



HAL
open science

Segmentation automatique des lésions de sclérose en plaques en adéquation avec les contraintes cliniques

Alexandre Fenneteau

► **To cite this version:**

Alexandre Fenneteau. Segmentation automatique des lésions de sclérose en plaques en adéquation avec les contraintes cliniques. Imagerie médicale. Université de Poitiers, 2022. Français. NNT : 2022POIT2292 . tel-04023896

HAL Id: tel-04023896

<https://theses.hal.science/tel-04023896>

Submitted on 10 Mar 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Thèse de Doctorat

Université de Poitiers

ED 610 – Sciences et Ingénierie des Systèmes, Mathématiques, Informatique (SISMI)
Sciences Fondamentales Appliquées – Institut de Recherche XLIM

Thèse pour obtenir le grade de
Docteur de l'Université de Poitiers
Traitement du signal

Présentée et soutenue par

Alexandre Fenneteau

Le 24 Novembre 2022

**SEGMENTATION AUTOMATIQUE DES LÉSIONS DE SCLÉROSE EN PLAQUES EN ADÉQUATION AVEC
LES CONTRAINTES CLINIQUES.**

Thèse dirigée par David Helbert et Christophe Habas

JURY :

Rapporteurs

M. Yves Lucas, Maître de Conférences – PRISME – Université d'Orléans

M. Jean-Christophe Burie, Professeur – L3I – Université de la Rochelle

Examineurs

M. Jean-Noël Vallée, Professeur – Hôpital des Quinze-Vingts – Université Paris-Saclay

M. Pascal Bourdon, Maître de Conférences – XLIM – Université de Poitiers

M. David Helbert, Maître de Conférences – XLIM – Université de Poitiers

Mme. Christine Fernandez-Maloigne, Professeur – XLIM – Université de Poitiers

Invité

M. Matthieu Lepetit Coiffe, Collaboration Manager – Siemens Healthineers – Saint-Denis



Remerciements

Tout d'abord merci à tous les membres du jury qui ont pris le temps de lire le manuscrit et notamment aux rapporteurs Yves Lucas et Jean-Christophe Burie pour leurs retours.

Merci à Siemens Healthineers qui a soutenu ce projet et notamment Matthieu Lepetit Coiffe et Marc Lapert qui ont apporté le soutien de l'entreprise à cette thèse.

Merci à David Helbert qui a dirigé cette thèse et qui a toujours su trouver du temps, même lorsqu'il n'y en avait pas, pour m'aider, m'encadrer et m'encourager.

Merci à Pascal Bourdon qui a co-encadré cette thèse et qui a toujours su apporter sa pierre à l'édifice, qui a lancé le travail de cette thèse et qui a su m'orienter.

Je voudrais remercier Christophe Habas, qui, bien qu'il nous ait quittés, a permis à ce travail d'exister et qui a su apporter son expertise radiologique à ce travail en plus de son soutien.

Merci à Christine et Rémy qui ont lancé ce travail au sein d'I3M ainsi qu'à toute l'équipe d'I3M qui a su magnifier ce travail et l'amener toujours au plus près de la clinique grâce à Mathieu et Pierre.

Un grand merci aux collègues du laboratoire XLIM qui ont été mon quotidien pendant la majorité du temps de cette thèse. Une bonne partie d'entre eux, d'ailleurs, sont passés de collègues à amis et ont tous contribué à traverser cette aventure. Je pense notamment à Adrien, Clément, Thibault, Damien ainsi qu'aux cobureaux du 2E91: Soukayna, Charles et Guillaume qui ont dû supporter mon clavier, notamment en période de rédaction.

Je souhaite aussi remercier plus largement les amis dont certains ont pu se libérer pour ma soutenance. Ils ont été une vraie ressource.

Et puis, bien sûr, merci à ma famille, Johan et Eugénie qui ont été un soutien même s'ils n'en ont pas eu conscience. Toute ma reconnaissance s'ajoute pour mes

parents qui ont toujours été là. Vous avez contribué bien plus que ce que vous ne pensez à ce travail.

Table des Matières

1	Introduction	12
2	Contexte	16
2.1	Le système nerveux central	18
2.1.1	Cellules	19
2.1.2	Tissus	21
2.1.3	La moelle spinale	23
2.1.4	L'encéphale	23
2.2	Principes de l'IRM	27
2.2.1	Les principes de la résonance magnétique nucléaire	28
2.2.2	Acquisition 3D	31
2.2.3	Pondération RMN	32
2.3	La sclérose en plaques	33
2.3.1	Les origines	34
2.3.2	SEP et IRM	34
2.3.3	Les formes	37
2.3.4	Le diagnostic et le suivi	39
2.4	Apprentissage profond et réseau de neurones convolutifs	41
2.4.1	Bref historique	41
2.4.2	Le modèle du réseau de neurones	42
2.4.3	Les réseaux de neurones convolutifs	43
2.5	La segmentation automatique des lésions de sclérose en plaques	52
3	Rafinement de l'architecture d'un réseau de neurones	53
3.1	État de l'art	55
3.2	Adaptation légère du U-net, le MPU-net	57
3.2.1	Données	58
3.2.2	Matériels et méthodes	60
3.2.3	Expérience	64

3.2.4	Réflexion sur les choix d'architecture	76
3.2.5	Conception du MPU-net	78
3.2.6	Discussion	80
3.3	Du MPU-net au MPU-next	81
3.3.1	Raffinement de l'architecture	82
3.3.2	Résultats et discussion	84
3.4	Adaptation du MPU-next pour la segmentation des nouvelles lésions de SEP	88
3.4.1	Contexte	88
3.4.2	Méthode proposée	90
3.4.3	Résultats du challenge	93
3.5	Conclusion	95
4	La réduction des données d'apprentissage	96
4.1	Apprentissage autosupervisé pour la segmentation des lésions de SEP	97
4.1.1	Contexte	98
4.1.2	Matériels et méthodes	99
4.1.3	Résultat et analyse	103
4.2	Nombre de patients nécessaire	106
4.2.1	Matériels et méthodes	107
4.2.2	Résultat et analyse	108
4.3	Apprentissage semi-supervisé de la segmentation des lésions	109
4.3.1	Introduction	109
4.3.2	Évaluation de l'ensemble de données	111
4.3.3	Conception d'une méthode semi-supervisée pour la segmentation des lésions de sclérose en plaques (SEP)	114
4.4	Conclusion	120
5	Vers un meilleur apprentissage pour la segmentation des lésions de SEP	121
5.1	La taille des données en entrées pour la segmentation des lésions de SEP	122
5.1.1	Contexte	123
5.1.2	Matériels et méthodes	124
5.1.3	Résultats et analyse	125
5.2	Adaptation de la fonction de coût vers une meilleure détection des lésions	129
5.2.1	Analyse du problème	129
5.2.2	Matériels et méthodes	130
5.2.3	Résultats et analyse	134

5.3 Conclusion	141
6 Conclusion	143
7 Bibliographie	148
Liste des travaux	149
Références	150

Table des Figures

2.1	Les différentes cellules du système nerveux central (modifié par Alexandre Fenneteau, Servier Medical Art, CC BY 3.0).	19
2.2	Schéma légendé d'un neurone (modifié par Alexandre Fenneteau, Servier Medical Art, CC BY 3.0).	21
2.3	Schéma de la circulation du liquide cérébro-spinal (LCS) (modifié par Alexandre Fenneteau, Servier Medical Art, CC BY 3.0).	22
2.4	Schémas de la moelle spinale.	23
2.5	Schéma des méninges entourant l'encéphale (modifié par Alexandre Fenneteau, Servier Medical Art, CC BY 3.0).	24
2.6	Le télencéphale (Servier Medical Art, CC BY 3.0).	25
2.7	Les différentes structures de l'encéphale (modifié par Alexandre Fenneteau, Servier Medical Art, CC BY 3.0).	26
2.8	Le système ventriculaire (modifié par Alexandre Fenneteau, Blausen Medical, CC BY 3.0).	27
2.9	Les spins des noyaux d'hydrogène (à gauche) responsables de l'aimantation macroscopique M_z (à droite) dans le champ magnétique B_0	28
2.10	Le phénomène RMN. Avant excitation, l'aimantation M des atomes d'hydrogène est dans l'axe z (1), après excitation B_1 , les spins sont basculés progressivement dans l'axe xy (2) jusqu'à précéder perpendiculairement à l'axe z (3).	29
2.11	La récupération du signal RMN en fonction des TR et TE.	30
2.12	Le contraste de différents tissus en fonction des TR et TE.	31
2.13	Exemple de lésions visibles avec les pondérations imagerie par résonance magnétique nucléaire (IRM) usuelles pour examens d'une SEP.	35
2.14	Localisations de lésions de SEP dans l'encéphale : coupes IRM fluide atténuation par inversion-récupération (FLAIR) d'un patient atteint de SEP en haut avec segmentation des lésions en rouge en bas.	36
2.15	Exemples d'évolution de forme récurrente rémittente.	37
2.16	Exemples d'évolution de forme progressive primaire.	38

