couches se situent entre la couche d'entrée et la couche de sortie. Elles sont appelées "cachées" car elles ne sont pas visibles dans les données d'entrée ou de sortie. Les neurones dans ces couches cachées effectuent des calculs et des transformations sur les données entrantes pour extraire des caractéristiques plus abstraites. Un RNA peut avoir une ou plusieurs couches cachées [50].
3. **Couche de sortie** : Cette couche produit les résultats finaux du réseau. Les neurones de la couche de sortie représentent les classes de prédiction ou les valeurs de sortie souhaitées. Par exemple, dans une tâche de classification, chaque neurone de la couche de sortie peut représenter une classe spécifique [31].

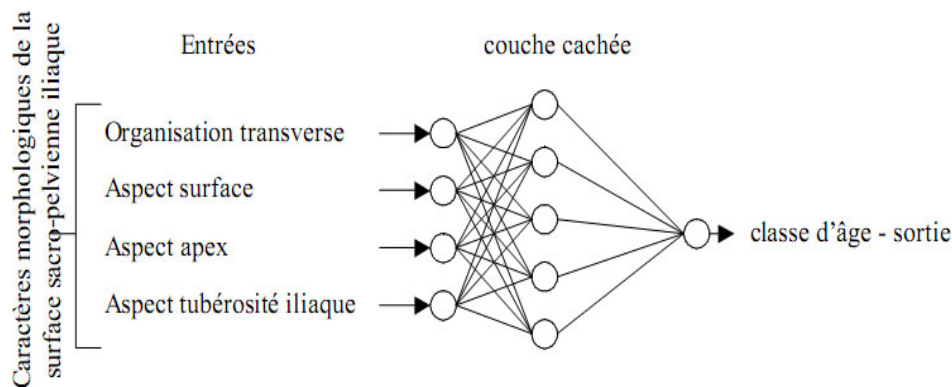


FIGURE 2.2 – Structure Réseaux de neurones artificiels [3]

3.2 Fonctionnement de base d'un RNA

Un réseau de neurones artificiels se compose d'au moins deux couches, chacune contenant plusieurs neurones ou nœuds, reliés entre eux d'une couche à l'autre [40]. Chaque neurone est associé à des données d'entrée, un poids, un seuil et des données de sortie, résultant de la multiplication de la valeur d'entrée du neurone par son poids. Lorsque les données de sortie d'un neurone dépassent le seuil spécifié, le neurone est activé et transmet ses données aux neurones de la couche suivante, et ainsi de suite.

L'apprentissage d'un réseau de neurones artificiels implique l'ajustement itératif des poids associés à chaque nœud, en partant de petites valeurs, dans le but de minimiser l'écart avec le résultat souhaité [50]. Pour ce faire, un ensemble d'apprentissage est utilisé, composé de paires associant une donnée d'entrée à une cible à atteindre. À chaque donnée d'entrée, le réseau estime les poids du réseau et une fonction d'erreur est utilisée pour calculer la différence entre la prévision réalisée et la valeur cible, permettant ainsi d'ajuster les poids progressivement. Ce processus est répété pour chaque paire de la base d'apprentissage.

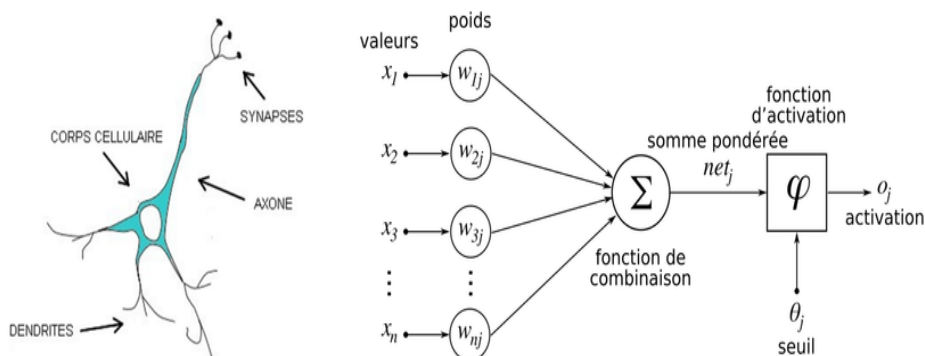


FIGURE 2.3 – Réseaux de neurones artificiels [49]

4 Réseaux de neurones convolutifs (CNN)

Les réseaux de neurones convolutifs (CNN), également appelés ConvNets, sont des réseaux de neurones artificiels spécialisés dans la reconnaissance de motifs dans les données. Principalement utilisés pour l'analyse d'images, ils ont également démontré leur efficacité dans des domaines tels que la reconnaissance vocale, le traitement du langage naturel et la détection d'objets [48] [51]. Les CNN se distinguent par l'utilisation de couches de convolution pour extraire des caractéristiques significatives des données d'entrée, suivies de couches de pooling pour réduire la dimensionnalité tout en préservant l'information essentielle [81]. Ces réseaux utilisent également des couches entièrement connectées pour la classification ou la régression sur les caractéristiques extraites [41]. Un avantage clé des CNN est leur capacité à capturer automatiquement des caractéristiques hiérarchiques à partir de données brutes, réduisant ainsi le besoin d'ingénierie manuelle des caractéristiques [80]. Cette capacité les rend particulièrement puissants dans les applications de vision par ordinateur, telles que la classification d'images, la détection d'objets et la reconnaissance faciale [48] [51].

4.1 Les Couches de CNN

Les réseaux de neurones convolutifs (CNN) sont devenus essentiels dans le domaine de la vision par ordinateur et du traitement d'images. Leur efficacité repose sur une architecture structurée en différentes couches spécialisées, chacune jouant un rôle crucial dans l'extraction et la transformation des caractéristiques des images d'entrée. Grâce à l'enchaînement de ces opérations spécifiques, les CNN peuvent capturer des détails à divers niveaux de complexité et de profondeur. Cela permet de réaliser des tâches variées telles que la reconnaissance d'objets, la classification d'images et la détection de motifs. Dans cette section, nous examinerons les principales couches constituant un CNN : la couche convolutive, la couche de mise en commun et la couche entièrement connectée, en détaillant leur fonctionnement et leur contribution à l'apprentissage et à la performance globale du réseau.

- **Couche Convulsive (CONV)** : La couche convulsive est essentielle dans les CNN, généralement placée en première position. Elle a pour rôle de détecter des features (caractéristiques) spécifiques dans les images d'entrée. Elle réalise cela via un filtrage par convolution : une fenêtre représentant une feature se déplace sur l'image et effectue le produit de convolution pour chaque portion parcourue. Chaque feature agit comme un filtre, et ces filtres sont appris par le réseau pendant la phase d'entraînement. Les noyaux des filtres, qui sont les poids de la couche convulsive, sont ajustés via la rétropropagation du gradient.

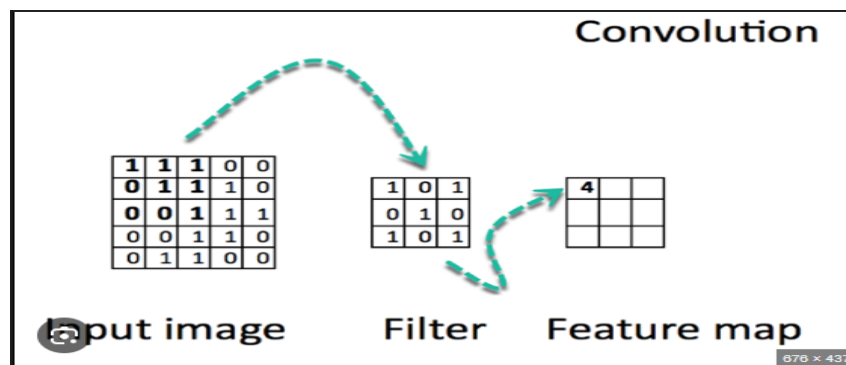


FIGURE 2.4 – Couche Convulsive (CONV)[36]

- **Couche de Mise en Commun (POOL)** : Cette couche, souvent située entre deux couches convolutives, réduit la dimension spatiale des cartes d’activation (feature maps) en prenant soit la valeur maximale (max pooling) soit la moyenne (average pooling) dans chaque région. Cela permet de diminuer le nombre de paramètres et de rendre le modèle plus robuste aux variations d’échelle et de position dans l’image.

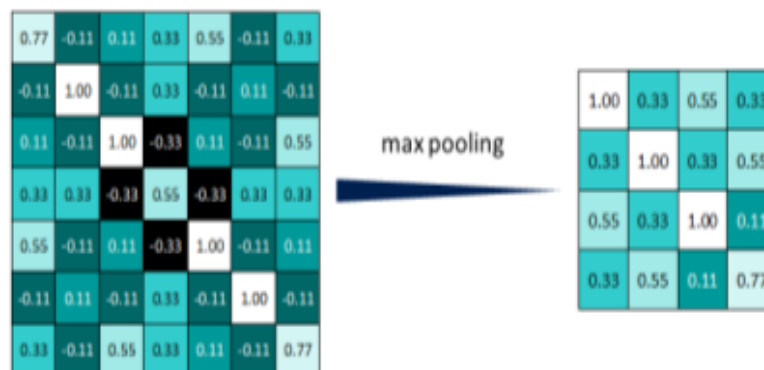


FIGURE 2.5 – Couche POOL[20]

- **Couche d’Activation ReLU (Rectified Linear Units)** : Appliquée après les couches convolutives ou de pooling, la fonction d’activation ReLU remplace les valeurs négatives par zéro et conserve les valeurs positives inchangées, introduisant ainsi une non-linéarité dans le modèle, ce qui est essentiel pour apprendre des représentations complexes.
- **Couche Entièrement Connectée (Fully-Connected, FC)** : La dernière couche d’un CNN est souvent une couche entièrement connectée (ou dense), qui utilise les caractéristiques extraites par les couches précédentes pour effectuer la classification finale. Chaque neurone de cette couche est connecté à tous les neurones de la couche précédente, ce qui permet d’apprendre des combinaisons linéaires des features.
- **Couche de Sortie (Output Layer)** : La couche de sortie génère les prédictions finales du modèle. Dans une tâche de classification, elle utilise souvent une fonction softmax pour convertir les valeurs de sortie en probabilités, représentant la classe

d'objet prédit par le réseau.

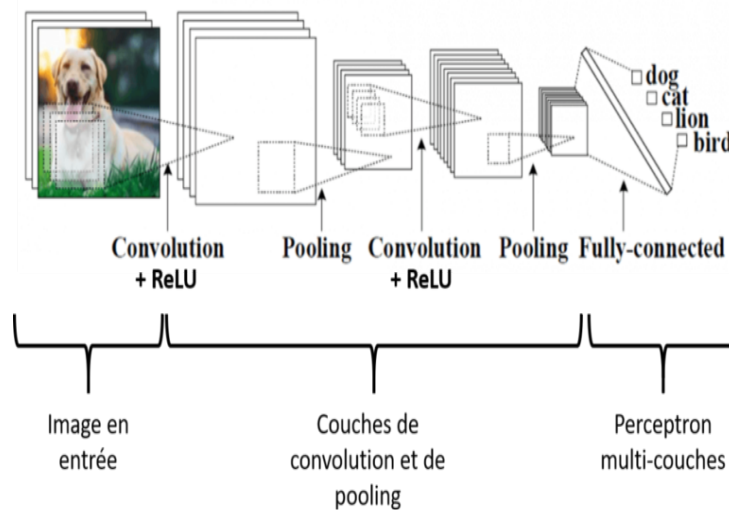


FIGURE 2.6 – Les Couches de CNN [63]

4.2 Quelques architectures de base de CNN

Avant de plonger dans les détails des architectures emblématiques de CNN, prenons un moment pour les mettre en contexte. Les réseaux de neurones convolutifs ont marqué un tournant majeur dans le domaine de l'apprentissage profond, en particulier dans le traitement d'images. Leur capacité à extraire des caractéristiques complexes à partir de données visuelles a conduit à des avancées significatives dans des domaines tels que la reconnaissance d'objets, la classification d'images et la détection de motifs. Dans cette exploration, nous passerons en revue cinq architectures clés de CNN, allant de LeNet-5, pionnier dans la reconnaissance de caractères, à ResNet, célèbre pour sa profondeur impressionnante et ses connexions de saut novatrices. En analysant ces architectures, nous découvrirons leurs contributions uniques et leurs implications dans le domaine en constante évolution du traitement d'images, le tableau ci-dessous représente quelques architectures de CNN.

TABLE 2.1 – Architectures de réseaux de neurones et leurs descriptions

Architecture	Architecte	Année	Description
LeNet-5 [51]	Yann LeCun, Leon Bottou, Yosuha Bengio et Patrick Haffner	1998	LeNet-5 était initialement conçu pour la reconnaissance de caractères manuscrits et imprimés. Il comprend deux ensembles de couches convolutives, chacun suivi d'une couche de pooling, puis d'une couche de "Flattening Convolutional". Enfin, deux couches entièrement connectées et un classifieur Softmax.

Architecture	Architecte	Année	Description
AlexNet [48]	Alex Krizhevsky, Geoffrey Hinton et Ilya Sutskever	2012	AlexNet a été le premier à démontrer l'efficacité des CNN sur des tâches de classification d'images. Il est profond avec huit couches, dont cinq couches convolutives suivies de couches de pooling maximal, et trois couches entièrement connectées. Il utilise la fonction d'activation ReLU.
VGGNet [7]	Simonyan et Zisserman	2014	VGGNet est connu pour sa profondeur remarquable. Il comprend 13 couches convolutives avec des filtres de taille 3x3 et des couches de pooling, suivies de trois couches entièrement connectées. Il utilise la fonction ReLU dans toutes les couches cachées.
GoogLeNet (Inception) [71]	Christian Szegedy et autres chercheurs de Google	2014	GoogLeNet a introduit le concept de "blocs Inception", qui permettent d'effectuer différentes opérations de convolution simultanément. Il comprend également des couches de pooling et des couches entièrement connectées. Sa complexité architecturale lui permet de capturer des caractéristiques à différentes échelles.
ResNet [87]	Kaiming He	2015	ResNet est célèbre pour ses "connexions de saut" ou "skip connections", qui permettent d'apprendre des réseaux très profonds en facilitant le flux d'informations à travers les couches. Il comprend 152 couches et a été introduit pour résoudre le problème de la disparition du gradient dans les réseaux profonds.

5 L'apprentissage par transfert

L'apprentissage par transfert (ou Transfer Learning) est une technique utilisée en intelligence artificielle, principalement dans le domaine de l'apprentissage automatique et de l'apprentissage profond. Cette méthode consiste à réutiliser un modèle préalablement entraîné sur une tâche spécifique pour l'appliquer à une nouvelle tâche, souvent différente mais liée. Cela permet de tirer parti des connaissances acquises par le modèle lors de son premier apprentissage et de réduire le besoin d'une grande quantité de données pour la nouvelle tâche.

Le principe clé de l'apprentissage par transfert est que les représentations apprises

dans les couches profondes d'un réseau neuronal peuvent capturer des caractéristiques générales des données (par exemple, des formes dans des images), qui sont utiles même dans des contextes différents. Ce type d'apprentissage est particulièrement efficace dans des scénarios où les données annotées sont rares ou coûteuses à obtenir, comme dans la reconnaissance d'image, le traitement du langage naturel, et même dans des domaines comme la médecine et la reconnaissance vocale.

L'apprentissage par transfert peut être mis en œuvre de plusieurs façons, dont :

- **Transfert de caractéristiques** : On conserve les premières couches du réseau d'origine (qui capturent des caractéristiques génériques) et on ne réentraîne que les dernières couches pour adapter le modèle à la nouvelle tâche.
- **Apprentissage fin (fine-tuning)** : Le modèle préentraîné est ajusté en réentraînant l'ensemble du modèle ou une partie sur le nouveau jeu de données, mais avec une petite vitesse d'apprentissage.
- **Congélation partielle des couches** : Certaines couches du modèle sont "gelées", c'est-à-dire qu'elles ne sont pas réentraînées, tandis que d'autres sont réentraînées pour s'adapter à la nouvelle tâche[68].

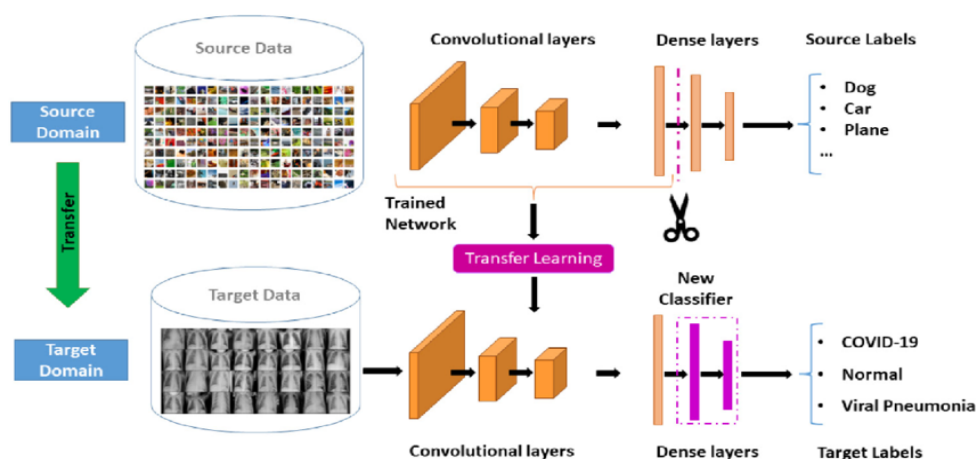


FIGURE 2.7 – Processus d'apprentissage par transfert[14]

ji2019optimized

5.1 ResNet50

Le ResNet-50 (Residual Network avec 50 couches) est un modèle de réseau de neurones profond, introduit par Kaiming He et ses collègues dans le cadre des réseaux résiduels (Residual Networks), lors du concours ImageNet de 2015. ResNet-50 est l'une des architectures de réseaux de neurones les plus populaires, principalement utilisée pour des tâches de classification d'images, de détection d'objets et de reconnaissance faciale, entre autres.