2.17 Exemples d'évolution de forme progressive secondaire.	39
2.18 Réseau de neurones dense ou perceptron multicouche simple. Les trois couches de neurones (d'entrée, cachée et de sortie) sont liées entre elles par des connexions denses.	43
2.19 Architecture classique simplifiée de RNC pour une tâche de classification. Cette architecture contient des couches convolutives, des couches de sous-échantillonnage et des couches denses.	48
3.1 Le U-net de Ronneberger issu de l'article original [37].	58
3.2 La taille des patches en rouge et la taille des lésions en vert en vue axiale à gauche et sagittale à droite.	60
3.3 Le modèle minimal d'architecture employé pour l'adaptation simple du U-net.	61
3.4 Les différents changements appliqués à l'architecture simplifiée du U-net utilisée.	62
3.5 Variation avec deux (a) et trois (b) niveaux de l'architecture de référence en Figure 3.3 . Le nombre de cartes de caractéristiques de sortie est noté en rouge en dessous des couches convolutives.	63
3.6 La version avec deux niveaux de l'architecture avec des convolutions avec des pas de 2.	65
3.7 La version avec deux niveaux de l'architecture sans échantillonnage.	65
3.8 Évaluation des différents échantillonnages.	66
3.9 Les versions à deux niveaux de l'architecture sans connexions à gauche et sans connexions ni échantillonnage à droite.	67
3.10 Évaluation de l'apport des connexions par concaténation.	68
3.11 La version à deux niveaux de l'architecture avec deux couches convolutives par bloc.	69
3.13 Évaluation de l'impact de l'augmentation du nombre de cartes de caractéristiques.	72
3.14 Le score de Dice moyen de toutes les expériences.	73
3.15 Sensibilité (a) et précision (b) en fonction du nombre de couches convolutives.	73
3.16 Segmentation moyenne des versions de références.	74
3.17 Segmentation moyenne des meilleures versions de chaque expérience.	75
3.18 L'architecture proposée le MPU-net avec 33 332 paramètres d'apprentissages et 21 couches convolutives.	79

3.19	Les modèles d'architectures du minimally parameterized U-net (MPU-net) et du MPU-net++. Le contenu des blocs est illustré dans la Figure 3.20 .	83
3.20	Les différentes variations des blocs testés avec les modèles d'architectures présentés dans la Figure 3.19 .	84
3.21	Graphique en boîte à moustaches du score de Dice sur l'ensemble de test pour chaque architecture testée. La boîte représente le deuxième et troisième quartile et les moustaches contiennent le reste de la distribution des données exceptées les valeurs aberrantes représentées par les diamants.	86
3.22	Architecture proposée pour la segmentation des nouvelles lésions au challenge MSSEG2 : l'architecture contient un MPU-net amélioré (MPU-net) siamois comme bloc de segmentation et bloc de raffinement convolutif.	91
4.1	Le prétraitement réalisé avec l'exemple d'une image T2.	100
4.2	Étapes d'apprentissage : 1) Les patchs 3D de plusieurs séquences sont extraits à partir des volumes entiers. 2) Les patchs sont envoyés au premier réseau neuronal pour l'apprentissage de la tâche de localisation. 3) Les poids des couches sont transférés au CNN de segmentation. 4) Apprentissage de la tâche de segmentation.	102
4.3	Diagramme en boîte à moustache des scores obtenus sur l'ensemble de test. La médiane est représentée par les lignes orange, l'intervalle interquartile par la boîte, les moustaches symbolisent l'intervalle de répartition des données et les points sont les valeurs aberrantes.	104
4.4	Visualisation des différentes segmentations sur une coupe. Les principales zones de différence sont entourées par des ellipses orange.	106
4.5	Le score de Dice en fonction du nombre de patients d'apprentissage. Chaque point représente un patient testé pour chacune des 5 répétitions de l'expérience. Les moustaches représentent l'écart type centré autour de la moyenne.	108
4.6	Les scores de Dice et F1 moyens avec le même modèle appris séparément sur chaque examen et testé sur les examens restants. Les nombres sont les identifiants des patients. Nous avons découpé le nuage de points en trois classes pour classifier les exemples.	112

5.1	Performances du modèle en termes de score de Dice (à gauche) et de score F1 (à droite) en fonction de la taille du patch avec différentes résolutions : <i>Natif</i> ($0,8 \times 0,47 \times 0,47 \text{ mm}^3$) , <i>Iso min</i> ($0,47 \times 0,47 \times 0,47 \text{ mm}^3$) et <i>Iso max</i> ($0,8 \times 0,8 \times 0,8 \text{ mm}^3$)	126
5.2	La fonction de pondération utilisée avec sa représentation et son paramétrage k , $vmax$ et $inter$ représenté.	132
5.3	Les différentes valeurs d' <i>inter</i> et de k testées avec les profils de pondérations.	135
5.4	Les scores moyens de score de Dice, F1, sensibilité et précision de détection obtenues en variant les valeurs de k et d' <i>inter</i> . Pour chaque métrique, si la couleur est bleue cela signifie que la valeur est inférieure à la performance sans pondération (dans le Tableau 5.4) et si la couleur est rouge cela signifie qu'elle est supérieure.	137
5.5	Visualisation des segmentations avec apprentissage du modèle avec différentes pondérations.	139

1

Introduction

LA sclérose en plaques est une maladie dégénérative handicapante qui affecte des personnes jeunes. Le diagnostic et la caractérisation de la maladie sont déterminants pour fixer le traitement du patient et ainsi limiter au mieux l'avancée de la maladie et du handicap associé. Le neurologue, pour sa prise de décision thérapeutique, s'appuie sur le rapport du radiologue réalisé sur les images par résonance magnétique du patient qui quantifie et caractérise les lésions causées par la maladie dans le système nerveux central.

Dans son rapport, le radiologue détermine la localisation des lésions, leurs caractéristiques et s'intéresse à l'apparition d'éventuelles nouvelles lésions en cas de suivi du patient. La recherche de lésions de sclérose en plaques est une tâche longue, répétitive, fastidieuse qui est soumise à des variations selon les radiologues et même pour un même radiologue à un temps différent. Il y a donc un intérêt à automatiser cette tâche pour aider le radiologue en diminuant le temps d'analyse et en amenant vers une mesure reproductible d'observations visuelles.

La segmentation automatique des lésions permet au radiologue de repérer et de quantifier les lésions rapidement. Cette opération consiste en la classification binaire de chaque voxel de l'image comme un voxel appartenant à une lésion ou non. La segmentation automatique des lésions de sclérose en plaques reste un champ de recherche toujours actif malgré les nombreuses contributions sur le sujet.

Les dernières méthodes de segmentation automatique des lésions de sclérose en plaques, les plus performantes, font usage de l'apprentissage automatique et plus particulièrement de l'apprentissage profond avec des réseaux de neurones convolutifs. Cependant, le prix du gain de performance apporté par les réseaux de neurones artificiels est une explosion de la complexité des modèles nécessitant une grande puissance de calcul et un grand nombre de données annotées. Le travail présenté se focalise sur la réduction de la puissance de calcul et du nombre de données d'apprentissage annotées nécessaires qui sont des freins pour le développement actif de la recherche sur le sujet dans l'environnement clinique qui constitue pourtant la source du besoin

avec les connaissances médicales.

Le travail réalisé durant cette thèse CIFRE Siemens Healthineers résulte de la collaboration entre Siemens Healthineers, l'hôpital des Quinze-Vingts et le laboratoire XLIM avec la volonté de développer une imagerie toujours plus fine et informative vers la biopsie virtuelle. L'entreprise Siemens Healthineers est spécialisée dans l'ingénierie biomédicale et en particulier sur l'imagerie médicale et l'imagerie par résonance magnétique nucléaire. L'entreprise développe et produit des systèmes d'imagerie par résonance magnétique de pointe pour une utilisation clinique en radiologie. Siemens Healthineers, les laboratoires CNRS XLIM UMR 7252 et LMA UMR 7348 de l'Université de Poitiers et le CHU de Poitiers se sont rassemblés au sein du laboratoire commun CNRS entreprise I3M (Imagerie Métabolique Multi-noyaux Multi-organes). Le laboratoire I3M exploite l'IRM MAGNETOM Terra 7 Tesla à usages cliniques et de recherches acquise par le CHU de Poitiers en 2019 et travaille sur la mise en œuvre technique et innovante de méthodes pour le traitement et l'analyse des images IRM pour l'aide aux diagnostics et le suivi des pathologies. Le service de neuro-imagerie de l'hôpital des Quinze-Vingts s'est ajouté, par la suite, au laboratoire I3M avec un projet sur la segmentation des lésions de scléroses en plaques qui a débouché sur la présente thèse.