Le principal atout des réseaux ResNet est l'utilisation des connexions résiduelles (ou skip connections), qui permettent de contourner certaines couches du réseau en ajoutant directement l'entrée de ces couches à leur sortie. Cette architecture résiduelle a été proposée pour résoudre le problème du vanishing gradient (décroissance du gradient) qui affecte l'apprentissage des réseaux très profonds.

En intégrant ces connexions résiduelles, ResNet facilite l'entraînement des réseaux plus profonds, car il permet au modèle de mieux propager les gradients lors de la rétropropagation. Cela empêche le réseau de se dégrader, un problème courant avec les architectures traditionnelles qui deviennent inefficaces au-delà d'un certain nombre de couches.

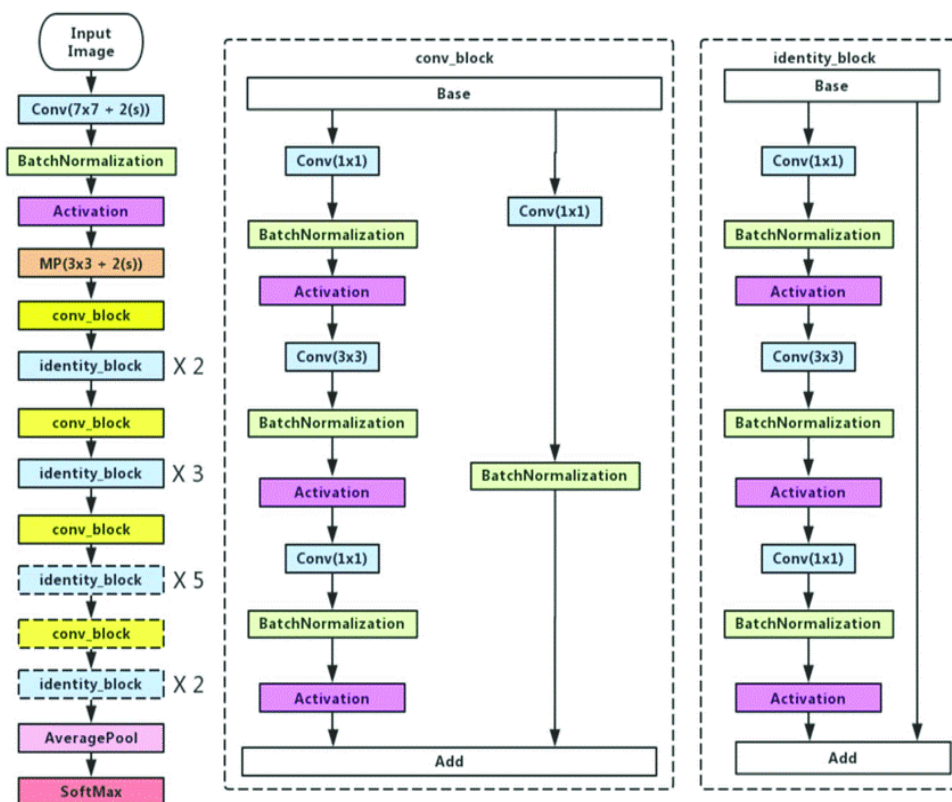


FIGURE 2.8 – Architecture de ResNet50 [44]

5.2 Types de connexions résiduelles

Les connexions résiduelles sont une caractéristique clé des réseaux résiduels (ResNets) et permettent de contourner certaines couches du réseau en ajoutant directement l'entrée d'une couche à sa sortie. Cela aide à faciliter l'entraînement de réseaux très profonds en atténuant les problèmes liés à la disparition des gradients (vanishing gradient problem). Il existe principalement deux types de connexions résiduelles.

1. Connexion Résiduelle Identitaire

Dans une connexion résiduelle identitaire, l'entrée d'une couche est ajoutée directement à sa sortie, sans modification. Formulé mathématiquement, si \mathbf{x} est l'entrée et $\mathbf{F}(\mathbf{x})$ la transformation non linéaire (comme une convolution suivie d'une activation ReLU), la sortie est :

$$\text{Sortie} = F(x) + x$$

2. Connexion résiduelle avec projection

Lorsque les dimensions de l'entrée et de la sortie sont différentes (par exemple,

en cas de modification de la taille du filtre ou de la résolution), une opération de projection est appliquée à l'entrée pour ajuster ses dimensions afin de l'additionner à la sortie. Cette projection peut se faire via une couche de convolution 1x1 ou un autre type de transformation linéaire.

$$\text{Sortie} = F(x) + Wsx$$

Où W_s est la matrice de projection qui ajuste les dimensions de l'entrée x pour correspondre à celles de $F(x)$ [42].

6 Les travaux Conex

Les tumeurs cérébrales, bien que représentant environ 2% de tous les cancers diagnostiqués chaque année, sont parmi les types de cancer les plus graves et dangereux en raison de leur localisation dans le cerveau et le système nerveux central. Elles présentent un défi significatif en termes de diagnostic et de traitement, en partie à cause de la complexité et de la sensibilité des structures cérébrales qu'elles affectent [67]. L'imagerie par Résonance Magnétique (IRM) est la méthode d'imagerie la plus couramment utilisée pour détecter ces tumeurs. Elle offre des images détaillées des structures cérébrales et est particulièrement efficace pour visualiser les tumeurs et les anomalies.

La revue est divisée en une section qui examine les approches hybrides, puis nous passons à la section sur l'apprentissage profond. Enfin, nous concluons la revue par un tableau contenant les travaux publiés qui traitent de la détection et la classification des tumeurs cérébrales.

6.1 Approches hybrides pour la classification des tumeurs cérébrales à l'aide de données d'IRM

Deepak et al [22] ont introduit une approche d'apprentissage d'ensemble pour la classification des tumeurs cérébrales trois classes pour différencier les gliomes, les méningiomes et Pituitary. Leur méthode combine plusieurs classificateurs, dont les forêts aléatoires (RF), les machines à renforcement de gradient (GBM) et les machines à vecteurs de support (SVM), afin d'améliorer la précision de la classification. La méthode d'ensemble a tiré parti des points forts de chaque classificateur, ce qui a permis d'obtenir un modèle robuste pour la classification des tumeurs à l'aide de données d'IRM. Le cadre de classification proposé utilise le GoogLeNet modifié et affiné pour extraire des caractéristiques des images IRM du cerveau. Les caractéristiques extraites sont ensuite classifiées en utilisant des modèles de classificateurs éprouvés, tels que le support vector machine (SVM) et le k-nearest neighbours (KNN). Ces classificateurs sont choisis en raison de leur performance éprouvée dans la classification des caractéristiques CNN. Les expériences sont réalisées sur un jeu de données ouvert provenant de figshare, comprenant 3064 images IRM de cerveaux de 233 patients diagnostiqués. Les principaux résultats obtenus montrent des taux de **Accuracy** :

- Modèle de transfert autonome : $92.3\% \pm 0.7\%$
- SVM sur les caractéristiques profondes du CNN : $97.8\% \pm 0.2\%$
- KNN sur les caractéristiques profondes du CNN : $98.0\% \pm 0.4\%$

La méthode proposée par Mathiyalagan et tous se concentre sur le développement d'un système de détection et de classification des tumeurs cérébrales de type gliome à l'aide de techniques d'apprentissage automatique appliquées aux images IRM. La méthode commence par des étapes de prétraitement visant à améliorer la qualité de l'image et la délimitation des tumeurs. Ces étapes comprennent la réduction du bruit à l'aide de filtres ridgelet, la détection des contours à l'aide de la logique floue et l'amélioration du contraste grâce à l'égalisation locale de l'histogramme. Ensuite, l'extraction des caractéristiques est effectuée à l'aide de la transformation de Gabor qui capture les caractéristiques de texture et de forme cruciales pour distinguer les régions tumorales des régions non tumorales. Les caractéristiques extraites sont ensuite optimisées à l'aide d'algorithmes génétiques afin de sélectionner les caractéristiques les plus discriminantes pour la classification ultérieure. Pour la classification, les auteurs utilisent un système d'inférence adaptatif neuro-flou (ANFIS) et la segmentation des régions de gliome est réalisée à l'aide de l'algorithme C-means flou. Les auteurs fournissent une évaluation complète de leur approche en utilisant deux ensembles de données en libre accès. Une précision moyenne de détection et de segmentation obtenus supérieure à 85%.

Ayadi et all propose une nouvelle technique pour classifier trois types de tumeurs cérébrales : gliome, méningiome et tumeur pituitary. a méthode suggérée comprend trois étapes principales :

1. **Normalisation d'Intensité** : La normalisation des valeurs d'intensité des images MRI.
2. **Extraction des Caractéristiques** : Deux techniques sont utilisées pour extraire des caractéristiques discriminantes des images MRI
 - **Dense Speeded Up Robust Features (DSURF)** Technique utilisée pour détecter et décrire des points d'intérêt dans les images.
 - **Histogram of Oriented Gradients (HoG)** Technique qui compte les occurrences d'orientation des gradients dans des portions localisées de l'image, souvent utilisée pour la reconnaissance d'objets.
3. **Classification** : Un classificateur Support Vector Machine (SVM)[4] est employé pour la classification des tumeurs en utilisant les caractéristiques extraites. Le SVM est choisi pour sa capacité à gérer des problèmes de classification multi-classes de manière efficace. L'étude utilise un jeu de données comprenant 3064 images MRI provenant de 233 patients, acquises de 2005 à 2010 dans des hôpitaux en Chine. Les images sont segmentées manuellement pour inclure les tissus environnants de la tumeur, ce qui permet d'extraire des caractéristiques indicatives. Les résultats obtenus montrent que la méthode proposée atteint une précision de classification de 90.27% Cette précision a été obtenue grâce à une analyse statistique stricte utilisant la validation croisée k-fold.

Asaf Raza et al[72] propose une approche hybride basée sur l'apprentissage profond pour la classification des tumeurs cérébrales. L'objectif principal est de classer trois types de tumeurs cérébrales : gliome, méningiome et tumeur hypophysaire, en utilisant une architecture CNN modifiée. L'approche proposée, nommée DeepTumorNet, utilise l'architecture GoogLeNet comme base. Les cinq dernières couches de GoogLeNet ont été retirées et remplacées par quinze nouvelles couches. De plus, la fonction d'activation Leaky ReLU a été intégrée pour augmenter l'expressivité du modèle. Cette architecture hybride combine les techniques de transfert

d'apprentissage pour améliorer les performances de la classification des images IRM. Le modèle DeepTumorNet a été évalué sur un dataset public, CE-MRI, contenant des images IRM des trois types de tumeurs mentionnés. Les résultats obtenus montrent une précision de 99,67%, une précision de 99,6%, un rappel de 100%, et un F1-score de 99,66%.

Gumaei et al propose une méthode hybride d'extraction de caractéristiques combinée avec une machine d'apprentissage extrême régularisée (RELM) pour la classification des tumeurs cérébrales. La méthodologie se compose de trois étapes principales :

- **Prétraitement des Images Cérébrales** : Les images brutes sont transformées en images d'intensité normalisées dans la plage $[0, 1]$ en utilisant une règle de normalisation min-max. Cette transformation améliore le contraste des bords et des régions du cerveau, facilitant ainsi l'extraction des caractéristiques.
- **Extraction des Caractéristiques** : Le descripteur GIST est appliqué pour capturer la structure spatiale dominante des images cérébrales, suivi d'une normalisation avec la norme L2 pour atténuer les variations d'illumination et d'ombrage. Par la suite, une analyse en composantes principales (PCA) est réalisée sur les descripteurs GIST normalisés (NGIST) afin de réduire la dimensionnalité tout en conservant les caractéristiques significatives pour la classification.
- **Classification des Tumeurs Cérébrales** : La machine d'apprentissage [34] extrême régularisée (RELM) est employée pour la classification des types de tumeurs cérébrales, sélectionnée pour sa rapidité de formation et sa régularisation efficace, ce qui prévient le surapprentissage. Les paramètres de cette approche sont optimisés via un algorithme de recherche par grille.

L'ensemble de données utilisé pour les expérimentations comprend un total de 3064 images IRM, réparties en 708 images de méningiomes, 1426 images de gliomes et 930 images de tumeurs hypophysaires. L'exactitude de classification a été améliorée de 91,51% à 94,233% en utilisant la technique de validation croisée aléatoire.

L'étude	Méthode de travail	Type de classification	DataSet	Résultats
[Deepak et al., 2019]	GoogLeNet modifié pour extraire des caractéristiques classifiées en utilisant SVM, KNN	Classification Multiclasse	Dataset Public de 3064 images IRM	Accuracy : -Modèle de transfert autonome : 92.3% -SVM : 97.8% -KNN : 98.0%
[Ayadi et al., 2020]	- la transformation de Gabor pour extraire des caractéristiques ensuite optimisées à l'aide d'algorithmes génétiques - classifiées en utilisant SVM	Classification Multiclasse	Dtaset Public de 3064 image IRM	Accuracy : 90.27
[Asaf Raza et al., 2022]	approche proposée, nommée DeepTumorNet utilise l'architecture GoogLeNet comme base	Classification Multiclasse	Dtaset Public de 3064 image IRM	Accuracy : 99,67- précision ; 99,6- rap- pel :100- F1-score : 99,66
[Gumaei et al., 2019]	- GIST pour capturer la structure spatiale dominante. - PCA sur les descripteurs GIST normalisés pour générer un sous-ensemble de caractéristiques significatives et réduire la dimensionnalité. - RELM pour classer les types de tumeurs cérébrales	Classification Multiclasse	Dtaset Public de 3064 image IRM	Accuracy : 94,233 %

TABLE 2.2 – Résumé pour Approches hybrides pour la classification des tumeurs cérébrales.

6.2 Approches d'apprentissage profond pour la classification des tumeurs cérébrales à l'aide de données d'IRM

Vinod Kumar Dhakshnamurthy et al [23] explore l'utilisation des modèles de deep learning pour la détection et la classification des tumeurs cérébrales, en mettant l'accent sur l'efficacité des modèles de transfert d'apprentissage tels que AlexNet, VGG16, et ResNet-50. Ce travail vise à améliorer la précision et l'efficacité du diagnostic médical, un processus traditionnellement coûteux et exigeant en termes de temps. L'étude utilise un ensemble de données provenant de Kaggle, composé de 3264 images IRM de différents types de tumeurs cérébrales : gliome, méningiome, tumeur pituitaire, et absence de tumeur. Les images sont prétraitées et divisées en ensembles d'entraînement, de test, et de validation dans des proportions de 80%, 10%, et 10% respectivement. Les modèles de base évalués incluent AlexNet, VGG16, et ResNet-50. Ces modèles sont ajustés pour correspondre au nombre de classes de l'ensemble de données. L'architecture hybride proposée, combinant VGG16 et ResNet-50, a montré des performances exceptionnelles avec **Accuracy** et une spécificité de 99,98%, et un score F1 de 99,98%. Ces résultats surpassent ceux obtenus par les modèles individuels et les autres méthodes comparées dans l'étude. L'article met en lumière plusieurs points importants :

- Les modèles de transfert d'apprentissage, notamment VGG16 et ResNet-50, ont démontré une haute efficacité dans la classification des tumeurs cérébrales.
- La combinaison des modèles (hybride VGG16-ResNet-50) a permis d'atteindre des niveaux de précision remarquables, dépassant les performances des modèles pris individuellement.

Sarah Ali Abdelaziz Ismael el al [43] présente une méthode améliorée pour la classification des images IRM de tumeurs cérébrales en utilisant des réseaux résiduels (ResNet). Les données utilisées proviennent d'un ensemble de données disponible publiquement, composé de 3064 images IRM de différents types de tumeurs cérébrales : gliome, méningiome, tumeur pituitaire . Le modèle utilisé est basé sur l'architecture ResNet50, un réseau résiduel comprenant 50 couches. Le modèle a été entraîné pendant 500 époques . Le modèle a obtenu une précision moyenne de 99% pour la classification au niveau de l'image et de 97% au niveau du patient. Le score F1 macro-moyenné a été de 99% pour la classification au niveau de l'image et de 97% pour celle au niveau du patient .

Badža et al [5] propose une méthodologie détaillée pour la classification des tumeurs cérébrales à partir d'images IRM en utilisant un réseau de neurones convolutifs (CNN) , Les images utilisées proviennent d'une base de données contenant 3064 images IRM pondérées en T1 avec contraste . Les types de tumeurs inclus sont les méningiomes, les gliomes et les tumeurs hypophysaires . Le CNN développé se compose de plusieurs blocs : un bloc d'entrée, deux blocs principaux, un bloc de classification et un bloc de sortie. Le modèle a été implémenté avec Matlab R2018a . La performance du réseau a été testée en utilisant quatre approches différentes combinant deux méthodes de validation croisée (record-wise et subject-wise) et deux bases de données (originale et augmentée). Les résultats sont présentés sous forme de matrices de confusion et de mesures de precision 97.15%, rappel 97.82%, F1-score 97.47%, et exactitude 97.28%.