Dans une première partie, le contexte du travail multidisciplinaire de la thèse avec ses spécificités est abordé. Les dimensions biologiques et physiologiques des systèmes affectés par la maladie aux niveaux microscopiques à macroscopiques et les principes d'acquisition des images par résonance magnétique nucléaire sont d'abord présentés pour servir de base à la description de la sclérose en plaques, de son diagnostic et de son suivi. Les approches de segmentation abordées dans ce manuscrit font appel aux méthodes d'apprentissage profond et aux réseaux de neurones convolutifs qui sont ensuite introduits avant de donner un aperçu de la tâche de segmentation automatique des lésions de sclérose en plaques qui est le propos principal de la thèse.

Le second chapitre s'articule autour des contributions apportées pour créer un modèle de segmentation des lésions léger, adapté à la tâche. La conception de l'architecture légère a pour but de réduire le coût en calcul de l'apprentissage d'un modèle performant et ainsi baisser les besoins matériels ainsi que le besoin en données d'apprentissage. Ce chapitre contient un état de l'art sur les méthodes de segmentation des lésions de sclérose en plaques focalisé sur les dernières méthodes utilisant des réseaux de neurones convolutifs et leurs architectures, puis, la conception de notre premier modèle léger, le MPU-net, est détaillée pour mener à son amélioration : le MPU-next. Enfin, l'architecture d'un modèle spécialement conçue pour la détection des nouvelles lésions, basée sur le MPU-next, est détaillée et comparée à d'autres

méthodes.

Le besoin et la réduction du nombre d'exemples d'apprentissage pour nos méthodes sont investigués dans le troisième chapitre de ce manuscrit. Dans la cette partie, l'utilisation de l'autosupervision, visant à réduire le besoin en exemple d'apprentissage, ne permet pas un apport significatif en termes de qualité de segmentation. Guidées par l'hypothèse que le nombre d'exemples d'apprentissage est suffisant, les recherches sur le nombre d'exemples nécessaires sont présentés avant de passer à l'élaboration d'une technique semi-supervisée visant à réduire l'apport d'annotation pour l'apprentissage de la segmentation des lésions.

Enfin, dans le dernier chapitre, des études sont menées pour améliorer encore les performances de segmentation et de détections des lésions de sclérose en plaques. La forme des images d'entrées avec leurs diversités, leurs tailles physiques et voxeliques sont étudiées pour fournir les meilleurs résultats de segmentation possible. Puis, le problème de la détection des petites lésions est abordé. En effet, les petites lésions ont tendance à ne pas être segmentées avec les algorithmes d'apprentissage profond bien que leur détection soit tout autant sinon plus importante que les grandes lésions. Une méthode pour détecter de façon plus sélective les lésions en fonction de leur taille est élaborée afin de permettre une meilleure détection générale des lésions de sclérose en plaques vers un meilleur outil pour le radiologue avant de conclure sur les contributions de la thèse.

2

Contexte

Sommaire

2.1	Le système nerveux central	18
2.1.1	Cellules	19
2.1.2	Tissus	21
2.1.3	La moelle spinale	23
2.1.4	L'encéphale	23
2.2	Principes de l'IRM	27
2.2.1	Les principes de la résonance magnétique nucléaire	28
2.2.2	Acquisition 3D	31
2.2.3	Pondération RMN	32
2.3	La sclérose en plaques	33
2.3.1	Les origines	34
2.3.2	SEP et IRM	34
2.3.3	Les formes	37
2.3.4	Le diagnostic et le suivi	39
2.4	Apprentissage profond et réseau de neurones convolutifs	41
2.4.1	Bref historique	41
2.4.2	Le modèle du réseau de neurones	42
2.4.3	Les réseaux de neurones convolutifs	43

2.5 La segmentation automatique des lésions de sclérose en plaques . . . 52

LE travail présenté dans ce manuscrit est focalisé sur la segmentation automatique des lésions de sclérose en plaques. La recherche effectuée repose sur le développement de méthodes de vision par ordinateur et d'apprentissage profond spécifiques et adaptées pour l'imagerie médicale par résonance magnétique nucléaire sur une problématique d'aide aux diagnostics.

Pour comprendre les enjeux médicaux et mécanismes biologiques sous-jacents, nous décrivons ici, dans un premier temps, l'anatomie et les fonctions cellulaires du système nerveux central qui est altéré par la sclérose en plaques. Puis, pour appréhender l'objet analysé tout au long du manuscrit, le principe physique de l'imagerie par résonance magnétique nucléaire et l'information contenue par les images produites sont présentés. Ensuite, la sclérose en plaques est présentée et notamment l'apport de l'IRM et du radiologue dans le diagnostic et le suivi de la maladie. Après quoi, l'apprentissage profond qui est l'ensemble de méthodes retenues pour répondre aux problématiques de la thèse est présenté avant d'introduire la tâche de segmentation automatique des lésions de sclérose en plaques.

2.1 Le système nerveux central

La sclérose en plaques affecte le système nerveux. Ce dernier est composé du système nerveux périphérique et du système nerveux central (SNC). Le SNC est composé de l'encéphale (ou cerveau) et de la moelle spinale et le système nerveux périphérique est constitué des nerfs et ganglions dans le reste du corps. Le SNC est responsable du traitement des informations reçues et pilote la majorité des réponses nerveuses, il est constitué de différents tissus eux-mêmes constitués de plusieurs types cellulaires ayant

des rôles différents.

2.1.1 Cellules

Le système nerveux contient un certain nombre de cellules spécialisées qui sont chacune spécialisées pour une ou plusieurs fonctions.

Les cellules du SNC vivent dans un écosystème complexe et régulé par différentes cellules gliales chacune spécialisée pour un ensemble de tâches afin de permettre aux cellules neuronales de propager et traiter l'information (voir **Figure 2.1**).

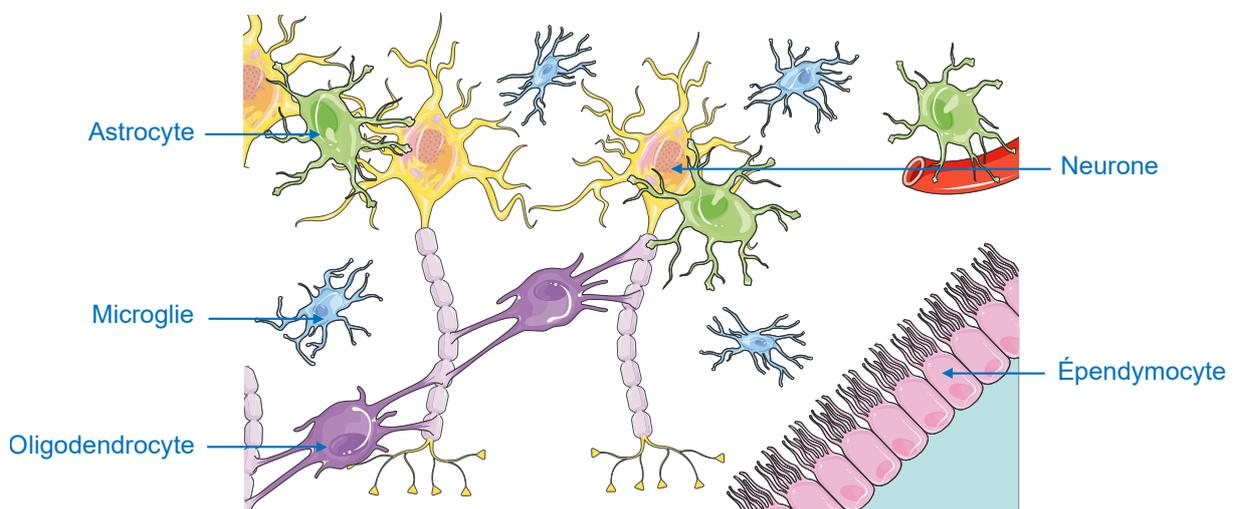


Figure 2.1 – Les différentes cellules du système nerveux central (modifié par Alexandre Fenneteau, Servier Medical Art, CC BY 3.0).

Cellules gliales

Les cellules gliales sont des cellules du système nerveux qui assurent la survie, le développement, le maintien et la défense des neurones. Il existe plusieurs types de cellules gliales qui ont chacun des rôles différents.

Les épendymocytes sont des cellules épithéliales (cellules de revêtement) qui produisent le liquide cérébro-spinal (voir **Section 2.1.2**) et assurent sa circulation. Ces cellules sont au cœur des différents échanges avec les autres cellules nerveuses et le liquide cérébro-spinal.

La Microglie assure les fonctions immunitaires dans le SNC.