L'étude	Méthode de travail	Type de classification	DataSet	Résultats
[Vinod Kumar Dhakshnamurthy et al , 2024]	- L'architecture proposée, combinant VGG16 et ResNet-50	Classification Multiclasse	Dataset Public de 3264 images IRM	- une spécificité : 99,98- score F1 : 99,98
[Sarah Ali Abdelaziz Ismael et al , 2019]	- modèle utilisé est basé sur l'architecture ResNet50	Classification Multiclasse	Dataset Public de 3064 image IRM	Accuracy : 97.28%. - de précision : 97.15% - rappel : 97.82%, - F1-score : 97.47%,
[Badža et al , 2020]	- un réseau de neurones convolutifs (CNN)	Classification Multiclasse	Dataset Public de 3064 image IRM	Accuracy : 97.28- de précision : 97.15% - rappel : 97.82%, - F1-score : 97.47%,

TABLE 2.3 – Approches d'apprentissage profond pour la classification des tumeurs cérébrales à l'aide de données d'IRM.

Conclusion

L'apprentissage profond, notamment via les réseaux de neurones convolutifs (CNN), a transformé l'analyse d'images, particulièrement en imagerie médicale. Il s'est imposé comme un outil clé pour la reconnaissance de motifs et la classification d'images. L'apprentissage par transfert offre également des solutions pour adapter les modèles à de nouvelles tâches, même avec peu de données. En résumé, l'apprentissage profond améliore la précision et la rapidité du diagnostic, contribuant ainsi à l'efficacité des soins.

Chapter 3

Conception

Conception

Introduction

La détection précoce des tumeurs cérébrales représente un défi majeur dans le domaine de la médecine, avec un impact significatif sur les options de traitement et les taux de survie des patients. Les tumeurs cérébrales, en tant que pathologies complexes, nécessitent une identification précise et rapide pour permettre une intervention médicale efficace. Les méthodes d'imagerie cérébrale, telles que l'IRM, fournissent des informations cruciales pour le diagnostic, mais l'analyse manuelle des images peut être laborieuse et sujette à des erreurs.

Dans ce contexte, l'application des techniques d'intelligence artificielle, et plus particulièrement du deep learning, offre des perspectives prometteuses pour améliorer la détection et la classification des tumeurs cérébrales. Ce chapitre explore la conception et la méthodologie de notre approche pour la classification des tumeurs cérébrales en utilisant un modèle avancé de deep learning basé sur ResNet50, enrichi par l'intégration du Convolutional Block Attention Module (CBAM).

Nous débuterons par une explication des étapes de prétraitement des données, qui incluent le recadrage des images pour se concentrer sur les régions d'intérêt, la normalisation pour harmoniser les valeurs des pixels, et le redimensionnement pour adapter les images à l'entrée du modèle. Ces étapes sont cruciales pour préparer les données d'entrée et garantir leur qualité pour l'entraînement du modèle.

Ensuite, nous aborderons la construction du modèle, détaillant l'architecture ResNet50 et la manière dont le bloc CBAM améliore la capacité du modèle à se concentrer sur les caractéristiques clés des images médicales. L'entraînement du modèle sera également discuté, en mettant en lumière les techniques d'augmentation des données employées pour optimiser la généralisation et la performance du modèle.

Ce chapitre fournit une vue d'ensemble complète des méthodes et technologies utilisées pour développer un système de classification des tumeurs cérébrales, avec un accent particulier sur les approches innovantes et les bénéfiques pour le diagnostic précoce et précis des pathologies cérébrales.

2 Objectif

L'objectif principal de ce projet est de développer un modèle de classification d'images capable de détecter les tumeurs cérébrales avec une grande précision en utilisant des techniques de deep learning (CNN, L'apprentissage par transfert), nous visons à :

- **Améliorer l'Architecture du Modèle :** Intégrer le module d'attention CBAM à l'architecture ResNet50 pour renforcer la capacité du modèle à se concentrer sur les caractéristiques importantes des images médicales, augmentant ainsi la précision de la détection des tumeurs.
- **Évaluer l'Efficacité du Modèle :** Utiliser des métriques de performance telles que la précision, le rappel, et le F1-score pour évaluer l'efficacité du modèle sur un ensemble de données de test indépendantes.
- **Contribuer à la Recherche Médicale :** Fournir un modèle fiable et performant qui peut potentiellement être utilisé comme outil d'aide au diagnostic pour les professionnels de la santé, facilitant ainsi la détection précoce des tumeurs cérébrales.

En atteignant ces objectifs, ce projet vise à démontrer l'efficacité des techniques d'attention et des modèles de deep learning dans le domaine de la détection des tumeurs cérébrales, tout en apportant des contributions significatives à la recherche et à la pratique médicale.

3 Architecture de système

Dans ce travail, nous proposons une approche pour la détection et la classification des tumeurs cérébrales en utilisant une approche d'apprentissage profond.

Notre système vise à déterminer si une lésion donnée est une tumeur cérébrale, un gliome, un méningiome, une tumeur hypophysaire ou une lésion non tumorale.

L'architecture de notre système de détection des tumeurs cérébrales se compose de deux phases :

1. **La phase prétraitement des données :** Cette phase comprend toutes les étapes nécessaires à la préparation de l'ensemble de données qui sera introduit dans la technique d'apprentissage profond lors de la phase suivante. Le processus comprend la Collectation des Données, Annotations, enrichissement des images, Redimensionnement, Normalisation. . .
2. **Phase de Construction et d'Entraînement du Modèle :** Cette phase est la plus importante se concentre sur la construction de l'architecture du modèle, son entraînement et son évaluation Elle comprend le Choix de l'Architecture de Base et Intégration du Bloc CBAM.

3.1 La phase prétraitement des données

Dans la phase de prétraitement, nous nous assurons que l'entrée est appropriée pour l'architecture proposée en appliquant des étapes de crop et redimensionnement, de normalisation de l'image, enrichissement des images. Ces étapes sont expliquées comme suit :

1. Crop

Traite une image pour enlever les bordures noires et ne conserver que la région du cerveau. Cela permet de focaliser le modèle sur les zones pertinentes de l'image pour améliorer la précision lors de l'entraînement et de la classification des tumeurs cérébrales.

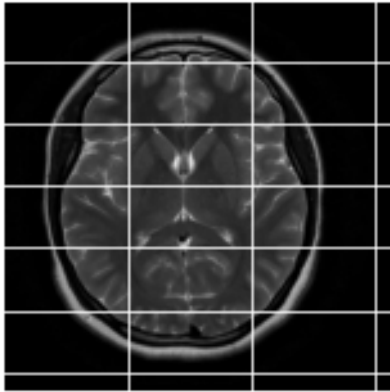


FIGURE 3.1 – Image original

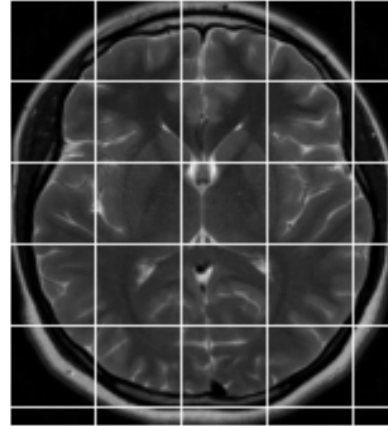


FIGURE 3.2 – Image Cropé

2. Enrichissement des l'images

Cette étape applique le CLAHE (Contrast Limited Adaptive Histogram Equalization) aux images pour améliorer leur contraste. CLAHE est utilisé pour ajuster localement le contraste dans les régions de l'image, ce qui aide à rendre les détails plus visibles, surtout dans des zones avec peu de contraste. Cette amélioration du contraste permet au modèle de mieux distinguer les caractéristiques importantes des images, comme les tumeurs cérébrales, ce qui peut améliorer les performances de la détection.

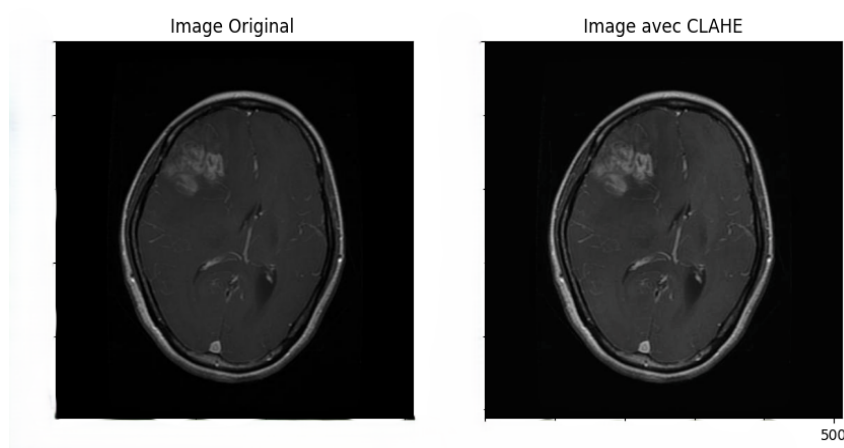


FIGURE 3.3 – Image avec CLAHE

3. La normalisation des images

La normalisation des images consiste à ajuster les valeurs des pixels pour les placer dans une plage spécifique, généralement entre 0 et 1 ou entre -1 et 1. Cette étape est cruciale car elle permet d'assurer que toutes les images ont une échelle de pixel cohérente, ce qui facilite l'apprentissage du modèle. En normalisant les images, on aide le modèle de deep learning à converger plus rapidement et à éviter les biais dus à des variations extrêmes de luminosité ou de contraste entre les images.

4. Redimensionnement

Le redimensionnement des images consiste à ajuster leurs dimensions pour correspondre à une taille spécifique, souvent celle requise par le modèle de deep learning. Cette étape est importante car elle permet de garantir que toutes les images d'entrée ont les mêmes dimensions, ce qui est nécessaire pour que le modèle puisse traiter les images de manière cohérente. Le redimensionnement aide à uniformiser les données tout en maintenant les proportions des objets dans l'image, facilitant ainsi l'entraînement du modèle et améliorant sa performance sur des données de taille variable.

5. Augmentation des données

L'augmentation des données consiste à générer des versions modifiées des images d'entraînement en appliquant diverses transformations, telles que la rotation, le retournement, le zoom, et les modifications de contraste. Cette technique a pour rôle d'enrichir l'ensemble de données sans en collecter de nouvelles, augmentant ainsi la diversité des exemples présentés au modèle.

- **Rotation** : La rotation est une technique d'augmentation des données qui consiste à tourner les images d'un certain angle entre et 359 degree , généralement choisi de manière aléatoire, autour de leur centre. Le rôle de la rotation est de simuler différentes orientations des objets dans les images.

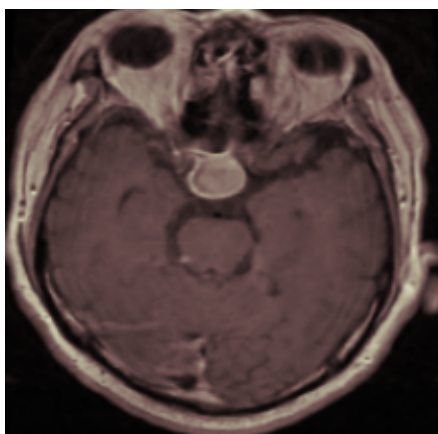


FIGURE 3.4 – Image originale

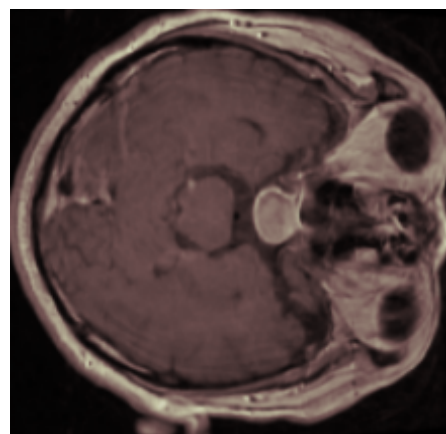


FIGURE 3.5 – Image après le Rotation

- **La translation** : Consiste à déplacer l'intégralité de l'image ou une partie de l'image dans une direction spécifique, soit horizontalement (gauche ou droite),

soit verticalement (haut ou bas). Cette technique modifie la position des objets dans l'image tout en préservant leur structure et contenu. La translation enrichit l'ensemble d'entraînement rendant le modèle moins dépendant de la position fixe des objets dans les images originales.

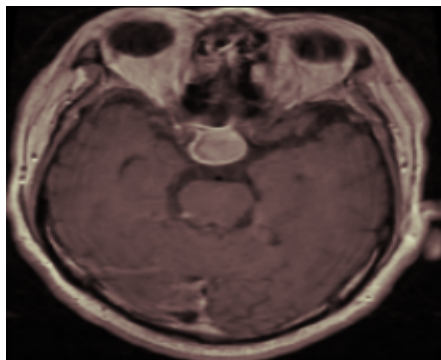


FIGURE 3.6 – Image original

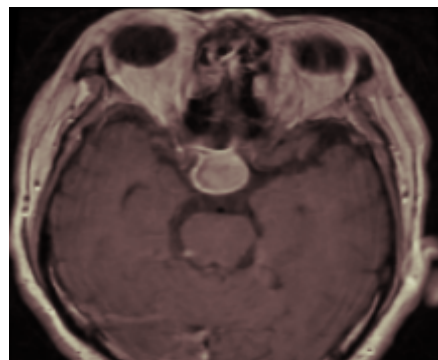


FIGURE 3.7 – Image après decolage de la largeur

- **Retournement de l'image** : consiste à inverser une image, soit horizontalement (flip horizontal), soit verticalement (flip vertical).

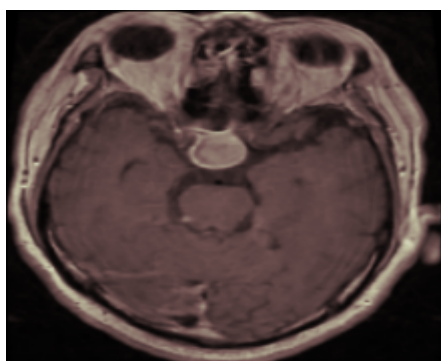


FIGURE 3.8 – Image original

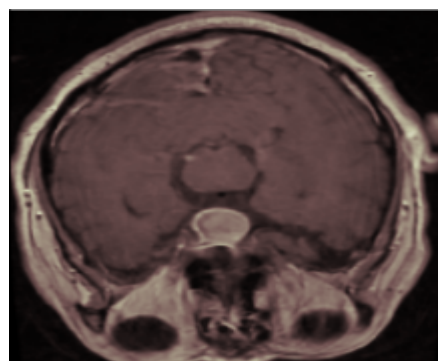


FIGURE 3.9 – Image après un flip horizontal et vertical

3.2 Phase de Construction et d'Entraînement du Modèle

3.2.1 CBAM (Convolutional Block Attention Module)

Le CBAM (Convolutional Block Attention Module) a été introduit par Sanghyun Woo, Jongchan Park, Joon-Young Lee et In So Kweon dans leur article intitulé "**CBAM : Convolutional Block Attention Module**", présenté à l'European Conference on Computer Vision (ECCV) en 2018 [63]. Leurs recherches ont démontré que l'ajout de mécanismes d'attention canalisée et spatiale dans les réseaux convolutifs permettait d'améliorer les performances sur diverses tâches de vision par ordinateur, telles que la classification d'images, la détection d'objets et la segmentation.

Dans le contexte de la détection de tumeur cérébrale, le CBAM joue un rôle crucial en améliorant la capacité du modèle à identifier et se concentrer sur les zones pertinentes des

images cérébrales, ce qui peut conduire à une meilleure précision de classification. Voici comment le CBAM contribue à cette tâche :

- **Amélioration des performances** : En affinant les caractéristiques à la fois au niveau des canaux et des régions spatiales, le CBAM permet au modèle de se concentrer sur les informations pertinentes, améliorant ainsi les performances sur diverses tâches de vision par ordinateur, y compris la classification d'images, la détection d'objets et la segmentation.
- **Modularité et flexibilité** : Le CBAM peut être facilement intégré dans n'importe quel CNN existant sans nécessiter de modifications majeures de l'architecture. Il peut être appliqué à différents niveaux de profondeur dans le réseau pour obtenir des améliorations de performance à plusieurs échelles.

CBAM combine à la fois l'attention spatiale et l'attention par canal, ce qui permet de mieux modéliser les relations complexes dans les données visuelles.

Voici un aperçu des deux principaux composants du CBAM :

1. **Attention par canal (Channel Attention)** :

- Cette partie du module se concentre sur l'importance relative de chaque canal de caractéristiques.
- Pour chaque canal, le module calcule une pondération en utilisant des opérations de global average pooling et de global max pooling, suivies de couches entièrement connectées et d'une fonction d'activation sigmoïde.
- Les pondérations obtenues sont ensuite appliquées aux canaux de caractéristiques pour ajuster leur importance.

2. **Attention spatiale (Spatial Attention)** :

- Cette partie se concentre sur les régions importantes dans l'image.
- Le module génère une carte d'attention spatiale en appliquant des opérations de global max pooling et de global average pooling sur les dimensions des canaux, puis en les concaténant.
- Ensuite, une convolution 2D est appliquée suivie d'une fonction d'activation sigmoïde pour produire la carte d'attention spatiale.
- Cette carte est multipliée aux caractéristiques d'entrée pour ajuster l'importance des différentes régions de l'image.

L'architecture de notre bloc CBAM est illustrée dans la figure suivante

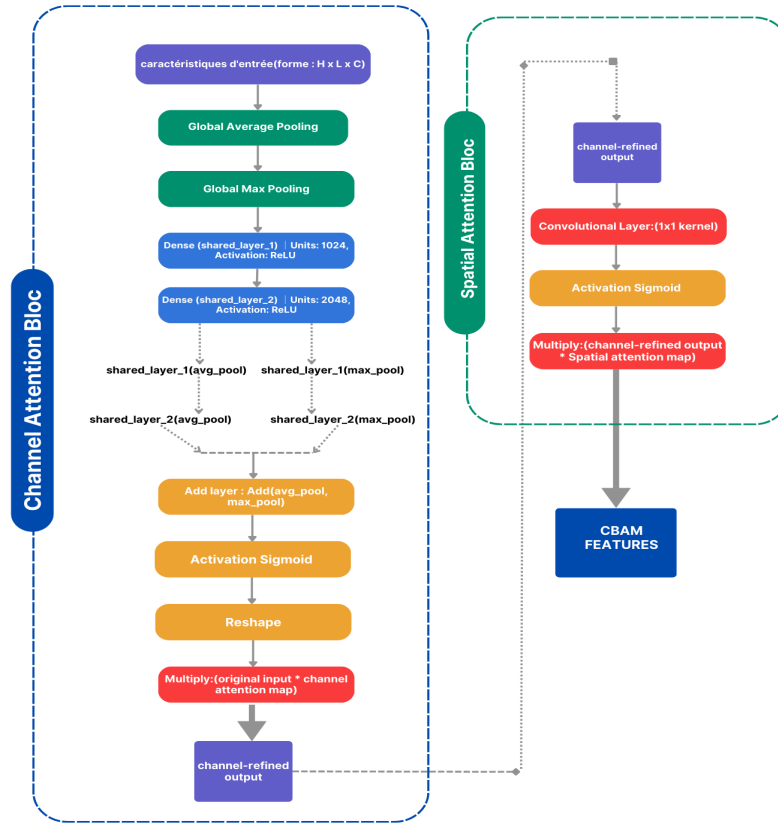


FIGURE 3.10 – L'architecture de bloc CBAM

3.2.2 Description détaillée des couches du bloc CBAM

Le bloc CBAM (Convolutional Block Attention Module) se compose de deux sous-modules : l'attention par canal (Channel Attention Module) et l'attention spatiale (Spatial Attention Module). Chaque sous-module affine les caractéristiques d'entrée en mettant en évidence les informations importantes pour améliorer les performances du modèle. Voici une description détaillée des couches utilisées dans chaque sous-module.

1. **Attention par canal (Channel Attention Module)** L'attention par canal ajuste l'importance des canaux individuels dans la carte de caractéristiques.
 - Global Average Pooling (GAP) : Effectue une moyenne des valeurs sur chaque canal pour obtenir un vecteur de taille égale au nombre de canaux.

$$F_{avg} = \frac{1}{HW} \sum_{i=1}^H \sum_{j=1}^W x_{i,j,c}$$

- Global Max Pooling (GMP) : Prend la valeur maximale sur chaque canal pour obtenir un vecteur de taille égale au nombre de canaux.

$$F_{max} = \max_{i,j} x_{i,j,c}$$

Où \mathbf{H} et \mathbf{W} sont la hauteur et la largeur de la carte de caractéristiques, et $\mathbf{F}(\mathbf{i}, \mathbf{j})$ représente les valeurs de la carte de caractéristiques.

- **Couches Dense partagées** : Les deux vecteurs de pooling passent ensuite par des couches entièrement connectées (dense) partagées pour réduire puis restaurer le nombre de canaux. Cela permet de capturer des relations non linéaires entre les canaux.
- **Fusion et activation sigmoïde** : Les sorties des couches dense sont additionnées, puis passent par une activation sigmoïde pour obtenir les pondérations d'attention par canal.

$$F_{att} = \sigma(F''_{avg} + F''_{max})$$

- **Multiplication** : Les pondérations d'attention par canal sont multipliées aux caractéristiques d'entrée pour ajuster l'importance des canaux.

$$F_{scale} = F F_{att}$$

où \mathbf{F} est la carte de caractéristiques d'entrée.

Résultat : F_{scale} qui est la carte de caractéristiques d'entrée modifiée en fonction de l'importance des canaux. sont ensuite ajustées par l'attention spatiale. Cela se fait en multipliant les caractéristiques modifiées par les pondérations d'attention spatiale calculées pour chaque position dans la carte de caractéristiques.

2. Attention spatiale (Spatial Attention Module) :

L'attention spatiale se concentre sur les régions importantes des caractéristiques d'entrée.

- **Convolution** : une couche de convolution avec un noyau de 1×1 est appliquée pour générer la carte d'attention spatiale. Cette carte est ensuite traitée par une activation sigmoïde pour produire les poids d'attention spatiale finaux.

$$F_{spatial} = \sigma(Conv([F_{spatial}]))$$

où **Conv** est une couche de convolution et σ est la fonction sigmoïde.

- **Multiplication** : La carte d'attention spatiale est multipliée aux caractéristiques ajustées par canal pour ajuster l'importance des régions.

$$F_{final} = F_{scale} F_{spatial}$$

Où **Fscale** est la sortie de l'attention par canal.

Résultat : F_{final} qui est la carte de caractéristiques finale où l'importance des régions a été ajustée. La sortie finale est une carte de caractéristiques optimisée qui contient les informations les plus pertinentes pour la tâche de détection des tumeurs cérébrales, avec une attention améliorée sur les canaux et les régions importantes. Cette carte de caractéristiques ajustée est ensuite utilisée comme entrée pour les couches suivantes du réseau, contribuant à une meilleure performance globale du modèle.

3.3 Architecture de Model

Nous utilisons des modèles pré-entraînés pour notre système de classification des tumeurs cérébrales, qui partagent tous l'architecture ResNet50. Nous avons gardé la partie de base de ResNet50 pour l'extraction des caractéristiques.

En outre, nous avons ajouté nos CBAM bloc et couches supérieures personnalisées afin d'adapter le modèle pré-entraîné à notre tâche spécifique.

Le choix de ResNet50 comme modèle de base pour la classification des tumeurs cérébrales est justifié par sa profondeur, ses capacités d'apprentissage résiduel, ses avantages en matière d'apprentissage par transfert, ses performances avérées, sa modularité, le soutien de la communauté et son efficacité de calcul.

ResNet50 utilise l'apprentissage résiduel par le biais de connexions sautées, ce qui permet d'atténuer le problème du gradient qui s'évanouit. Cela permet au réseau d'entraîner des couches plus profondes plus efficacement, améliorant ainsi l'extraction et la représentation des caractéristiques.

L'architecture de ResNet50, détaillée au chapitre deux, Ces connexions résiduelles présentent sous deux formes principales : la connexion de raccourci d'identité et la connexion de raccourci de projection.. Après avoir enlevé ses couches supérieures et les avoir remplacées par nos couches personnalisées.

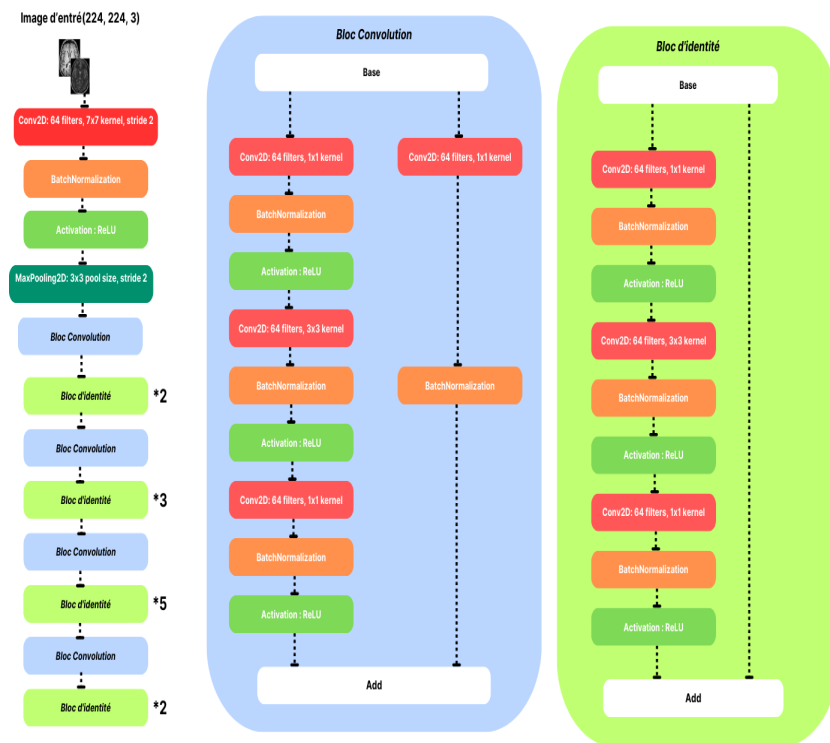


FIGURE 3.11 – Architecture du modèle pré-entraîné ResNet50 utilisé dans notre système.

3.3.1 Couches supérieures personnalisées

Selon l'architecture ResNet50, après les blocs résiduels, il y a une couche de average pooling, suivie d'une couche de sortie avec une fonction d'activation softmax qui correspond à la classification catégorielle d'ImageNet. Ces couches supérieures ont été supprimées et remplacées par notre couches personnalisées. Après les couches de base de ResNet50, nous avons le bloc CBAM qui est expliqué en détail précédemment. Ensuite, nous avons nos couches supérieures personnalisées pour notre tâche de classification des tumeurs cérébrales. Ces couches sont illustrées dans la Figure 3.8, et expliquées dans l'ordre suivant :



FIGURE 3.12 – l'architecture général de système

1. **Global Average Pooling (GAP)** Une couche Global Average Pooling réduit les dimensions spatiales des cartes de caractéristiques d'entrée en calculant la valeur moyenne de chaque canal de la carte de caractéristiques. Cette opération réduit considérablement le nombre de paramètres dans le modèle et permet d'éviter l'ajustement excessif. En calculant la moyenne de chaque carte de caractéristiques, elle crée une statistique récapitulative qui conserve les informations globales des cartes de caractéristiques. La couche GAP traite les cartes de caractéristiques de la base ResNet50, en les transformant de la forme (H, W, C) en (C) , où H est la hauteur, W est la largeur et C est le nombre de canaux. Pour une carte de caractéristiques d'entrée $\mathbf{F}(H,W,C)$:

$$GAP(F) = \frac{1}{HW} \sum_{i=1}^H \sum_{j=1}^W F_{i,j,k} \quad for \quad k = 1, 2, 3, \dots, C$$

2. **Couches denses avec activation de la ReLU** Les couches denses, également connues sous le nom de couches entièrement connectées, sont utilisées pour apprendre des modèles complexes dans les données en combinant les caractéristiques extraites par les couches précédentes. La fonction d'activation ReLU (Rectified Linear Unit) introduit la non-linéarité, ce qui permet au réseau d'apprendre des fonctions plus complexes. La séquence de couches denses réduit progressivement le nombre d'unités, canalisant l'information dans une représentation plus compacte.

$$y = \text{ReLU}(W \cdot x + b)$$

où \mathbf{x} est l'entrée, \mathbf{W} est la matrice de poids, \mathbf{b} est le vecteur de biais, et $\text{ReLU}(z) = \max(0, z)$.

3. **Couche Dropout** Une couche d'exclusion est une technique de régularisation qui met aléatoirement à zéro une fraction des unités d'entrée pendant la formation. Cela empêche le modèle de devenir trop dépendant de neurones spécifiques, réduisant ainsi l'ajustement excessif et améliorant la généralisation. Au cours de chaque étape de formation, la couche Dropout fait "tomber" (met à zéro) de manière aléatoire 50 % des unités de la sortie de la couche précédente. Ce caractère aléatoire oblige le modèle à apprendre des représentations redondantes, ce qui améliore sa robustesse.

$$y = \begin{cases} 0 & \text{with probability } p \\ \frac{x_i}{1-p} & \text{otherwise} \end{cases}$$

4. **Couche Reshape** Une couche Reshape modifie la forme du tenseur d'entrée pour qu'il corresponde aux dimensions d'entrée attendues pour les couches suivantes. Cette fonction est particulièrement utile pour passer d'un type de couche à un autre, par exemple d'une couche dense à une couche convolutive ou à une couche d'attention.
5. **Global Max Pooling and Global Average Pooling** Les couches Global Max Pooling et Global Average Pooling sont toutes deux utilisées pour résumer les informations contenues dans le tenseur d'entrée. La mise en commun maximale capture les caractéristiques les plus importantes en sélectionnant la valeur maximale, tandis que la mise en commun moyenne calcule la valeur moyenne, fournissant une statistique récapitulative.
6. **Concatenate Layer** Une couche Concaténer fusionne les sorties de plusieurs couches le long d'un axe spécifié. Cette opération permet de combiner différents types de caractéristiques en un seul tenseur. La couche Concaténer fusionne les résultats de la mise en commun maximale globale et de la mise en commun moyenne globale le long de la dimension de la caractéristique, ce qui donne un tenseur ayant la forme suivante (256). Cela permet de combiner les points forts des deux méthodes de mise en commun.
7. **Couche de sortie dense avec activation sigmoïde** La couche dense finale produit une probabilité pour chaque classe à l'aide de la fonction d'activation sigmoïde. Elle convient aux tâches de classification multi-classes où chaque classe s'exclut mutuellement. La couche dense prend les caractéristiques concaténées et

applique une transformation linéaire suivie d'une activation sigmoïde, produisant une probabilité pour chacune des 4 classes.

$$y = \sigma(W \cdot x + b)$$

où $\sigma(z) = \frac{1}{1+\exp^{-z}}$ est la fonction sigmoïde, \mathbf{x} est l'entrée \mathbf{W} est la matrice de poids, et \mathbf{b} est le vecteur de biais. La forme de la sortie est (4).

Conclusion

Dans ce chapitre, nous avons détaillé le processus de conception de la solution proposée pour répondre aux problématiques identifiées dans les chapitres précédents. Nous avons commencé par un processus rigoureux de prétraitement des données, y compris le recadrage des images, la normalisation, et l'augmentation des données, nous avons préparé un ensemble de données diversifié et de haute qualité qui a permis d'améliorer les performances du modèle. L'intégration de CBAM a permis d'ajouter un mécanisme d'attention, améliorant la capacité du modèle à se concentrer sur les caractéristiques les plus importantes des images, ce qui s'est traduit par une meilleure précision dans la classification des tumeurs.

Chapter 4

Implementation et Realisation

Implementation et Realisation

Introduction

Le présent chapitre est consacré à l'implémentation et à la réalisation du système de détection des tumeurs cérébrales, basé sur l'architecture ResNet50 concaténé avec Convolutional Block Attention Module (CBAM). Après avoir détaillé les aspects théoriques et méthodologiques dans les chapitres précédents, nous abordons ici les aspects pratiques du projet, en décrivant les étapes concrètes qui ont permis de transformer les concepts théoriques en un modèle opérationnel.

Nous commencerons par une présentation de l'environnement de développement, incluant les outils, les langages de programmation, et les configurations matérielles utilisés pour l'implémentation. Ensuite, nous détaillerons les étapes de prétraitement des données, cruciales pour préparer les images cérébrales à l'entraînement du modèle, en appliquant des techniques telles que le recadrage, la normalisation, et l'augmentation des données.

Par la suite, nous décrirons la construction du modèle, en expliquant comment l'architecture ResNet50 a été adaptée et enrichie avec le bloc CBAM pour améliorer la capacité du modèle à détecter et à classifier les tumeurs cérébrales. L'entraînement du modèle sera également couvert, avec une attention particulière portée à la sélection des hyperparamètres, au suivi de l'entraînement.

Enfin, nous présenterons les résultats obtenus lors de la phase de test et de validation.

2 Environnement de Développement

2.1 Outils et Langages Utilisés

- **Python** : Python est un langage de programmation interprété, polyvalent et de haut niveau, créé par Guido van Rossum et publié pour la première fois en 1991. Il est connu pour sa syntaxe claire et lisible, qui permet aux développeurs de créer des logiciels plus rapidement et avec moins de lignes de code. Python prend en charge plusieurs paradigmes de programmation, y compris la programmation procédurale, orientée objet et fonctionnelle. Il est largement utilisé dans des domaines tels que le

développement web, l'automatisation, l'analyse de données, l'intelligence artificielle et le développement de logiciels[25].