Les astrocytes supportent les neurones structurellement et leur apportent les nutriments nécessaires à leur fonctionnement. Ils sont aussi impliqués dans la tenue de la barrière hématoencéphalique qui assure la séparation entre le sang et le SNC pour le protéger. Les astrocytes interviennent également dans la communication entre les neurones en maintenant un espace intercellulaire propice à l'échange rapide et efficace de neurotransmetteur.

Les oligodendrocytes sont des cellules qui enroulent leur membrane riche en lipides et lipoprotéines autour des axones des neurones pour former une gaine protectrice et isolante: la myéline (voir **Section 2.1.1**) dans le SNC. Un oligodendrocyte peut myéliniser plusieurs neurones.

Les cellules de Schwann ont le même rôle que les oligodendrocytes, mais dans le système nerveux périphérique. Elles forment aussi une gaine de myéline autour des neurones, mais à la différence des oligodendrocytes, une cellule de Schwann ne recouvre qu'un axone de neurone périphérique et un grand nombre de cellules de Schwann peuvent être nécessaires pour recouvrir l'axone entier qui peut atteindre plus d'un mètre de longueur.

Neurones

Le neurone est la cellule du système nerveux central capable de capter, transmettre et traiter l'information dans le but de générer une réponse adéquate. Il existe plusieurs types de neurones avec des fonctions et structures qui varient. Chaque neurone est constitué d'un corps cellulaire contenant les structures suivantes illustrées dans la **Figure 2.2** :

- le noyau et les organites nécessaires à la vie de la cellule,
- de dendrites qui reçoivent l'information sous forme chimique (neurotransmetteur) ou électrique (jonction communicante),
- l'axone qui achemine l'information sous forme d'un potentiel d'action électrique,
- les synapses qui transmettent l'information à une cellule réceptrice (neurone ou autre).

Certains neurones ont les axones myélinisés, c'est-à-dire couverts d'une gaine de myéline. La gaine de myéline est riche en lipides et agit comme une gaine électriquement isolante autour de l'axone. Lorsque l'axone est myélinisé, ce sont en fait de multiples cellules de Schwann ou oligodendrocytes (voir **Section 2.1.1**) qui entourent

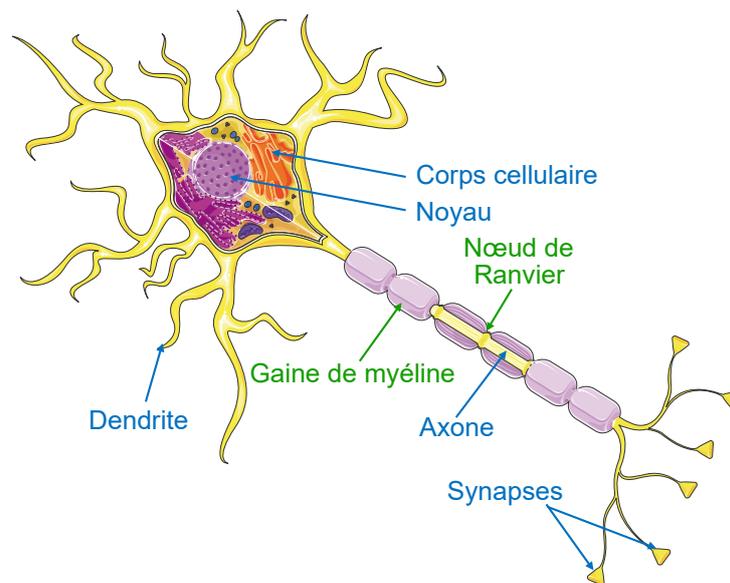


Figure 2.2 – Schéma légendé d'un neurone (modifié par Alexandre Fenneteau, Servier Medical Art, CC BY 3.0).

une partie de leur membrane autour de l'axone. Il se forme entre chaque "rouleau" de myéline consécutif un petit espace dépourvu de myéline appelé nœud de Ranvier (voir **Figure 2.2**). Dans un axone myélinisé, le potentiel d'action électrique se propage de nœud de Ranvier en nœud de Ranvier de façon saltatoire alors que le potentiel se propage de façon continue le long de l'axone non myélinisé. La conduction saltatoire permet d'augmenter fortement la vitesse de l'influx le long de l'axone. La myéline est donc vitale pour la propagation rapide de l'information au sein du système nerveux.

L'ensemble de cellules détaillées est organisé en tissus qui interagissent eux-mêmes avec d'autres tissus. Dans la partie suivante, nous allons détailler les principaux tissus du SNC.

2.1.2 Tissus

Le système nerveux est constitué principalement par trois tissus biologiques : le liquide cérébro-spinal, la substance blanche et la substance grise.

Le liquide cérébro-spinal

Le liquide cérébro-spinal (LCS) est un liquide qui entoure le système nerveux qui a une composition proche de celle du plasma et qui est produit en majorité par les

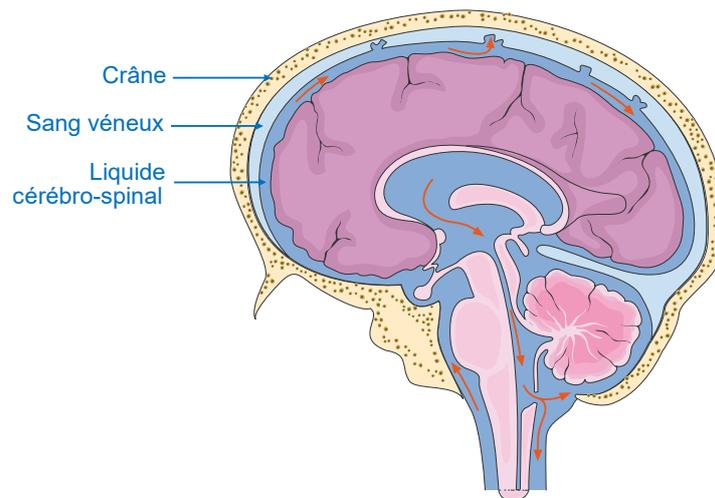


Figure 2.3 – Schéma de la circulation du LCS (modifié par Alexandre Fenneteau, Servier Medical Art, CC BY 3.0).

plexus choroïdes. Les échanges entre le sang et le LCS sont limités par la barrière hémato-méningée pour protéger le système nerveux. Le LCS ne contient que certaines cellules immunitaires, nutriments, protéines, hormones et minéraux. Il permet notamment l'évacuation des déchets toxiques en trop grande concentration. Le LCS protège mécaniquement le cerveau, mais l'isole aussi du reste de l'organisme pour éviter d'éventuelles infections (voir **Figure 2.3**). Il a un rôle de transport notamment pour les hormones et les neurotransmetteurs.

La substance blanche

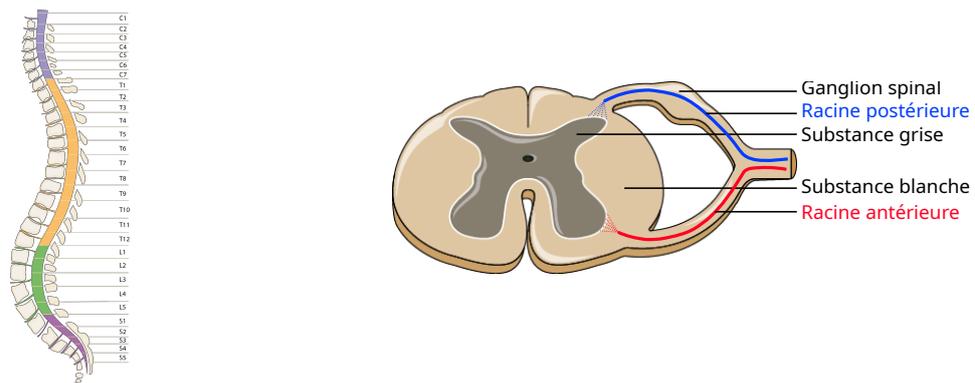
La substance blanche contient majoritairement des axones myélinisés. Le nom de la substance blanche provient du fait que la couleur du tissu après traitement au formaldéhyde est blanche à cause de la myéline riche en lipides. La substance blanche forme un réseau d'axones qui relie plusieurs aires de substance grise entre elles.

La substance grise

La substance grise contient majoritairement des dendrites, des corps cellulaires et des axones non myélinisés (voir **Section 2.1.1**), quelques cellules gliales (voir **Section 2.1.1**) et des capillaires sanguins. Elle contient les régions du cerveau impliquées notamment dans le contrôle moteur et sensitif. La substance grise est par opposition à la substance blanche plus sombre, car contient moins de fibres myélinisées.

L'ensemble de ces tissus constituent le SNC, composé de la moelle épinière et de l'encéphale eux-mêmes subdivisés en plusieurs parties.

2.1.3 La moelle spinale



(a) Schéma de la moelle spinale avec les prolongements des nerfs rachidiens (Servier Medical Art, CC BY 3.0). (b) Schéma légendé d'une coupe de moelle spinale (modifié par Alexandre Fenneteau, Servier Medical Art, CC BY 3.0).