- **TensorFlow** : TensorFlow est une bibliothèque open-source développée par Google pour le calcul numérique et l'apprentissage automatique. Lancée en 2015, elle est largement utilisée pour créer et entraîner des modèles de deep learning. TensorFlow offre des fonctionnalités avancées pour le traitement de données massives et le calcul sur des réseaux de neurones, permettant aux chercheurs et développeurs de construire des modèles d'intelligence artificielle complexes. Il est conçu pour être extensible, fonctionnant sur des plateformes variées, allant des appareils mobiles aux clusters de serveurs distribués[1].
- **Keras** : Keras est une bibliothèque open-source de haut niveau pour le développement de modèles d'apprentissage profond. Initialement conçue pour être simple et rapide à utiliser, Keras fournit une interface utilisateur intuitive pour construire et entraîner des réseaux de neurones, en s'appuyant sur des bibliothèques comme TensorFlow ou Theano pour exécuter le calcul sous-jacent. Keras est particulièrement apprécié pour sa modularité, sa capacité à prototyper rapidement des modèles complexes, et son support pour une vaste gamme de couches, de fonctions de perte, et d'algorithmes d'optimisation[15].
- **Pandas** : est une bibliothèque open-source pour le langage Python, principalement utilisée pour la manipulation et l'analyse de données. Elle offre des structures de données faciles à utiliser, telles que les DataFrames, qui permettent de manipuler des données tabulaires de manière intuitive et efficace. Pandas est particulièrement apprécié pour ses fonctionnalités puissantes de nettoyage, de transformation, et d'analyse des données, ce qui en fait un outil indispensable pour les scientifiques de données et les analystes[62].
- **Numpy** : est une bibliothèque open-source pour Python, qui fournit un support pour les tableaux et matrices multidimensionnels, ainsi qu'une large collection de fonctions mathématiques pour opérer sur ces structures de données. NumPy est essentiel pour les calculs scientifiques et les traitements numériques en Python, car il permet d'effectuer des opérations mathématiques et algébriques complexes de manière rapide et efficace, notamment grâce à son utilisation d'opérations vectorisées[39].
- **OpenCv** : (Open Source Computer Vision Library) est une bibliothèque open-source conçue pour le traitement d'images et la vision par ordinateur. Développée initialement par Intel, OpenCV offre une vaste gamme d'algorithmes pour la manipulation d'images, la reconnaissance d'objets, la détection de mouvements, et bien plus encore. Elle est largement utilisée dans des domaines tels que la robotique, la réalité augmentée, et l'analyse vidéo, grâce à sa rapidité et à sa compatibilité avec plusieurs langages de programmation, dont Python, C++, et Java[9].
- **Scikit-learn** : (ou sklearn) est une bibliothèque open-source pour Python, spécialisée dans l'apprentissage automatique et l'analyse de données. Elle fournit des outils simples et efficaces pour des tâches courantes telles que la classification, la régression, le clustering, et la réduction de dimension. Scikit-learn est conçue pour être facile à utiliser tout en offrant des fonctionnalités puissantes pour le développement de modèles de machine learning, grâce à une interface cohérente et bien documentée[70].

- **Matplotlib** : Matplotlib est une bibliothèque open-source pour Python qui permet de créer des visualisations statiques, animées et interactives de données. Elle offre une grande flexibilité pour générer des graphiques, des histogrammes, des diagrammes en dispersion, des courbes et d'autres types de visualisations, en utilisant une interface orientée objet ou des commandes similaires à celles de MATLAB. Matplotlib est largement utilisé pour l'analyse de données, la visualisation scientifique, et la présentation de résultats.

2.2 Configuration Matérielle

Pour la mise en œuvre et l'entraînement du modèle de détection des tumeurs cérébrales, une configuration matérielle robuste a été essentielle pour gérer les exigences computationnelles élevées. La configuration matérielle utilisée est la suivante :

- **Mémoire Vive (RAM)** : 256 Go
 - **Rôle** : Une capacité élevée de RAM est cruciale pour le traitement efficace des grandes quantités de données d'image, permettant un chargement rapide et un traitement simultané de plusieurs images pendant l'entraînement du modèle. La RAM ample permet également de maintenir plusieurs processus en parallèle, réduisant ainsi les temps de latence et améliorant la fluidité des opérations.
- **Stockage** : 450 Go
 - **Rôle** : Le stockage de 450 Go a été utilisé pour conserver les ensembles de données, les modèles entraînés, et les résultats intermédiaires. Un espace de stockage suffisant est nécessaire pour gérer les fichiers volumineux associés aux images médicales ainsi que les checkpoints et les sauvegardes du modèle, garantissant que toutes les données importantes sont sécurisées et facilement accessibles.
- **Processeur (CPU)** : 32 cœurs à 3.1 GHz
 - **Rôle** : Le processeur avec 32 cœurs à une vitesse de 3.1 GHz permet un traitement parallèle efficace des tâches. Cette puissance de calcul est particulièrement bénéfique pour les opérations intensives en calcul telles que l'entraînement du modèle et le traitement des images, réduisant les temps d'exécution et améliorant la performance globale du système.

2.3 Environnements Logiciels

Pour l'implémentation et la gestion du projet de détection des tumeurs cérébrales, un environnement cloud spécifique a été utilisé pour optimiser les performances et faciliter le développement. L'environnement logiciel sélectionné est **Lightning AI Studio** , qui offre des capacités avancées adaptées aux exigences des projets de deep learning.

1. Lightning AI Studio :

- **Description** : Lightning AI Studio est une plateforme cloud dédiée au développement et à l'entraînement de modèles d'intelligence artificielle. Elle fournit une infrastructure flexible et évolutive pour le traitement des données et l'exécution des algorithmes de machine learning.
- **Rôle** : Cet environnement permet une gestion efficace des ressources de calcul, telles que les GPU, et fournit des outils pour le développement, le suivi et

la gestion des expériences. Lightning AI Studio facilite la collaboration et le partage de résultats en offrant des fonctionnalités de gestion des versions et de visualisation des performances du modèle.

2. Outils et Bibliothèques :

- **Frameworks :** Utilisation de frameworks de deep learning comme TensorFlow ou PyTorch pour la construction et l’entraînement du modèle. Ces frameworks sont intégrés à Lightning AI Studio, permettant une gestion fluide des modèles et des processus d’entraînement.
- **Outils de Développement :** L’environnement fournit des outils pour le codage, le suivi des expérimentations, et la visualisation des résultats, simplifiant ainsi le processus de développement et de débogage.

3. Avantages de l’Environnement Cloud :

- **Scalabilité :** L’accès à des ressources de calcul à la demande permet de s’adapter aux besoins variés du projet sans les contraintes de l’infrastructure locale.
- **Accessibilité :** Permet le travail collaboratif et l’accès aux ressources depuis différents endroits, facilitant ainsi le développement et la gestion du projet en équipe.
- **Optimisation des Performances :** L’utilisation de GPU et d’autres accélérateurs matériels optimisés pour le deep learning assure des temps de traitement réduits et une gestion efficace des ressources pour les tâches intensives en calcul. L’utilisation de Lightning AI Studio a permis de tirer parti des avantages d’une plateforme cloud spécialisée, facilitant ainsi la réalisation du projet tout en offrant une infrastructure solide et adaptée aux exigences du deep learning.

2.4 Outils d’évaluation des performances

- Matrice de Confusion

La matrice de confusion est l’une des estimations utilisées pour évaluer la performance d’un modèle. Elle indique la relation entre les valeurs prédites et les valeurs réelles.

Vrai positif (TP) : patients diagnostiqués positifs et qui le sont réellement.

Vrai négatif (VN) :- patients diagnostiqués négatifs et réellement négatifs.

Faux positif (FP) : patients diagnostiqués positifs mais réellement négatifs.

Faux négatif (FN) : patients diagnostiqués négatifs et qui sont réellement positifs

	Prédiction Positive	Prédiction Négative
Réel Positif	Vrais Positifs (TP)	Faux Négatifs (FN)
Réel Négatif	Faux Positifs (FP)	Vrais Négatifs (TN)

- Exactitude (Accuracy)

L’exactitude est le pourcentage de prédictions correctes parmi le nombre total de prédictions.

$$Exactitude = \frac{\text{Nombre de prédictions correctes}}{\text{Nombre total de prédictions}}$$

- Précision (Precision)

La précision est le pourcentage de prédictions positives correctes parmi toutes les prédictions positives.

$$\text{Précision} = \frac{\text{Vraispositifs}(TP)}{\text{Vraispositifs}(TP) + \text{Fauxpositifs}(FP)}$$

- Rappel (Recall) ou Sensibilité (Sensitivity)

Le rappel est le pourcentage de prédictions positives correctes parmi toutes les instances positives réelles.

$$\text{Rappel} = \frac{\text{Vraispositifs}(TP)}{\text{Vraispositifs}(TP) + \text{Fauxnégatives}(FN)}$$

- F-mesure (F1-score)

La F-mesure est la moyenne harmonique de la précision et du rappel. Elle est utile lorsque vous avez besoin d'un équilibre entre les deux.

$$F1 = 2 \frac{\text{PrécisionRappel}}{\text{Précision} + \text{Rappel}}$$

- AUC - ROC (Area Under the Receiver Operating Characteristic Curve)

L'AUC - ROC mesure la capacité du modèle à distinguer entre les classes. Plus l'AUC est élevé, meilleure est la performance du modèle.

AUC – ROC=Aire sous la courbe ROC, qui trace le taux de vrais positifs contre le taux de faux positifs

3 Collecte de données

L'ensemble de données utilisé dans cette recherche provient de la plateforme Kaggle et est constitué d'images d'IRM cérébrales. Ce dataset est un ensemble de données public et contient un total de 7 023 images classifiées en quatre catégories distinctes. Les catégories incluent :

- **Gliome :**

Images correspondant à des tumeurs cérébrales de type gliome. Les gliomes sont des tumeurs qui se forment dans les cellules gliales du cerveau et peuvent varier en termes de gravité.

- **Méningiome :**

Images représentant des tumeurs de type méningiome. Les méningiomes sont des tumeurs généralement bénignes qui se développent à partir des méninges, les membranes qui entourent le cerveau et la moelle épinière.

- **Aucune Tumeur :**

Images qui ne montrent aucune anomalie tumorale. Cette classe sert de base pour la détection des images normales, sans pathologie.

- **Hypophyse :**

Images associées à des anomalies de l'hypophyse, une glande située à la base du cerveau, qui peut parfois présenter des pathologies détectables par IRM.

Le Tableau ci-dessous résume la distribution des images au sein de chaque classe :

Classe	Entrainement	Test	Totale
Glioma	1321	300	1621
Meningioma	1339	306	1645
NoTumeur	1595	405	2000
Pituitary	1457	300	1757

TABLE 4.1 – la distribution des images

Le graphique ci-dessous illustre la distribution des images dans les ensembles d'entraînement et de test pour chaque classe du dataset :

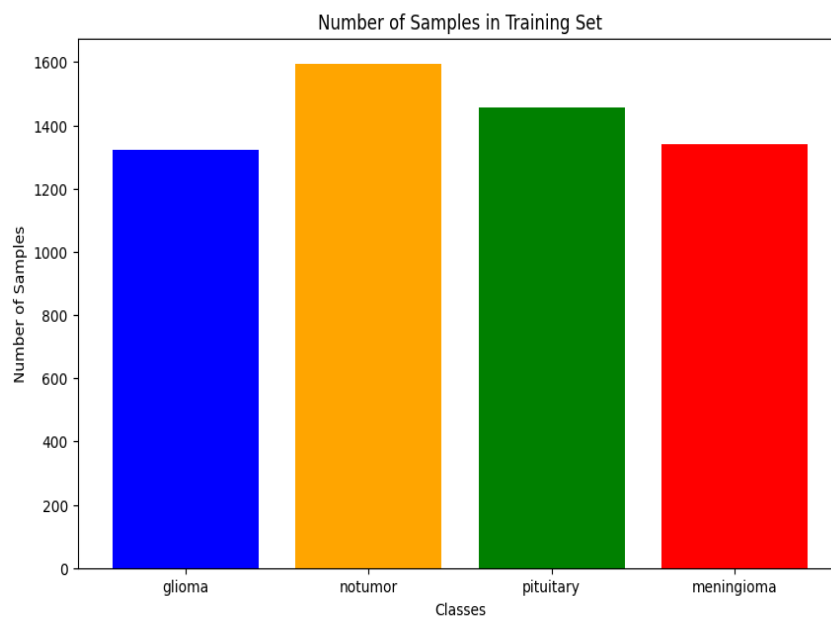


FIGURE 4.1 – Nombre d'Échantillons par Classe dans l'Ensemble d'Entraînement

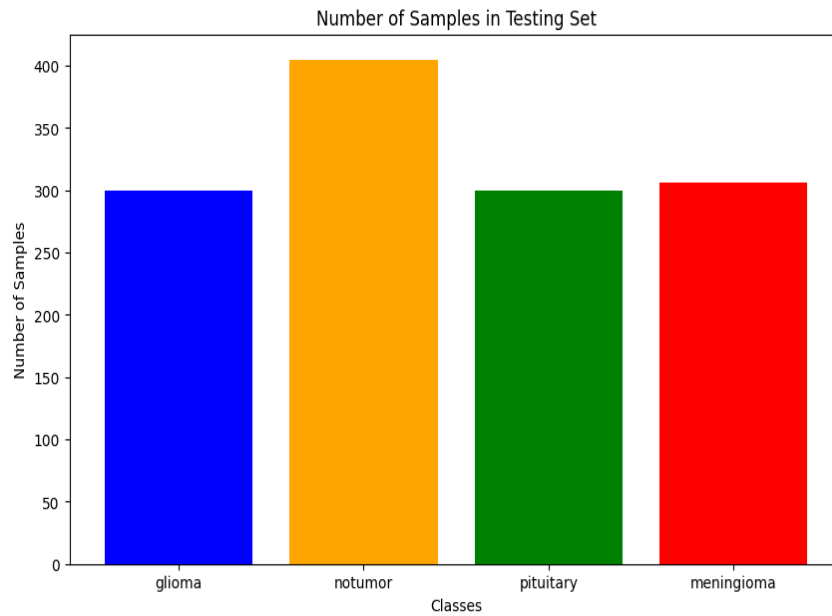


FIGURE 4.2 – Nombre d’Échantillons par Classe dans l’Ensemble d’Entraînement

Ce graphique permet de vérifier la représentativité et la diversité des données dans les ensembles d’entraînement et de test, assurant ainsi une évaluation fiable du modèle développé.

4 Implementation

La fonction `extBot = tuple(c[c[:, :, 1].argmax()][0])` extrait la coordonnée inférieure (la plus basse en y) d’une région d’intérêt définie par un contour `c`. Le tableau `c[:, :, 1]` sélectionne toutes les valeurs y de ce contour, et `argmax()` trouve l’indice de la valeur maximale, indiquant ainsi la position la plus basse. Le tuple formé contient les coordonnées (x, y) du point le plus bas, stockées dans `extBot`. Ensuite, l’image est recadrée en utilisant ces coordonnées extrêmes (`extTop`, `extBot`, `extLeft`, `extRight`) pour créer une nouvelle image `new_image`, qui contient uniquement la région d’intérêt. Cette sous-image est définie par les indices `[extTop[1] :extBot[1], extLeft[0] :extRight[0]]` sur les axes y et x respectivement, permettant de concentrer l’image sur la zone d’intérêt, comme un cerveau.

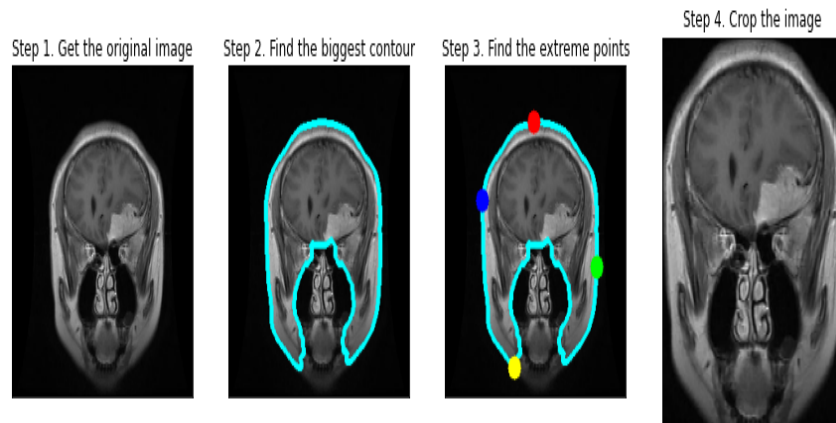


FIGURE 4.3 – Photo encadrée en utilisant les coordonnées extrêmes

La fonction `process_images_in_directory(directory_path)` parcourt ensuite tous les fichiers d'un répertoire donné, charge chaque image, la recadre avec une fonction `crop_brain()` similaire, puis sauvegarde l'image recadrée en écrasant l'image d'origine.

```

def crop_brain(image):
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    gray = cv2.GaussianBlur(gray, (5, 5), 0)

    thres = cv2.threshold(gray, 45, 255, cv2.THRESH_BINARY)[1]
    thres = cv2.erode(thres, None, iterations=2)
    thres = cv2.dilate(thres, None, iterations=2)

    cnts = cv2.findContours(thres.copy(), cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
    cnts = imutils.grab_contours(cnts)
    if len(cnts) == 0:
        return image # return the original image if no contours are found

    c = max(cnts, key=cv2.contourArea)

    extLeft = tuple(c[c[:, :, 0].argmin()][0])
    extRight = tuple(c[c[:, :, 0].argmax()][0])
    extTop = tuple(c[c[:, :, 1].argmin()][0])
    extBot = tuple(c[c[:, :, 1].argmax()][0])

    new_image = image[extTop[1]:extBot[1], extLeft[0]:extRight[0]]
    return new_image

def process_images_in_directory(directory_path):
    for filename in os.listdir(directory_path):
        if filename.lower().endswith(('.png', '.jpg', '.jpeg', '.bmp', '.tif', '.tiff')):
            file_path = os.path.join(directory_path, filename)
            image = cv2.imread(file_path)
            if image is not None:
                cropped_image = crop_brain(image)
                cv2.imwrite(file_path, cropped_image)
                print(f"Processed and saved: {filename}")
            else:
                print(f"Failed to read image: {filename}")
    
```

FIGURE 4.4 – Nombre d'Échantillons par Classe dans l'Ensemble d'Entraînement

fonction `apply_clahe_to_image(image_path)` qui applique l'algorithme CLAHE (Contrast Limited Adaptive Histogram Equalization) à une image spécifiée par `image_path`. L'image est d'abord chargée en mémoire avec `cv2.imread`. Si l'image est en niveaux de gris (grayscale) ou n'a qu'un seul canal de couleur, CLAHE est appliqué directement sur cette

image. Sinon, l'image est convertie dans l'espace colorimétrique LAB, où CLAHE est appliqué uniquement au canal L (luminosité). Les canaux A et B (couleur) sont ensuite recombinaés avec le canal L modifié pour reformer l'image, qui est reconverte dans l'espace colorimétrique BGR (Bleu, Vert, Rouge). Enfin, l'image traitée est sauvegardée en écrasant l'image d'origine.

```
import os
import cv2
def apply_clahe_to_image(image_path):
    # Load the image
    img = cv2.imread(image_path, cv2.IMREAD_COLOR)
    if img is None:
        print(f"Failed to read image: {image_path}")
        return
    # Check if the image is already grayscale
    if len(img.shape) == 2 or img.shape[2] == 1:
        gray = img
        # Apply CLAHE directly to the grayscale image
        clahe = cv2.createCLAHE(clipLimit=0.5, tileGridSize=(8, 8))
        cl = clahe.apply(gray)
        # Convert back to BGR to save the image in color
        enhanced_img = cv2.cvtColor(cl, cv2.COLOR_GRAY2BGR)
    else:
        # Convert to LAB color space
        lab = cv2.cvtColor(img, cv2.COLOR_BGR2LAB)
        # Split the LAB image into L, A, and B channels
        l, a, b = cv2.split(lab)
        # Apply CLAHE to the L channel
        clahe = cv2.createCLAHE(clipLimit=0.5, tileGridSize=(8, 8))
        cl = clahe.apply(l)
        # Merge the CLAHE enhanced L channel back with A and B channels
        limg = cv2.merge((cl, a, b))
        # Convert back to BGR color space
        enhanced_img = cv2.cvtColor(limg, cv2.COLOR_LAB2BGR)
    # Save the enhanced image (replacing the original image)
    cv2.imwrite(image_path, enhanced_img)
    print(f"Processed and saved CLAHE image: {image_path}")
def process_images_with_clahe(directory_path):
    for filename in os.listdir(directory_path):
        if filename.lower().endswith(('.png', '.jpg', '.jpeg', '.bmp', '.tif', '.tiff')):
            file_path = os.path.join(directory_path, filename)
            apply_clahe_to_image(file_path)
```

FIGURE 4.5 – Application de CLAHE sur les images]

La fonction `process_images_with_clahe(directory_path)` parcourt ensuite tous les fichiers d'un répertoire spécifié. Pour chaque image trouvée avec une extension valide (comme `.png`, `.jpg`, etc.) , elle applique le traitement CLAHE en appelant `apply_clahe_to_image`, ce qui améliore le contraste de chaque image dans le répertoire.

Ce code prépare les données d'entraînement et de test pour un modèle de classification de tumeurs cérébrales en quatre catégories : glioma, meningioma, notumor, et pituitary, définies dans la liste `labels`. Les images d'entraînement et de test sont chargées à partir des répertoires `/ teamspace /studios / this studio /projet _v13 / Training` et `/ teamspace / studios/ this_studio / projet _v13 / Testing` respectivement .

Chaque image est lue en niveaux de gris avec `cv2.imread`, débruitée avec `cv2.bilateralFilter`, puis transformée en une image pseudocolorée avec `cv2.applyColorMap` en utilisant la colormap `COLORMAP_BONE`. Ensuite, les images sont redimensionnées à 224x224 pixels avec `cv2.resize` avant d'être ajoutées aux listes `x_train` ou `x_test`. Les étiquettes correspondantes, basées sur l'index de chaque catégorie dans `labels`, sont ajoutées aux listes `y_train` et `y_test`. Finalement, les images sont normalisées en divisant par 255.0 pour obtenir des valeurs entre 0 et 1, et les dimensions de `x_train` et `x_test` sont affichées pour vérifier le traitement.

```
labels = ['glioma', 'meningioma', 'notumor', 'pituitary']
x_train = [] # training images.
y_train = [] # training labels.
x_test = [] # testing images.
y_test = [] # testing labels.
image_size = 224
for label in labels:
    trainPath = os.path.join('/teamspace/studios/this_studio/projet_v13/Training',label)
    for file in tqdm(os.listdir(trainPath)):
        image = cv2.imread(os.path.join(trainPath, file),0) # load images in gray.
        image = cv2.bilateralFilter(image, 2, 50, 50) # remove images noise.
        image = cv2.applyColorMap(image, cv2.COLORMAP_BONE) # produce a pseudocolored image.
        image = cv2.resize(image, (image_size, image_size)) # resize images into 150*150.
        x_train.append(image)
        y_train.append(labels.index(label))
    testPath = os.path.join('/teamspace/studios/this_studio/projet_v13/Testing',label)
    for file in tqdm(os.listdir(testPath)):
        image = cv2.imread(os.path.join(testPath, file),0)
        image = cv2.bilateralFilter(image, 2, 50, 50)
        image = cv2.applyColorMap(image, cv2.COLORMAP_BONE)
        image = cv2.resize(image, (image_size, image_size))
        x_test.append(image)
        y_test.append(labels.index(label))]
x_train = np.array(x_train) / 255.0 # normalize Images into range 0 to 1.
x_test = np.array(x_test) / 255.0

print(x_train.shape)
print(x_test.shape)
```

FIGURE 4.6 – Le redimensionnement et les étiquetages et la normalisation des données

Ce code prépare les données d'entraînement et de test pour un modèle de deep learning en suivant plusieurs étapes. Tout d'abord, les ensembles `x_train` (images d'entraînement) et `y_train` (étiquettes d'entraînement) sont mélangés de manière synchronisée à l'aide de la fonction `shuffle` avec un état aléatoire fixe (`random_state=42`), assurant ainsi que les images et leurs étiquettes restent correctement alignées tout en garantissant la reproductibilité. Ensuite, les étiquettes `y_train` et `y_test` sont transformées en vecteurs one-hot avec `tensorflow.keras.utils.to_categorical`, où chaque étiquette est convertie en un vecteur binaire où une seule position est activée, correspondant à la classe. Les données d'entraînement sont ensuite divisées en ensembles d'entraînement et de validation avec `train_test_split`, réservant 20% des données pour la validation (`test_size=0.2`), et le mélange est également reproductible grâce à `random_state=42`. Enfin, la forme de l'ensemble de validation `x_val` est affichée pour vérifier la taille et le nombre d'images après la division.

```
x_train, y_train = shuffle(x_train,y_train, random_state=42)

y_train = tensorflow.keras.utils.to_categorical(y_train) #One Hot Encoding on the labels
y_test = tensorflow.keras.utils.to_categorical(y_test)

x_train, x_val, y_train, y_val = train_test_split(x_train, y_train, test_size=0.2, random_state=42)

print(x_val.shape)
```

FIGURE 4.7 – Préparation des Données d’Entraînement et de Test pour le Modèle , One-Hot Encoding

Le code initialise un générateur d’images ImageDataGenerator pour appliquer des augmentations de données sur les images d’entraînement `x_train`. Le générateur est configuré pour effectuer les transformations suivantes : une rotation des images allant jusqu’à 10 degrés (`rotation_range=10`), un déplacement horizontal et vertical des images jusqu’à 5% de la largeur (`width_shift_range=0.05`) et de la hauteur (`height_shift_range=0.05`), ainsi qu’un retournement horizontal des images (`horizontal_flip=True`). Après avoir configuré ces paramètres, le générateur est ajusté aux données d’entraînement en appelant `datagen.fit(x_train)`, ce qui lui permet de calculer les statistiques nécessaires pour appliquer les augmentations de manière appropriée aux images lors de l’entraînement du modèle.

```
# ImageDataGenerator transforms each image in the batch by a series of random translations, rotations, etc.
datagen = ImageDataGenerator(
    rotation_range=10,
    width_shift_range=0.05,
    height_shift_range=0.05,
    horizontal_flip=True)

# After you have created and configured your ImageDataGenerator, you must fit it on your data.
datagen.fit(x_train)
```

FIGURE 4.8 – Génération et Augmentation des Données d’Entraînement avec ImageData-Generator

5 Entraînement et Évaluation des Performances

5.1 Entraînement

Le code compile et entraîne un modèle de deep learning avec les paramètres spécifiés. Le modèle est compilé en utilisant l’optimiseur Adam avec un taux d’apprentissage de 0.0001, la fonction de perte `categorical_crossentropy`, et l’évaluation de la précision (`accuracy`)

```

model = ResNet50_CBAM()

#compile our model.
adam = Adam(learning_rate=0.0001)
model.compile(optimizer=adam, loss = 'categorical_crossentropy', metrics=['accuracy'])
model.summary()

```

FIGURE 4.9 – Difinition des callbacks

Ensuite, deux callbacks sont configurés pour le processus d’entraînement :

```

callbacks = [ModelCheckpoint('.mdl_wts.hdf5', monitor='val_loss',
                             ,mode='min',verbose=1, save_best_only=True),
             ReduceLROnPlateau(monitor='val_loss', factor=0.3, patience=2,
                                verbose=1, mode='min', min_lr=0.0000000001)]

```

FIGURE 4.10 – Compilation du Modèle

Le modèle est ensuite entraîné en utilisant le générateur de données datagen.flow pour les images d’entraînement (x_train, y_train) avec une taille de lot de 64, une validation effectuée sur x_val et y_val, pour 25 époques. Les callbacks configurés sont utilisés pour surveiller la performance du modèle et ajuster les paramètres d’entraînement en consequence.

```

history = model.fit(datagen.flow(x_train, y_train, batch_size=64)
                    , validation_data = (x_val,y_val), epochs = 50, callbacks = callbacks)

```

FIGURE 4.11 – Entraînement du Modèle sur 25 Époques Surveillance par Callbacks

Voici un tableaux de les détails spécifiques pour les paramètres d’entraînement de notre modèle :

TABLE 4.2 – Paramètres du Modèle

Paramètre	Valeur
Taille du filtre	3x3 ou 1x1
Dropout Rate	0.5
Fonction de perte	categorical_crossentropy
Forme de l'entrée	(224, 224, 3)
Optimiseur	Adam
Taux d'apprentissage	0.0001
Epochs	25
Métriques	accuracy
Callbacks	ModelCheckpoint, ReduceLROnPlateau

5.2 Résultats et discussions

Expérimentation 1 : évaluation des performances sur l'architecture ResNet50 Sans CBAM

Dans cette première expérimentation, nous avons évalué les performances de l'architecture ResNet50 sans l'intégration du Convolutional Block Attention Module (CBAM). Voici une description détaillée des résultats obtenus . . La La figure montre les courbes d'apprentissage pour le modèle



FIGURE 4.12 – graphique de accuracy et loss de l'entraînement et de la validation

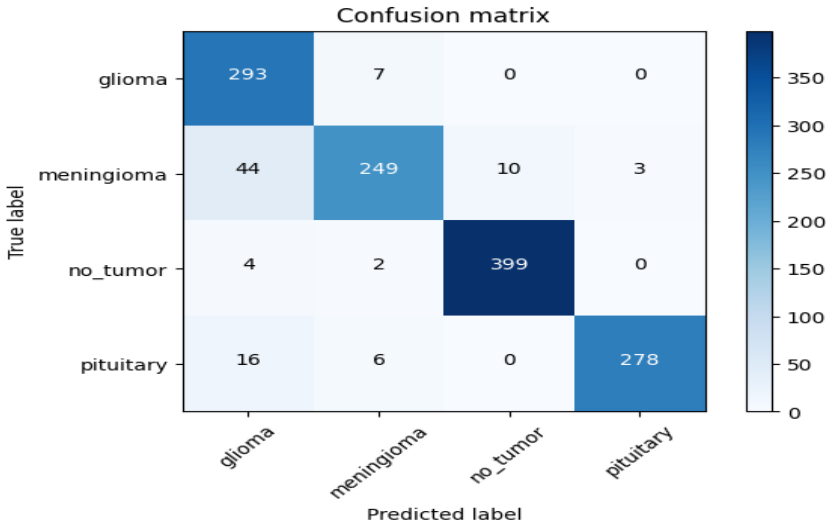


FIGURE 4.13 – la matrice de confusion de dataset de test

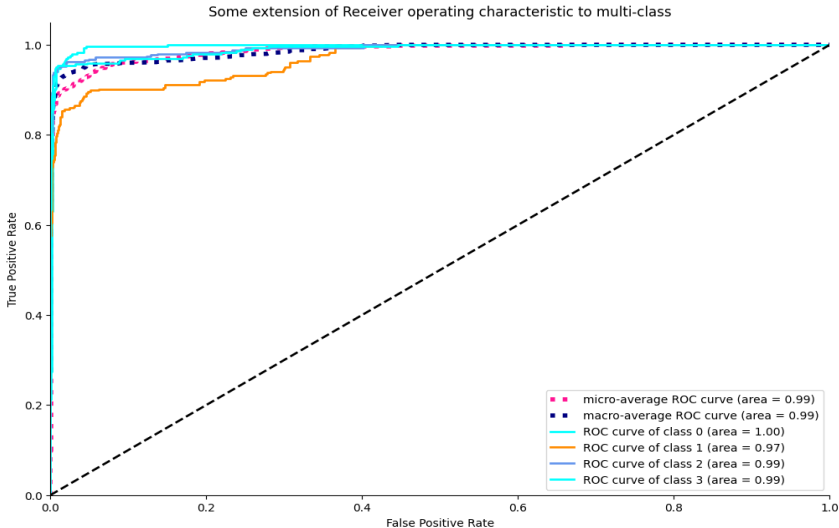


FIGURE 4.14 – le graph ROC

	precision	recall	f1-score	support
glioma	0.88	0.98	0.93	300
meningioma	0.95	0.83	0.89	306
no_tumor	0.96	0.99	0.98	405
pituitary	0.97	0.95	0.96	300
accuracy			0.94	1311
macro avg	0.94	0.94	0.94	1311
weighted avg	0.94	0.94	0.94	1311

FIGURE 4.15 – rapport de classification

Voici le tableau ci dessus qui continent tous les résultats de l'expérience : Les courbes

TABLE 4.3 – Métriques de Performance du Modèle

Métrique	Valeur
Accuracy globale sur le dataset de test	0.94
Loss globale sur le dataset de test	0.17
F1-Score global	0.94
Rappel global	0.94
Précision globale	0.94
ROC	0.99

d'entraînement et de validation montrent une performance solide du modèle sur les données de validation. Les précisions d'entraînement et de validation atteignent environ 94%, comme indiqué dans la figure 4.12, et se stabilisent après environ 18 époques. Cependant, après l'époque 18, il semble que le modèle n'améliore plus ses performances de manière significative, ce qui suggère qu'il atteint un plateau.

La courbe de perte suit une trajectoire descendante au début, avec une perte qui diminue rapidement avant de se stabiliser autour de 0,17, suggérant que le modèle minimise efficacement l'erreur pendant l'apprentissage.

Les mesures de performances globales présentées dans le tableau final soulignent une bonne généralisation du modèle. L'ROCC de 0.99 confirme la capacité de discrimination observée dans la courbe ROC. La précision, le rappel et le F1-Score globaux, tous à 0.94, indiquent que le modèle détecte les tumeurs avec une précision satisfaisante.

L'architecture ResNet50 sans CBAM montre des performances respectables, mais l'intégration de CBAM pourrait potentiellement améliorer encore ces résultats, en particulier la réduction des erreurs de classification et en passons le plateau observé après l'époque 18. ***Expérimentation 2 : évaluation des performances sur l'architecture ResNet50 intégrée au Convolutional Block Attention Module (CBAM).***

Cette expérimentation vise à évaluer les performances du modèle basé sur l'architecture ResNet50 avec le Convolutional Block Attention Module (CBAM) que nous avons intégrée. Le CBAM ajoute des mécanismes d'attention, à la fois pour les canaux et les

dimensions spatiales, afin d'améliorer la capacité du modèle à se concentrer sur les régions les plus pertinentes des images pour la classification. La Figure 4.16 montre les courbes d'apprentissage pour le modèle.