Figure 2.4 – Schémas de la moelle spinale.

La moelle spinale, illustrée dans la **Figure 2.4a**, a pour rôle de transmettre les informations entre l'encéphale et le reste du corps. Elle contrôle aussi un certain nombre de réflexes dont le réflexe rotulien. La moelle spinale est reliée avec les autres nerfs du corps par ses paires de nerfs spinaux ou rachidiens qui sont réparties tout le long de la colonne vertébrale du sacrum et du coccyx.

La moelle spinale est composée de substance grise au centre et de la substance blanche en périphérie (voir **Figure 2.4b**). La substance grise est répartie de façon symétrique en "papillon". Des paires de nerfs spinaux (gauches et droites) partent régulièrement de la moelle épinière.

2.1.4 L'encéphale

L'encéphale constitue la partie supérieure du SNC, c'est un organe complexe agissant comme le chef d'orchestre de l'organisme. Il intègre, traite et envoie les réponses nerveuses volontaires et involontaires et régule les différents systèmes endocriniens. Il est constitué de plusieurs parties anatomiques associées à différents rôles, chaque partie est composée de tissus eux-mêmes constitués de différents types cellulaires.

Anatomie

L'encéphale est constitué de quatre grandes structures illustrées dans la **Figure 2.7**:

- le télencéphale,
- le diencéphale,
- le tronc cérébral,
- le cervelet.

L'encéphale est un organe vital complexe contenu par la boîte crânienne, les méninges et le LCS qui assurent son maintien et sa protection et sont illustrés dans la **Figure 2.5**.

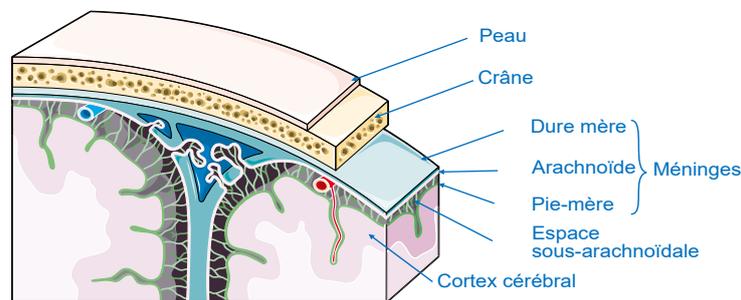


Figure 2.5 – Schéma des méninges entourant l'encéphale (modifié par Alexandre Fenneteau, Servier Medical Art, CC BY 3.0).

La boîte crânienne est une structure osseuse rigide qui contient l'encéphale enveloppé par les méninges qui sont elles-mêmes un ensemble de membranes (dure-mère, arachnoïde et pie-mère) qui maintiennent le SNC dans le LCS. Le LCS est contenu entre l'arachnoïde et la pie-mère dans l'espace sous-arachnoïdale (voir **Figure 2.5**).

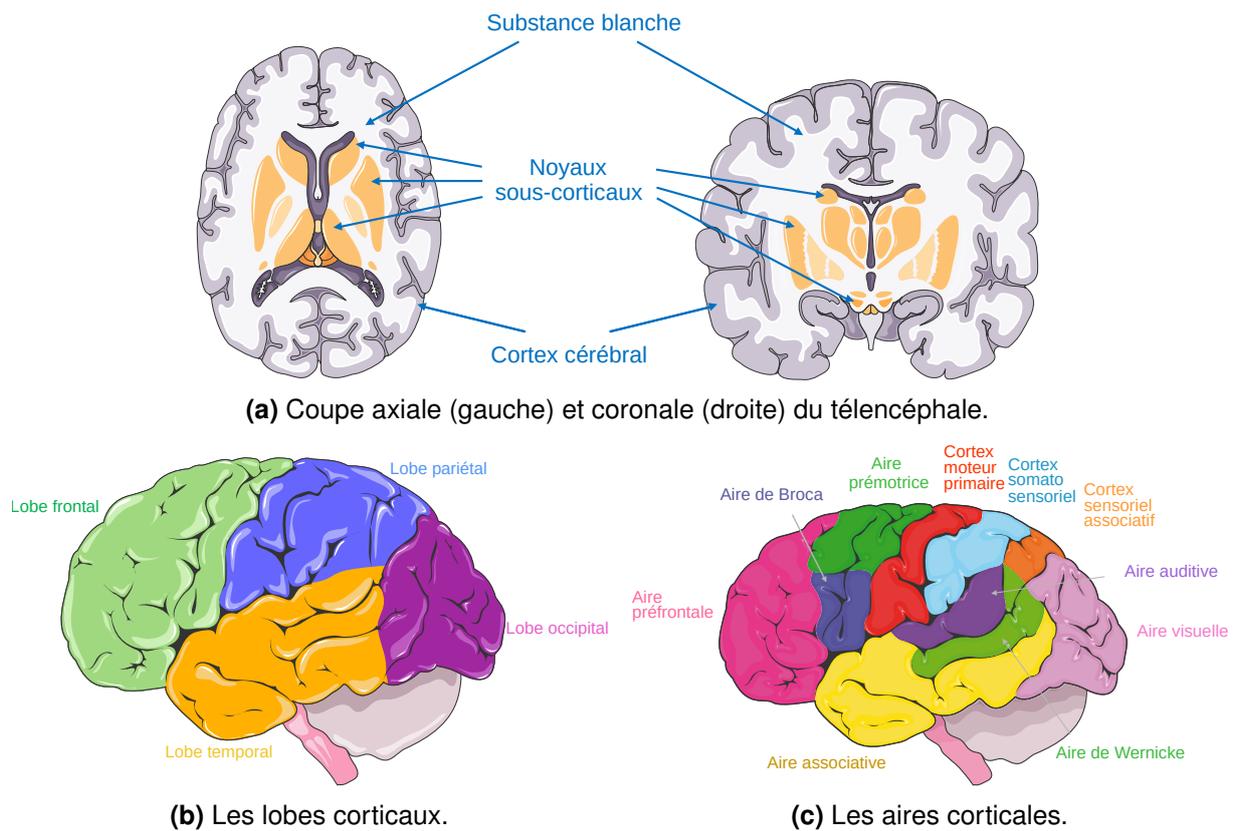


Figure 2.6 – Le télencéphale (Servier Medical Art, CC BY 3.0).

Le télencéphale

Le télencéphale est la partie de l'encéphale la plus volumineuse qui contient la substance blanche, le cortex cérébral et les structures sous-corticales des hémisphères cérébraux (voir **Figure 2.6**).

Le cortex cérébral est constitué de la substance grise à l'extérieur des hémisphères. C'est la région où la majorité des informations du SNC est traitée et intégrée. Le cortex cérébral est découpé en quatre grands lobes visibles dans la **Figure 2.6b** et en plusieurs aires cérébrales délimitées par leurs rôles fonctionnels illustrés dans la **Figure 2.6c**.

Les structures sous-corticales sont constituées de plusieurs noyaux de substance grise. Ils contiennent les amygdales, les ganglions de la base (ensemble de noyaux) et l'hippocampe. Ces noyaux sont tous constitués de substance grise.

Le diencéphale

Le diencéphale est une partie au centre de l'encéphale dans la **Figure 2.7**. Le

diencéphale regroupe plusieurs structures telles que le thalamus, l'hypothalamus, l'épithalamus et le sous-thalamus. Le diencéphale est impliqué dans les fonctions du système nerveux autonome, dans le sommeil et la régulation hormonale.

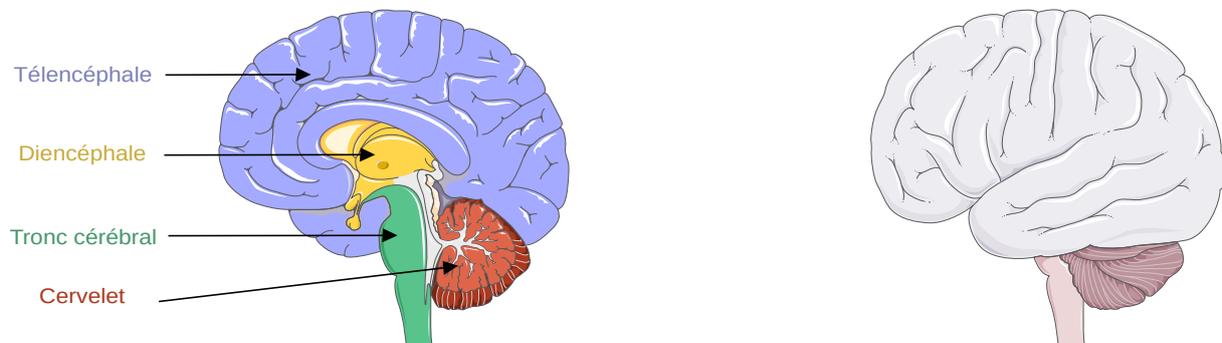


Figure 2.7 – Les différentes structures de l'encéphale (modifié par Alexandre Fenneteau, Servier Medical Art, CC BY 3.0).