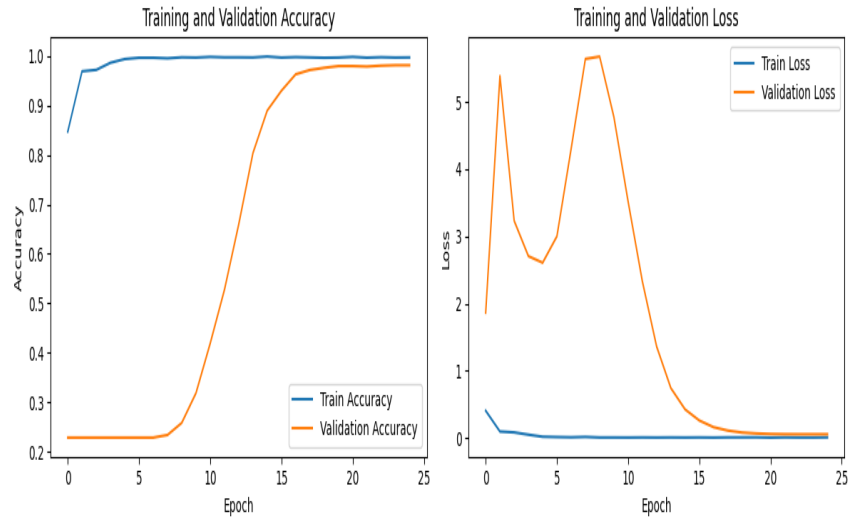


FIGURE 4.16 – graphique de accuracy et loss de l'entraînement et de la validation

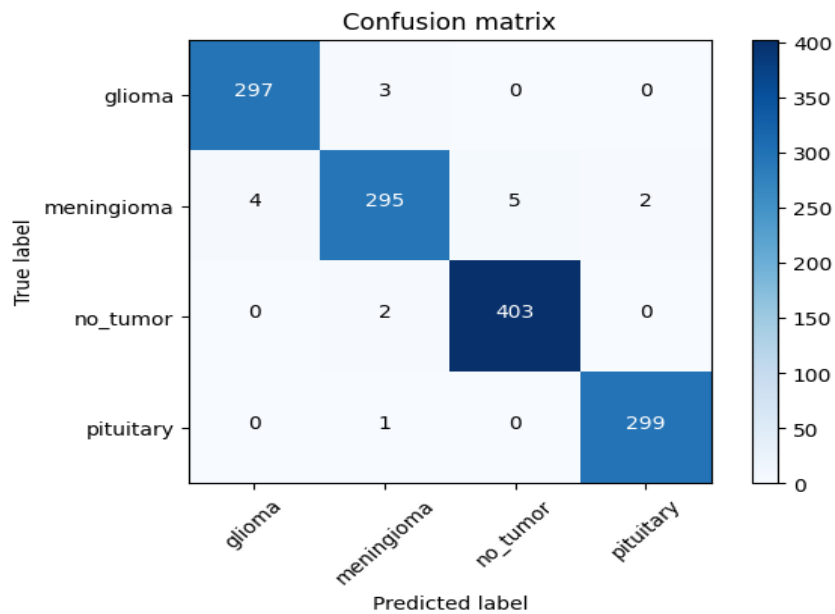


FIGURE 4.17 – la matrice de confusion

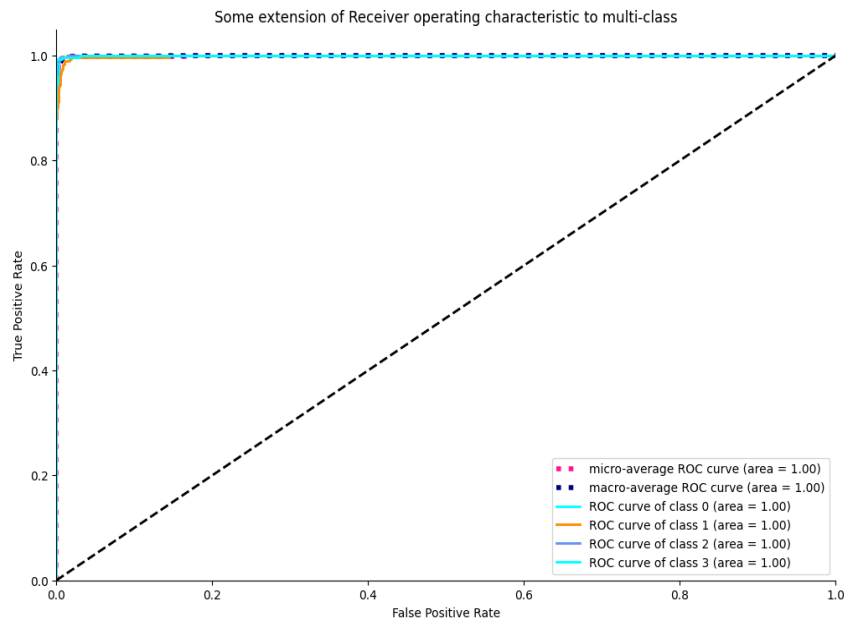


FIGURE 4.18 – le graph ROC

	precision	recall	f1-score	support
glioma	1.00	0.97	0.98	300
meningioma	0.98	0.98	0.98	306
no_tumor	1.00	1.00	1.00	405
pituitary	0.98	1.00	0.99	300
accuracy			0.99	1311
macro avg	0.99	0.99	0.99	1311
weighted avg	0.99	0.99	0.99	1311

FIGURE 4.19 – rapport de classification

Voici le tableau ci dessus qui continent tous les résultats de l'expérience :

Métrique	Valeur
Accuracy globale on test dataset	0.98
Loss globale on test dataset	0.045
F1-Score Global	0.99
Rappel Global	0.99
Précision Globale	0.99
ROC	1

TABLE 4.4 – Résumé des métriques d'évaluation du modèle

Les courbes d'entraînement et de validation montrent une performance améliorée remarquable du modèle sur les données de validation. Les précisions d'entraînement et de

validation atteignent respectivement environ 99%, comme indiqué dans la figure 1. et se stabilisent après environ 15 époques. Cela indique que le modèle converge efficacement et se généralise bien sur les données de validation. La courbe de perte Figure 4.16 suit une trajectoire descendante, avec une perte qui diminue rapidement au début, avant de se stabiliser autour de 0,045. Cela suggère que le modèle minimise effectivement l'erreur pendant l'apprentissage.

L'aire sous la courbe ROC de 1.00 représentée dans la figure 4.18 qui indique une capacité de discrimination exceptionnelle du modèle entre les différentes classes de tumeurs cérébrales et les images sans tumeur. Ce résultat démontre que le modèle est particulièrement efficace pour distinguer les différentes classes.

Cependant, la matrice de confusion figure 4.17 révèle que le modèle identifie correctement la majorité des images dans chaque classe, avec un très faible nombre de faux positifs et de faux négatifs Presque nul . Cette précision élevée montre que le modèle est bien équilibré et n'est pas biaisé en faveur d'une classe particulière.

Le F1-Score global de 0.99 est une mesure robuste qui prend en compte à la fois la précision et le rappel. Cette valeur élevée démontre que le modèle est performant non seulement en termes de précision mais aussi en termes de capacité à identifier correctement les classes positives.

Un rappel de 0.99 indique que le modèle est capable de détecter presque toutes les instances positives, ce qui est crucial pour une application médicale où la détection précise des tumeurs est essentielle.

Une précision de 0.99 signifie que presque toutes les prédictions positives du modèle sont correctes. Cela minimise le nombre de faux positifs, ce qui est important pour éviter des diagnostics erronés.

Les résultats de cette expérimentation montrent que concatination du CBAM avec l'architecture ResNet50 a considérablement amélioré la capacité du modèle à se concentrer sur les régions pertinentes des images, ce qui a conduit à des performances exceptionnelles sur toutes les métriques évaluées.

Voici la comparaison entre l'**Expérimentation 1** (ResNet50 sans CBAM) et l'**Expérimentation 2** (ResNet50 avec CBAM) présentée sous forme de tableau :

Critère	Exp 1 (Sans CBAM)	Exp 2 (Avec CBAM)	Comparaison
Précision Globale	0.94	0.99	Amélioration de la précision globale avec CBAM
Rappel Global	0.94	0.99	Meilleure identification des échantillons positifs
F1-Score Global	0.94	0.99	Performances plus équilibrées et précises avec CBAM
Classe Glioma - Précision	0.88	0.99	Précision améliorée pour la classe Glioma
Classe Meningioma - Précision	0.95	0.99	Réduction des erreurs pour la classe Meningioma
Classe No Tumor - Précision	0.96	0.99	Légère amélioration avec CBAM
Classe Pituitary - Précision	0.97	0.99	Meilleure performance dans la classification
Stabilité de l'Entraînement	Stabilisation après 18 époques	Stabilisation après 15 époques	Convergence plus rapide et efficace avec CBAM
Courbe ROC - AUC	0.99	1.00	Discrimination légèrement améliorée
Matrice de Confusion	Quelques erreurs de classification	Réduction des erreurs	Moins de faux positifs et négatifs avec CBAM
Plateau de Performances	Atteint après 18 époques	Atteint après 25 époques	Les performances continuent de s'améliorer avec CBAM

TABLE 4.5 – Comparaison des performances des expérimentations avec et sans CBAM

Ce tableau met en évidence l'impact significatif de l'intégration du module CBAM sur les performances du modèle de classification des tumeurs cérébrales, montrant des améliorations à travers presque tous les Métriques de performance.

6 L'interface du système

Pour rendre cet outil accessible et convivial, nous avons développé une interface interactive qui permet aux utilisateurs de télécharger des images médicales et de recevoir des prédictions instantanées. Notre interface est conçue pour être simple et efficace, comme indiqué dans la figure 4.20 .



FIGURE 4.20 – page index

l'utilisateur peut télécharger l'image IRM en cliquant sur le bouton « select image ».

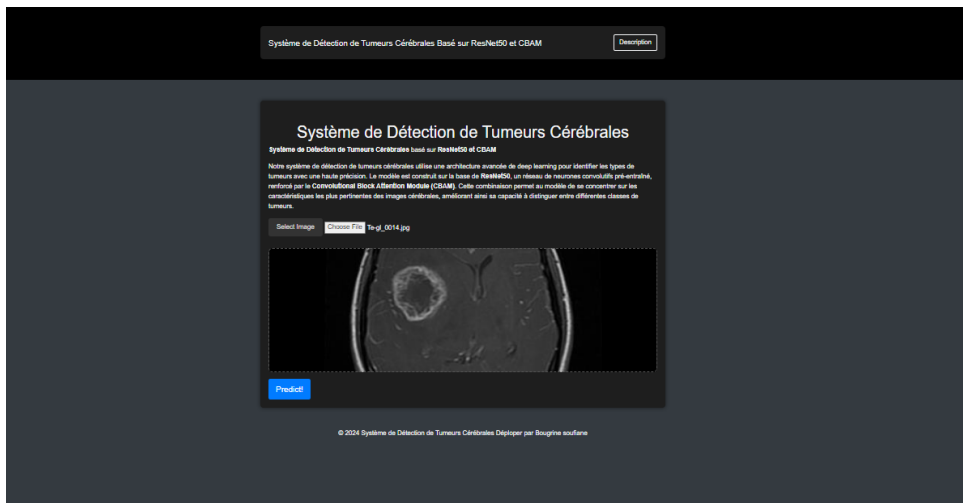


FIGURE 4.21 – select image

Après le téléchargement de l'image, le modèle effectuera le traitement et renverra le résultat ainsi que le type de tumeur cérébrale.

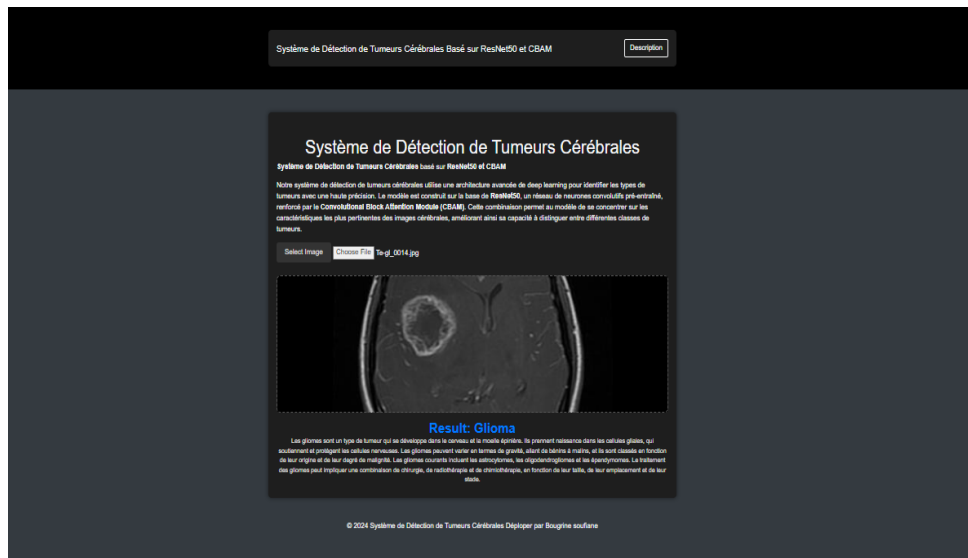


FIGURE 4.22 – Result Glioma

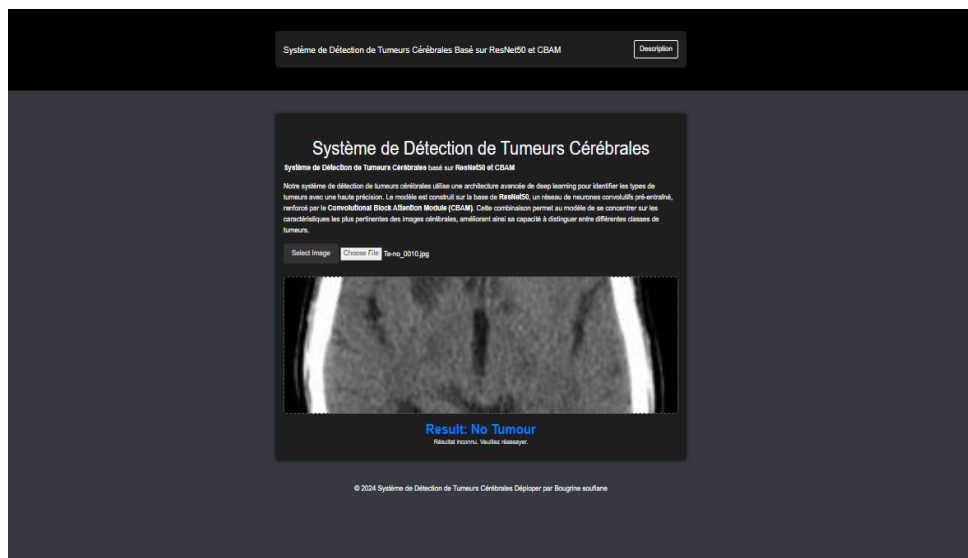


FIGURE 4.23 – result NO Tumeur

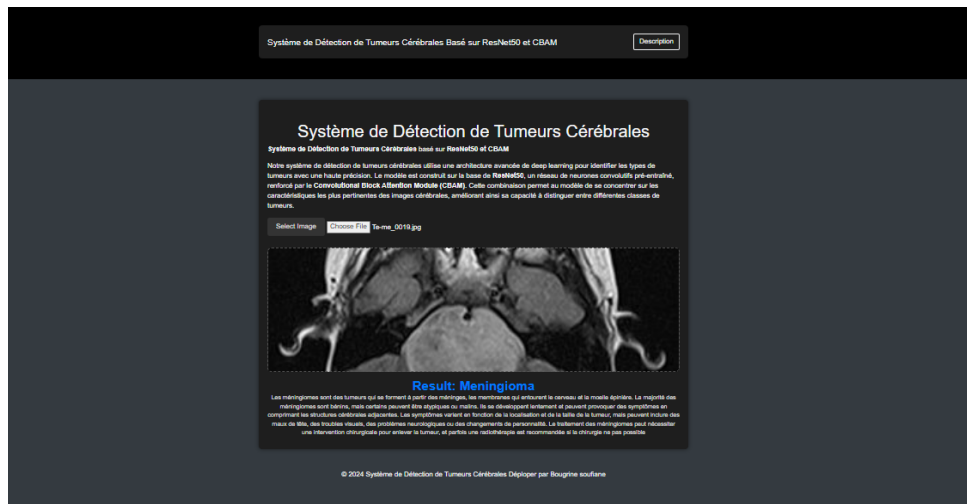


FIGURE 4.24 – result Meningioma

7 Discussion finale

L'analyse comparative des deux expériences souligne de manière significative l'importance cruciale de concatination du Convolutional Block Attention Module (CBAM) dans l'architecture ResNet50 pour la détection des tumeurs cérébrales.

Sans CBAM, le modèle basé uniquement sur ResNet50 a montré des performances respectables, mais a été limité dans sa capacité à se concentrer efficacement sur les caractéristiques les plus discriminantes des images cérébrales. Bien que le modèle ait atteint une bonne précision globale, il a été observé que certaines classes de tumeurs étaient moins bien différenciées, ce qui a entraîné des erreurs de classification et une légère diminution du rappel pour certaines classes de tumeurs.

L'intégration de CBAM a considérablement amélioré les performances du modèle. CBAM permet au modèle de mettre en avant les parties les plus pertinentes de l'image, à la fois au niveau des canaux et de l'espace, renforçant ainsi la capacité du modèle à détecter les caractéristiques importantes. Cela a conduit à une amélioration globale des mesures de performance, avec une précision, un rappel et un F1-score élevés pour toutes les classes. Le modèle avec CBAM a également montré une meilleure capacité de généralisation, minimisant les erreurs de classification, en particulier pour les classes de tumeurs les plus difficiles à distinguer.

En conclusion, concatination du CBAM avec l'architecture ResNet50 a permis de surmonter certaines des limitations rencontrées dans l'expérience initiale, où seule l'architecture ResNet50 était utilisée. Le modèle modifié non seulement montre une performance supérieure sur les métriques clés. Cette approche améliore la fiabilité du système de détection de tumeurs cérébrales et souligne l'importance d'utiliser des mécanismes d'attention pour renforcer les capacités de modèles de deep learning dans des tâches de classification médicale critiques.

Conclusion et Perspectives

En conclusion, nous avons minutieusement détaillé la mise en œuvre de notre système de détection des tumeurs cérébrales, en tirant parti de la puissance des réseaux neuronaux convolutifs (CNN) et du module d'attention par blocs convolutifs (CBAM) dans le domaine de l'apprentissage profond.

Nous avons examiné en détail chaque phase du processus de mise en œuvre, en commençant par les outils et les environnements utilisés. Les techniques de prétraitement appliquées aux données ont été développées, suivies d'une description détaillée de l'architecture du modèle et de l'intégration du module CBAM. En outre, nous avons présenté la série d'expériences menées pour évaluer les performances de notre modèle, décrit le développement de l'interface utilisateur et fourni une analyse complète des résultats.

Dans le cadre de l'amélioration continue de notre système de détection de tumeurs cérébrales, plusieurs pistes de recherche et de développement peuvent être envisagées pour les travaux futurs :

1. **Exploration d'Architectures Avancées** : Bien que ResNet50 combiné avec CBAM ait montré des résultats prometteurs, l'exploration d'autres architectures plus récentes et performantes, telles que EfficientNet ou les réseaux basés sur des mécanismes d'attention plus sophistiqués (comme les Transformers), pourrait offrir des améliorations significatives.
2. **Déploiement en Environnement Clinique** : Une étape cruciale serait de tester et de valider le système dans un environnement clinique réel. Cela inclurait des essais avec des données du monde réel pour évaluer la robustesse et l'efficacité du modèle en conditions réelles.
3. **Intégration avec d'Autres Modalités de Données** : L'intégration de notre modèle avec d'autres types de données médicales, telles que les données cliniques, génomiques ou radiologiques, pourrait permettre de créer un système plus complet et plus précis pour le diagnostic des tumeurs cérébrales.
4. **Amélioration de l'Interface Utilisateur** : Enfin, une amélioration continue de l'interface utilisateur serait bénéfique pour faciliter l'utilisation du système par les professionnels de santé. Cela pourrait inclure l'ajout de fonctionnalités interactives, une meilleure visualisation des résultats, et une adaptation aux besoins spécifiques des utilisateurs finaux.

Conclusion Générale

Les tumeurs cérébrales, bien que moins fréquentes que d'autres types de cancers, représentent un défi médical important en raison de leur complexité et de leur impact potentiellement fatal. Un diagnostic précoce est essentiel pour améliorer les chances de survie des patients, mais les méthodes traditionnelles d'imagerie médicale sont souvent insuffisantes pour identifier les tumeurs à un stade précoce.

L'automatisation du diagnostic des tumeurs cérébrales à l'aide du deep learning et de techniques d'imagerie avancées devient donc une nécessité. Les réseaux de neurones convolutifs (CNNs) se sont révélés très prometteurs pour l'analyse des images médicales grâce à leur capacité à détecter des motifs complexes.

Notre approche a cherché à améliorer les performances des modèles de classification en intégrant un module d'attention CBAM (Convolutional Block Attention Module) dans l'architecture ResNet50. Ce module permet au réseau de mieux se concentrer sur les caractéristiques les plus pertinentes des images de tumeurs.

Les résultats expérimentaux montrent que l'intégration de CBAM avec architecture ResNet50 permet d'augmenter les métriques de performance Accuracy 98% , Loss 0.045 . Ces améliorations sont significatives pour les tâches de classification des tumeurs cérébrales, et mettent en évidence la pertinence de l'ajout d'attention pour mieux guider l'apprentissage automatique dans des contextes médicaux complexes.

Les résultats encourageants obtenus ouvrent la voie à plusieurs axes d'amélioration futurs. D'une part, l'extension de cette méthode à d'autres types de tumeurs ou à d'autres types de données médicales pourrait confirmer la généralisabilité du modèle proposé. ainsi que l'utilisation de plus grandes bases de données d'images, pourrait permettre de perfectionner encore davantage les performances.

Bibliographie

- [1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, ..., and X. Zheng. Tensorflow : A system for large-scale machine learning. In *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pages 265–283. USENIX Association, 2016.
- [2] Author or Organization. Screenshot of 2020-01-29. https://user.oc-static.com/upload/2020/01/29/15803247432471_Capture%20d%E2%80%99e%CC%81cran%202020-01-29%20a%CC%80%2020.04.44.png, 2020. Accessed : 2024-09-10.
- [3] Author(s) or Organization. Title of the document or page. <https://journals.openedition.org/bmsap/4463>, 2024. Accessed : 2024-09-10.
- [4] W. Ayadi, I. Charfi, W. Elhamzi, and M. Atri. Brain tumor classification based on hybrid approach. *The Visual Computer*, 2020.
- [5] M. M. Badža and M. Č. Barjaktarović. Classification of brain tumors from mri images using a convolutional neural network. *Applied Sciences*, 10(6) :1999, 2020.
- [6] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding (CVIU)*, 110(3) :346–359, 2008.
- [7] Puja Bharati and Ankita Pramanik. Deep learning techniques—r-cnn to mask r-cnn : a survey. *Computational Intelligence in Pattern Recognition : Proceedings of CIPR 2019*, pages 657–668, 2020.
- [8] Frederik E. Bleeker, Remco J. Molenaar, and Sieger Leenstra. Recent advances in the molecular understanding of glioblastoma. *Journal of Neuro-Oncology*, 108(1) :11–27, 2012.
- [9] G. Bradski. The opencv library. *Dr. Dobb's Journal of Software Tools*, 2000. Online ; accessed August 2024.
- [10] Cameron W. Brennan, Roel G. Verhaak, Aaron McKenna, Benedikt Campos, Hugo Nounshmehr, Somasekar S. Salama, and et al. The somatic genomic landscape of glioblastoma. *Cell*, 155(2) :462–477, 2013.
- [11] Jerrold T. Bushberg, J. Anthony Seibert, Edwin M. Leidholdt Jr, and John M. Boone. *The Essential Physics of Medical Imaging*. Lippincott Williams & Wilkins, 3rd edition, 2011.
- [12] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Mathieu Douze, and the other authors. End-to-end object detection with transformers. In *European Conference on Computer Vision (ECCV)*, pages 213–229. Springer, 2020.

- [13] Michele Ceccarelli, Fredrik P. Barthel, Thiago M. Malta, Tiago S. Sabedot, Somasekar S. Salama, Benjamin A. Murray, and et al. Molecular profiling reveals biologically discrete subsets and pathways of progression in diffuse glioma. *Cell*, 164(3) :550–563, 2016.
- [14] E. Chamseddine, N. Mansouri, M. Soui, and M. Abed. Handling class imbalance in covid-19 chest x-ray images classification : Using smote and weighted loss. *Applied Soft Computing*, 129 :109588, 2022.
- [15] F. Chollet. Keras. <https://github.com/fchollet/keras>, 2015. Accessed : 2024-08-29.
- [16] Francesco Ciompi, Tarcisio de Lima de Carvalho, Bram van Ginneken, Johannes Lotz, and Jeroen AW van der Laak. The importance of standardization in digital pathology : from algorithms to diagnostic practice. *Journal of Digital Imaging*, 30(4) :493–506, 2017.
- [17] Commissariat à l'énergie atomique et aux énergies alternatives (CEA). L'essentiel sur l'imagerie médicale, 2024.
- [18] Marie Curie. *La radiologie et la guerre*. DigiCat, 2022.
- [19] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 886–893, 2005.
- [20] Data Science Today. Convolutional neural network diagram. <https://datasciencetoday.net/images/2018/08/cnn1/image019.png>, 2018. Accessed : 2024-09-10.
- [21] Richard Davis. Viral infections and their link to cancer. *Virology Journal*, 14(1) :1–10, 2015.
- [22] S. Deepak and P. M. Ameer. Brain tumor classification using deep cnn features via transfer learning. *Computers in Biology and Medicine*, 111 :103345, 2019.
- [23] V. K. Dhakshnamurthy, M. Govindan, K. Sreerangan, M. D. Nagarajan, and A. Thomas. Brain tumor detection and classification using transfer learning models. *Engineering Proceedings*, 62(1) :1, 2024.
- [24] Andre Esteva, Brett Kuprel, Roberto A. Novoa, Justin Ko, Susan M. Swetter, Helen M. Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639) :115–118, 2017.
- [25] Python Software Foundation. Python documentation, 2024.
- [26] D Galanaud, F Nicoli, D Figarella-Branger, P Roche, S Confort-Gouny, Y Le Fur, and PJ Cozzone. Spectroscopie par résonance magnétique des tumeurs cérébrales. *Journal de radiologie*, 87(6) :822–832, 2006.
- [27] Damien Galanaud and Fabrice Nicoli. Spectroscopie par résonance magnétique cérébrale. Semanticscholar, 2011.
- [28] Elena Garcia. Age as a factor in cancer development. *Journal of Aging Research*, 11(2) :115–125, 2017.
- [29] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448. IEEE, 2015.

- [30] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587. IEEE, 2014.
- [31] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.
- [32] Google User. Image title or description. <https://lh3.googleusercontent.com/Z0gcMyELwxS5X6yC0zjPd-bk8cIQmBqcUJtnShCkR1gwfXY3i80xZFxzj2MEKXvqzmz99yqYAHGgo=s600-w600>, 2024. Accessed : 2024-09-10.
- [33] Hayit Greenspan, Bram van Ginneken, and Ronald M Summers. Guest editorial deep learning in medical imaging : Overview and future promise of an exciting new technique. *IEEE Transactions on Medical Imaging*, 35(5) :1153–1159, 2016.
- [34] Ahmad Gumaei, M. M. Hassan, M. R. Hassan, A. Alelaiwi, and Giancarlo Fortino. A hybrid feature extraction method with regularized extreme learning machine for brain tumor classification. *IEEE Access*, 7 :168350–168359, 2019.
- [35] Govind Gupta and Joan Massagué. Cancer metastasis : a multistep process. *Cancer Research*, 66(10) :4796–4803, 2006.
- [36] Guru99. Convolutional neural network (convnet) diagram. https://www.guru99.com/images/tensorflow/082918_1325_ConvNetConv3.png, 2018. Accessed : 2024-09-10.
- [37] Douglas Hanahan and Robert A. Weinberg. The hallmarks of cancer. *Cell*, 100(1) :57–70, 2000.
- [38] Douglas Hanahan and Robert A. Weinberg. Hallmarks of cancer : the next generation. *Cell*, 144(5) :646–674, 2011.
- [39] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, ..., and T. E. Oliphant. Array programming with numpy. *Nature*, 585(7825) :357–362, 2020.
- [40] Simon Haykin. *Neural networks : A comprehensive foundation*. Macmillan, 1994.
- [41] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [42] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.
- [43] Sarah Ali Abdelaziz Ismael, Ammar Mohammed, and Hesham Hefny. An enhanced deep learning approach for brain cancer mri images classification using residual networks. *Artificial Intelligence in Medicine*, 2019.
- [44] Q. Ji, J. Huang, W. He, and Y. Sun. Optimized deep convolutional neural networks for identification of macular diseases from optical coherence tomography images. *Algorithms*, 12(3) :51, 2019.
- [45] Laura Johnson. Genetic risk factors for cancer. *Genetics in Medicine*, 22(5) :100–110, 2017.

- [46] Andreas Kamilaris and Francesc X. Prenafeta-Boldú. Deep learning in agriculture : A survey. *Computers and Electronics in Agriculture*, 147 :70–90, 2018.
- [47] Bruno Kastler, P Anstett, Bruno Kastler, and Daniel Vetter. *Comprendre l'IRM*. Elsevier, 2011.
- [48] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105. 2012.
- [49] Deeply Learning. Fonctionnement du neurone artificiel, sans date.
- [50] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553) :436–444, 2015.
- [51] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) :2278–2324, 1998.
- [52] G Léonetti. Difficultés et limites de l'autopsie traditionnelle. *Journal de Radiologie*, 86(10) :1452, 2005.
- [53] Daniel Lichtenstein. *L'échographie générale en réanimation*. Springer Science & Business Media, 2002.
- [54] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen AW van der Laak, Bram van Ginneken, and Clara I Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42 :60–88, 2017.
- [55] Luyang Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jian Chen, Xinwang Liu, and Matti Pietikäinen. Deep learning for generic object detection : A survey. *International Journal of Computer Vision*, 128(2) :261–318, 2020.
- [56] Xiao Liu and Wil MP van der Aalst. Retail analytics with deep learning. *International Journal of Business Analytics*, 5(4) :32–45, 2018.
- [57] David N. Louis, Hiroko Ohgaki, Otmar D. Wiestler, and Webster K. Cavenee. The 2007 who classification of tumours of the central nervous system. *Acta Neuropathologica*, 114(2) :97–109, 2007.
- [58] David G Lowe. Object recognition from local scale-invariant features. *Proceedings of the International Conference on Computer Vision*, pages 1150–1157, 1999.
- [59] Sridharan Madhavan, Jean C. Zenklusen, Yuri Kotliarov, Heather Sahni, Howard A. Fine, and Kenneth Buetow. Rembrandt : Helping personalized medicine become a reality through integrative translational research. *Molecular Cancer Research*, 7(2) :157–167, 2009.
- [60] Daniel S. Marcus, Ting-Hsin Wang, John Parker, John G. Csernansky, John C. Morris, and Randy L. Buckner. Open access series of imaging studies (oasis) : Cross-sectional MRI data in young, middle-aged, nondemented, and demented older adults. *Journal of Cognitive Neuroscience*, 19(9) :1498–1507, 2007.
- [61] Gloria Martinez. Lifestyle factors and cancer prevention. *American Journal of Lifestyle Medicine*, 12(4) :320–330, 2018.

- [62] W. McKinney. Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference*, pages 51–56. Python in Science Conference, 2010.
- [63] Medium. Image title or description. https://miro.medium.com/v2/resize:fit:768/1*nNGcN3pLeMQ-vastC1Jx7w.png, 2024. Accessed : 2024-09-10.
- [64] Bjoern H. Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, and et al. The multimodal brain tumor image segmentation benchmark (brats). *IEEE Transactions on Medical Imaging*, 34(10) :1993–2024, 2015.
- [65] K. Miller, K. Aldape, D.J. Brat, and D.N. Louis. Brain tumors. *Nature Reviews Disease Primers*, 8 :1–25, 2022.
- [66] Amaury Omuro and Lisa M. DeAngelis. Primary brain tumors in adults. *New England Journal of Medicine*, 369(5) :429–437, 2013.
- [67] Q. T. Ostrom, H. Gittleman, G. Truitt, A. Boscia, C. Kruchko, and J. S. Barnholtz-Sloan. Cbtrus statistical report : Primary brain and other central nervous system tumors diagnosed in the united states in 2011-2015. *Neuro-Oncology*, 20(suppl_4) :iv1–iv86, 2018.
- [68] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10) :1345–1359, 2010.
- [69] Dimitri Papathanassiou, Claire Bruna-Muraille, Christelle Jouannaud, Laurence Gagneux-Lemoussu, Jean-Paul Eschard, and Jean-Claude Liehn. Tomographie d’émission monophotonique couplée au scanner et pathologie osseuse. *Revue du rhumatisme*, 76(9) :843–850, 2009.
- [70] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, ..., and É. Duchesnay. Scikit-learn : Machine learning in python. *Journal of Machine Learning Research*, 12 :2825–2830, 2011.
- [71] Talliboina Ramu and A Muthukumar. A googlenet architecture based facial emotions recognition using eeg data for future applications. In *2022 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–7. IEEE, 2022.
- [72] A. Raza, H. Ayub, J. A. Khan, I. Ahmad, A. S. Salama, Y. I. Daradkeh, D. Javeed, A. Ur Rehman, and H. Hamam. A hybrid deep learning-based approach for brain tumor classification. *Electronics*, 11(7) :1146, 2022.
- [73] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once : Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016.
- [74] Joseph Redmon, Shroff Divvala, Ross Girshick, and Ali Farhadi. You only look once : Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788. IEEE, 2016.
- [75] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn : Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015.
- [76] Paul Robinson. Methods for diagnosing tumors. *Clinical Oncology*, 27(3) :305–315, 2021.

- [77] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net : Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.
- [78] Wilhelm Conrad Röntgen. *On a New Kind of Rays*. Kessinger Publishing, 1895.
- [79] Dinggang Shen, Guorong Wu, and Heung-Il Suk. Deep learning in medical image analysis. *Annual Review of Biomedical Engineering*, 19 :221–248, 2017.
- [80] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv :1409.1556*, 2014.
- [81] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.
- [82] F Varcus, JL Peix, N Berger, and D Bordos. La scintigraphie thyroïdienne : quelle place dans le bilan préopératoire des nodules thyroïdiens ? In *Annales de chirurgie*, volume 127, pages 685–689. Elsevier, 2002.
- [83] Bert Vogelstein, Nickolas Papadopoulos, Victor E Velculescu, Sidong Zhou, Luis A Diaz, and Kenneth W Kinzler. Cancer genomics : Mutational landscapes and pathways. *Cell*, 173(2) :319–331, 2013.
- [84] Xiaosong Wang, Yong Peng, Le Lu, Zhiyong Lu, Mojtaba Bagheri, and Ronald M. Summers. Chestx-ray8 : Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2097–2106, 2017.
- [85] Emily Williams. Exposure to toxins and cancer risk. *Environmental Health Perspectives*, 128(3) :213–225, 2016.
- [86] Margaret Wrensch, Yee Minn, Tai-Lan Chew, Melissa Bondy, and Mitchell S. Berger. Epidemiology of primary brain tumors : current concepts and review of the literature. *Neuro-oncology*, 4(4) :278–299, 2002.
- [87] Wannu Xu, You-Lei Fu, and Dongmei Zhu. Resnet and its application to medical image processing : Research progress and challenges. *Computer Methods and Programs in Biomedicine*, page 107660, 2023.
- [88] Zhi-Qi Zhao, Peng Zheng, Shou-Tao Xu, and Xindong Wu. Object detection with deep learning : A review. *IEEE Transactions on Neural Networks and Learning Systems*, 30(11) :3212–3232, 2019.
- [89] S. Kevin Zhou, Hayit Greenspan, and Dinggang Shen. Deep learning for medical image analysis : A survey. *Computers in Biology and Medicine*, 43(4) :197–209, 2017.
- [90] Zhengxia Zou, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years : A survey. *International Journal of Computer Vision*, 128(2) :261–318, 2019.