Le Cervelet

Le cervelet est la structure de l'encéphale sous le télencéphale qui possède aussi une surface corticale. Le cervelet est séparé du télencéphale par la tente du cervelet qui est une membrane constituée d'une double couche de dure-mère (voir **Section 2.1.4**). Sa forme est proche de celle du télencéphale et le terme latin pour le cervelet, *cerebellum*, signifie littéralement "petit cerveau". Le cervelet est connu principalement pour être impliqué dans le contrôle moteur fin, mais joue aussi un rôle dans plusieurs autres fonctions cognitives comme le langage.

Le Tronc cérébral

Le tronc cérébral est la structure qui relie la moelle spinale et le reste de l'encéphale. La majorité des nerfs crâniens (nerfs qui sortent directement de l'encéphale) sortent du télencéphale. Cette structure est notamment impliquée dans la régulation du rythme cardiaque et de la respiration. C'est aussi au sein du télencéphale que les mouvements volontaires sont filtrés lors du sommeil.

Le système ventriculaire

Le système ventriculaire est constitué de plusieurs ventricules et de canaux (voir **Figure 2.8**) contenant du LCS (voir **Section 2.1.2**). Le système ventriculaire sécrète le

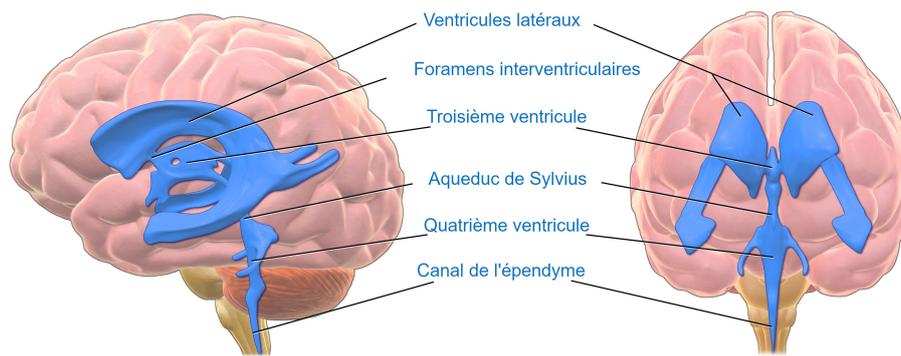


Figure 2.8 – Le système ventriculaire (modifié par Alexandre Fenneteau, Blausen Medical, CC BY 3.0).

LCS au niveau de ses plexus choroïdes et fait circuler le LCS dans le SNC.

Les structures et différents tissus du SNC ainsi que leurs potentielles altérations sont visibles et mises en évidence dans l'imagerie par résonance magnétique nucléaire.

Le système nerveux et ses structures sont visibles en IRM. L'étude des images de l'encéphale par le radiologue peut mettre en évidence des lésions et pathologies dont la sclérose en plaques.

2.2 Principes de l'IRM

L'imagerie par résonance magnétique nucléaire (IRM) est une technique d'imagerie médicale qui permet la visualisation des tissus et organes *in vivo*. L'IRM permet l'acquisition 3D ou en coupe 2D d'une partie du corps d'un patient. L'IRM est particulièrement adaptée pour visualiser les tissus mous (en opposition aux tissus osseux) notamment en imagerie cérébrale. Ce mode d'imagerie contrairement au CT-scan ou au PET-scan ne repose pas sur des rayonnements ionisants ou radioactifs, mais nécessite cependant un temps d'acquisition plus long. L'IRM se distingue des autres systèmes d'acquisition de radiologie par son principe de fonctionnement qui repose sur la résonance magnétique nucléaire (RMN).

2.2.1 Les principes de la résonance magnétique nucléaire

La résonance magnétique nucléaire

La découverte du phénomène de RMN remonte aux années 1940. Naturellement, certains noyaux atomiques, ceux qui ont un spin non nul, ont des propriétés magnétiques. C'est le cas du noyau d'hydrogène qui est utilisé en IRM conventionnelle. Les noyaux d'hydrogène, placés dans un champ magnétique B_0 suffisamment puissant et homogène, s'alignent de façon parallèle et antiparallèle avec celui-ci. Placée dans B_0 , une petite majorité de noyaux d'hydrogène s'alignent de façon parallèle et sont responsables d'une aimantation locale M_z , macroscopique (voir **Figure 2.9**). Dans la suite, nous ne nous intéresserons qu'aux protons avec les spins parallèles contribuant à M_z . Le spin des noyaux d'hydrogène dans B_0 précessent autour de B_0 à la fréquence de Larmor ω_0 sans engendrer aucun signal RMN puisque leur précession n'est pas en phase. La fréquence de Larmor s'exprime comme suivant:

$$\omega_0 = \gamma B_0, \quad (2.1)$$

où γ est le rapport gyromagnétique qui est une constante pour chaque noyau atomique.

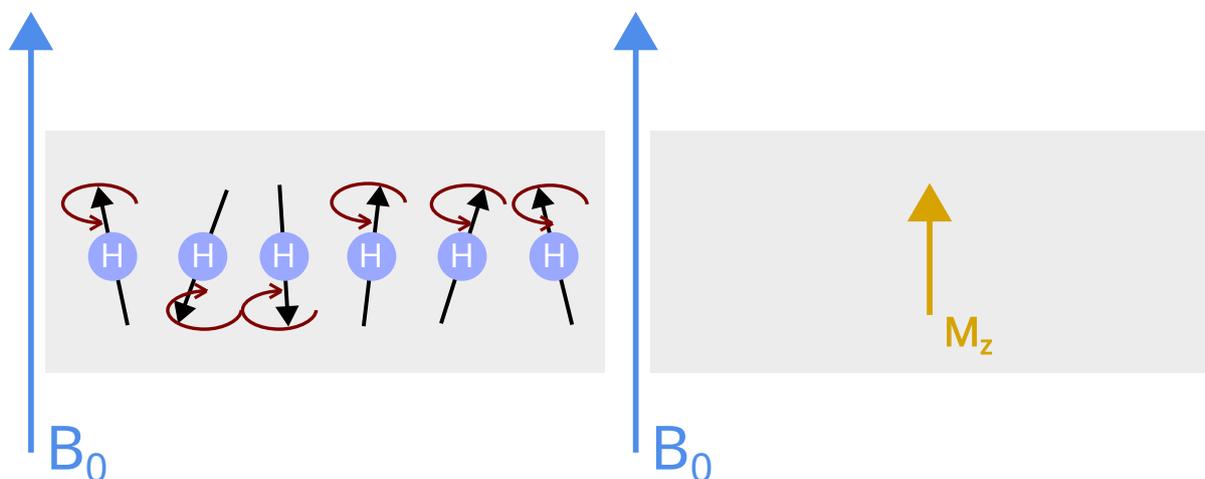


Figure 2.9 – Les spins des noyaux d'hydrogène (à gauche) responsables de l'aimantation macroscopique M_z (à droite) dans le champ magnétique B_0 .

Lorsque les atomes d'hydrogène sont soumis à un pulse de radio-fréquence (RF) de fréquence ω_0 créant un champ électromagnétique B_1 , les spins des atomes vont alors se synchroniser pour précesser progressivement de façon perpendiculaire dans l'axe xy constituant le phénomène de RMN. L'aimantation M_z va alors diminuer au profit de l'aimantation M_{xy} .

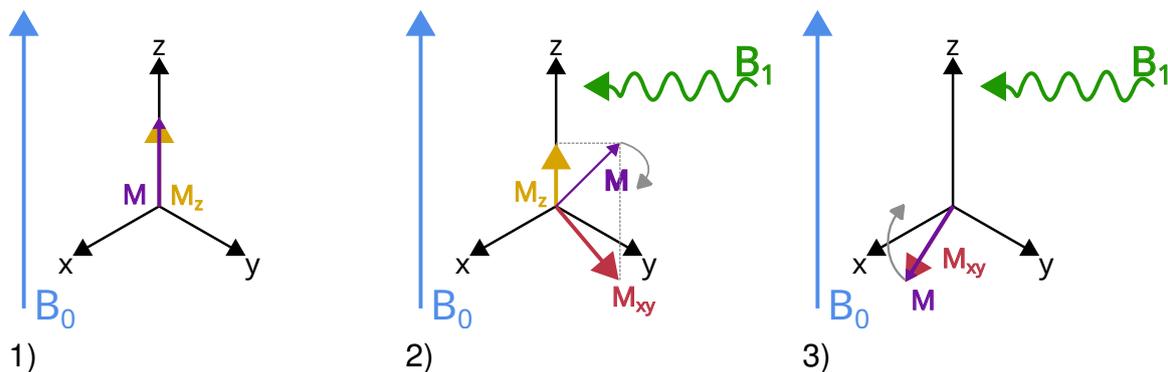


Figure 2.10 – Le phénomène RMN. Avant excitation, l'aimantation M des atomes d'hydrogène est dans l'axe z (1), après excitation B_1 , les spins sont basculés progressivement dans l'axe xy (2) jusqu'à précesser perpendiculairement à l'axe z (3).

La relaxation

Après l'arrêt de l'onde électromagnétique B_1 , les spins reviennent peu à peu à leur état d'équilibre en libérant de l'énergie sous forme d'onde électromagnétique qui peut être captée par une antenne. De façon macroscopique, il existe deux types de relaxation :

- la relaxation T1, qui est le retour de l'aimantation M_z consécutive au retour des spins dans le sens parallèle à B_0 ,
- la relaxation T2, qui est le déphasage des spins qui entraîne la disparition de l'aimantation M_{xy} .

Les deux types de relaxations sont progressifs et dépendent, dans le champ magnétique B_0 , de l'environnement chimique dans lesquels les protons sont placés. Les tissus biologiques ont des compositions chimiques différentes et ont donc des relaxations différentes. Il est donc possible d'observer un contraste de relaxation RMN dans les tissus biologiques.

Les relaxations suivent des lois de croissance et décroissances exponentielles qui s'écrivent :

$$\begin{aligned} M_z(t) &= M_{z0}(1 - e^{-t/T1}), \\ M_{xy}(t) &= M_{xy0}e^{-t/T2}, \end{aligned} \quad (2.2)$$

où M_{z0} est l'aimantation initiale dans l'axe z , M_{xy0} l'aimantation dans l'axe xy après basculement complet des spins par la radiofréquence B_1 , T1 la constante de relaxation pour que 63% de l'aimantation M_z du tissu aie été récupérée et T2 la constante de

relaxation pour que 63% de l'aimantation M_{xy} du tissu aie été perdue.

Le signal mesuré dans l'IRM dépend uniquement de l'aimantation transversale M_{xy} . Cependant, du fait des inhomogénéités de champs magnétiques microscopiques du champ B_0 , ce signal ne décroît pas selon T2, mais selon T2*. Par le biais d'un jeu de pulse RF et l'écho de spin, il est possible de récupérer directement la valeur d'aimantation transversale reflétant la vraie relaxation T2.

Il n'est possible d'accéder qu'à l'aimantation M_{xy} . Pour créer les contrastes entre les différents tissus, il est possible de jouer sur deux paramètres importants :

- le temps de répétition (TR) qui est le temps écoulé entre les *pulses* RF,
- le temps d'écho (TE) qui est le temps entre le *pulse* et le recueil du signal.

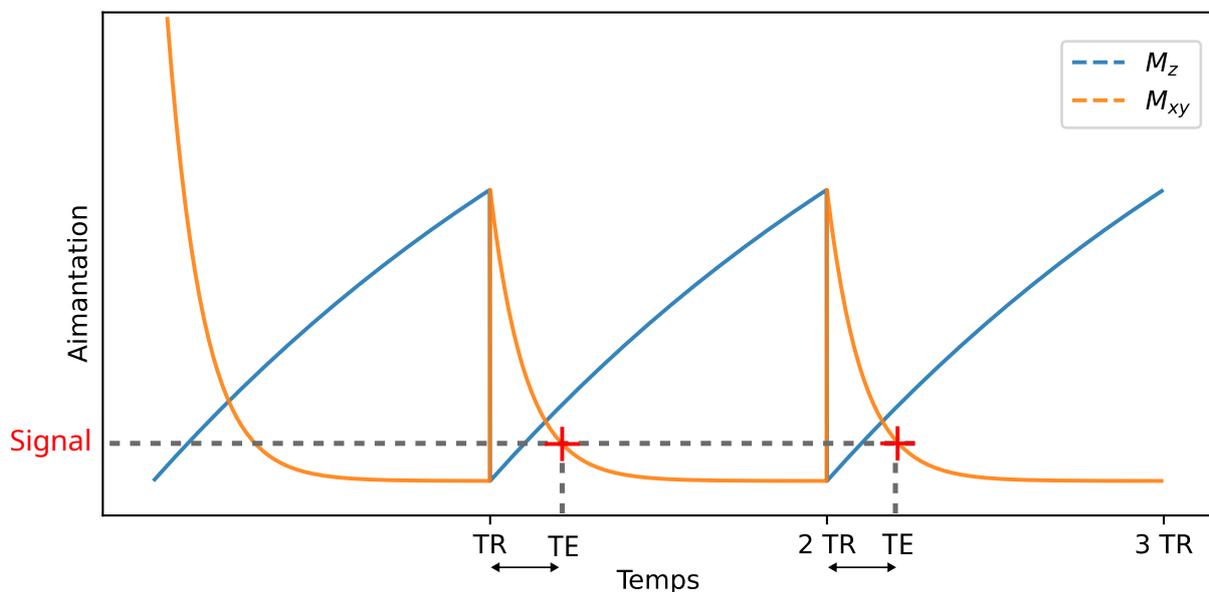


Figure 2.11 – La récupération du signal RMN en fonction des TR et TE.

Après le deuxième *pulse* RF, la valeur d'aimantation transversale maximum M_{xy0} dépend du TR et du T1. Après le temps TE, le signal qui dépend à la fois de M_{xy0} et du T2 de l'échantillon est mesuré (voir **Figure 2.11**). Il est donc possible d'ajuster le TR et le TE pour récupérer un signal plus ou moins contrasté entre les différents tissus qui dépendent de leurs constantes respectives de T1 et de T2 (voir **Figure 2.12**).

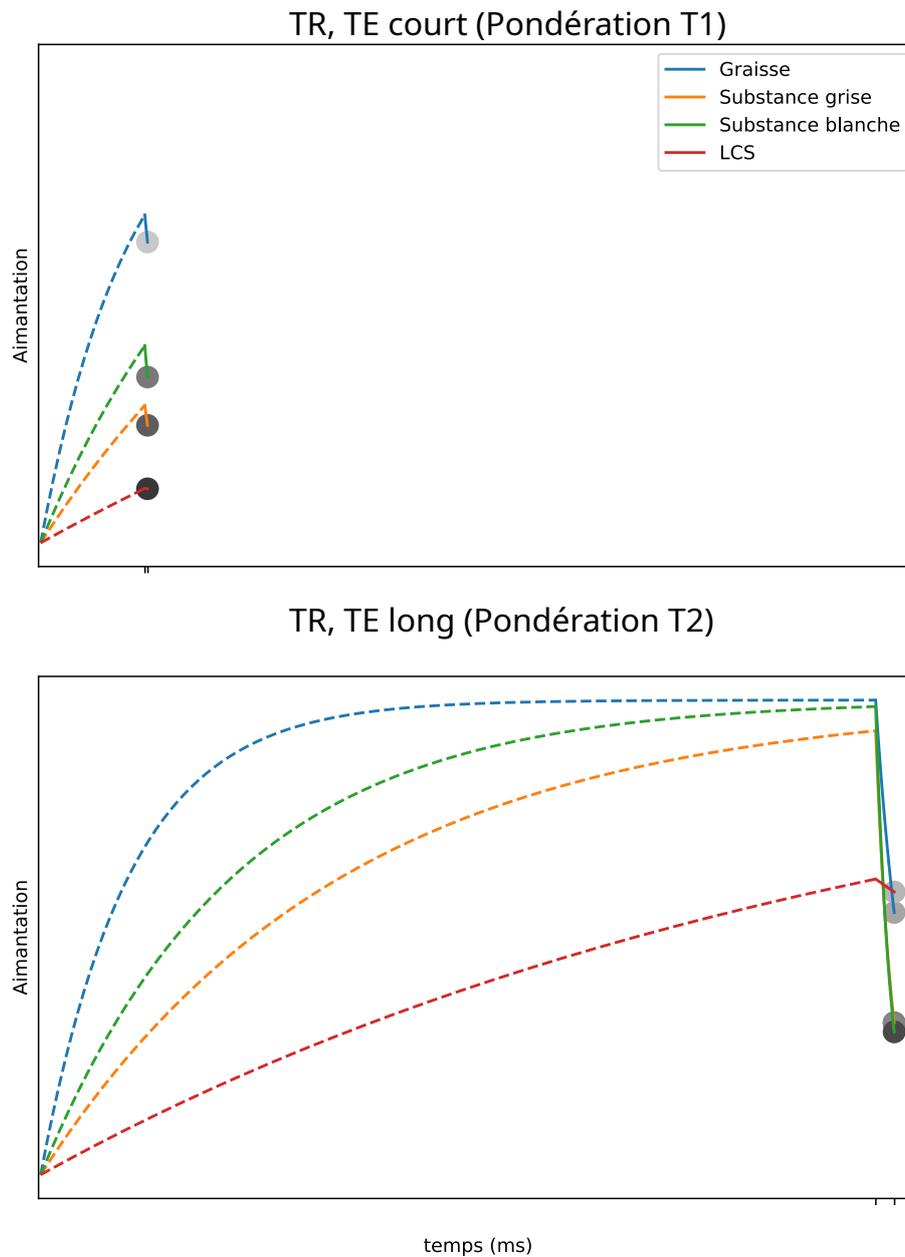


Figure 2.12 – Le contraste de différents tissus en fonction des TR et TE.

2.2.2 Acquisition 3D

En IRM, le signal est acquis spatialement en 3D avec différents gradients. Le gradient de sélection de coupe permet d'exciter séparément des coupes 3D, la force du gradient appliquée conditionne l'épaisseur de la coupe. Pour pouvoir accéder à l'information spatiale au sein de cette coupe, un gradient de codage de phase est appliqué pour discriminer horizontalement chaque point et un gradient de codage en fréquence est

appliqué pour discriminer verticalement chaque point.

Ainsi, il est possible de coder le signal reçu pour chaque coupe en 2D avec les fréquences et phases. Le signal récupéré n'est donc pas une image spatiale, mais le codage de l'image dans l'espace de Fourier. Cette image n'est pas exploitable en l'état, mais est transformable en une image 2D spatiale en appliquant la transformation de Fourier inverse.

L'acquisition en IRM est toujours un compromis entre la résolution spatiale et le temps d'acquisition. Une acquisition de très bonne résolution spatiale prendra beaucoup plus de temps qu'une acquisition à faible résolution spatiale.

2.2.3 Pondération RMN

Pondération T1

La pondération T1 est la pondération la plus commune. Elle est caractérisée par des valeurs de TR et TE courtes pour créer un maximum de contraste selon la relaxation longitudinale M_z . Cette séquence est généralement utilisée, parce qu'une bonne résolution d'acquisition peut être réalisée en un temps acceptable.

En IRM cérébrale, la séquence T1 est la séquence anatomique de référence, sur laquelle la substance blanche est plus claire que la substance grise et avec un signal du LCS très faible.

En fonction des besoins cliniques, le patient se voit injecter dans le sang un produit contrastant qui va influencer sur la relaxation T1 ou T2 des tissus dans lesquels il circule. Les produits de contrastes injectés dans le sang vont donc circuler avec celui-ci. En imagerie cérébrale, l'injection de gadolinium est effectuée notamment pour visualiser s'il y a rupture de la barrière hématoencéphalique; dans ce cas, le sang sera en contact avec la zone du cerveau concernée et produira une différence de contraste visible par le radiologue.

Pondération T2

La pondération T2 est aussi une pondération très commune dans les examens IRM, elle nécessite un TR et un TE plus longs nécessitant donc une acquisition plus longue que pour la pondération T1 avec la même résolution.

En imagerie cérébrale, la substance grise est plus claire que la substance blanche en pondération T2 (contraste inversé par rapport au T1) et le LCS est en hyperintensité. La pondération T2 permet de détecter certaines lésions cérébrales dont les lésions de SEP qui apparaissent en hyperintensité.

Pondération FLAIR

La séquence FLAIR est une séquence pondérée en T2, mais dont le signal de l'eau a été fortement atténué.

En neuroradiologie, l'atténuation du signal de l'eau la rend particulièrement adaptée pour l'analyse des lésions hyperintenses en T2 qui sont proches des ventricules (remplis de LCS aussi hyperintenses).

Pondération densité de proton

La pondération en densité de proton est une séquence avec un TR long et un TE court. Cette pondération, bien que non quantitative, donne un contraste sur les concentrations relatives en hydrogène des tissus.

L'IRM est un système d'imagerie de choix pour visualiser les tissus cérébraux. L'IRM est largement utilisée pour le suivi et le diagnostic de la sclérose en plaques puisqu'elle permet la visualisation des lésions provoquées par la maladie.

2.3 La sclérose en plaques

La sclérose en plaques (SEP) est une maladie chronique auto-immune handicapante qui affecte le SNC. Les causes de la maladie ne sont pas clairement identifiées et les symptômes de la maladie varient d'un patient à l'autre. Cette maladie affecte beaucoup de personnes dans le monde. Son suivi et diagnostic sont capitaux pour traiter au mieux

la SEP et ralentir l'évolution de la maladie.

2.3.1 Les origines

Les origines de la maladie ne sont pas encore complètement connues. Cependant, un certain nombre de facteurs ont été identifiés comme jouant un rôle important dans le développement. L'exposition au virus d'Epstein-Barr (connu notamment pour provoquer la mononucléose) double les chances de développer la maladie. De plus, un manque de vitamine D, généralement provoqué par un manque d'exposition au soleil, est une des causes identifiées pour le développement de la SEP. La consommation de tabac augmente aussi significativement les probabilités de développer la maladie. Certains facteurs génétiques ont aussi été identifiés comme ayant un rôle dans le développement de la maladie [1].

Le nombre de cas de SEP est estimé à 2 000 000 dans le monde. La prévalence varie de 0 à 300 cas pour 100 000 habitants d'une région géographique à une autre s'expliquant notamment par l'exposition au soleil [2]. De plus, il existe un ratio de 3:1 de femmes atteintes de SEP par rapport aux hommes, ce qui s'explique en partie par la consommation de tabac plus élevée chez ces dernières [1]. L'âge moyen du diagnostic de la maladie est proche des 30 ans avec une forme récurrente rémittente, la SEP est la maladie handicapante non traumatique qui affecte le plus les jeunes adultes.

La sclérose en plaques résulte de la réaction immunitaire contre la myéline de l'organisme. Chez un patient atteint par la maladie, des antigènes de la myéline sont reconnus comme étrangers par les cellules immunitaires, ce qui déclenche une réaction inflammatoire et la destruction de la gaine de myéline. La destruction de la gaine de myéline (ou démyélinisation) empêche la propagation rapide de l'influx nerveux des neurones concernés. La réaction auto-immunitaire provoque aussi la rupture de la barrière hématoencéphalique, une perte d'oligodendrocytes, une dégénération axonale et une neurodégénération pouvant aller jusqu'à la mort du neurone [3].

2.3.2 SEP et IRM

Les réactions immunitaires contre la myéline provoquent des lésions dans le SNC. Ces lésions sont visibles et peuvent être caractérisées en IRM. Il existe plusieurs localisations caractéristiques des lésions de SEP.

Lésions et pondération IRM

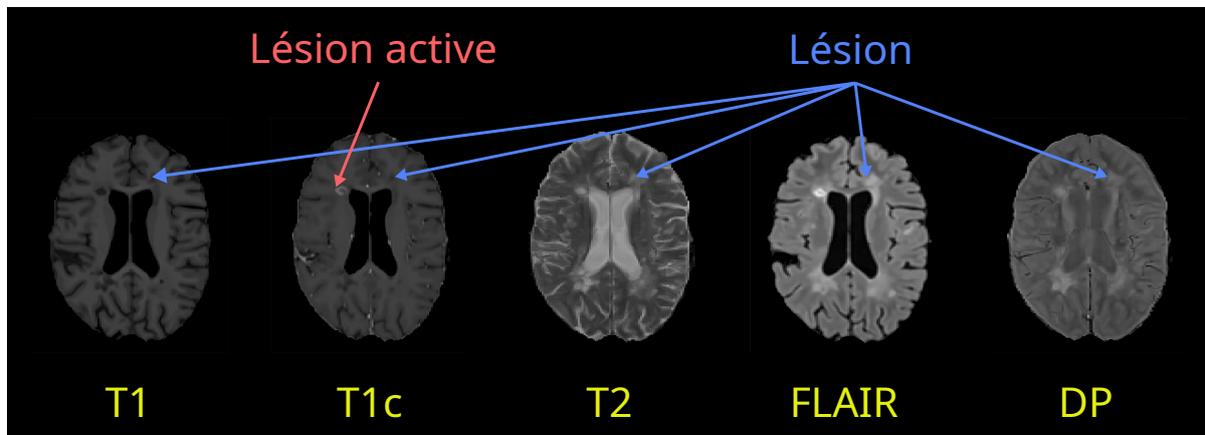


Figure 2.13 – Exemple de lésions visibles avec les pondérations IRM usuelles pour examens d'une SEP.

En pondération T1, les lésions sont visibles comme des hypointensités ou *black holes*. En pondération T2, Densité de Proton (DP), FLAIR, les lésions apparaissent en hyperintensités (voir **Figure 2.13**).

L'imagerie T1 n'est pas la pondération sur laquelle les lésions de SEP sont les plus visibles, cependant, elle est souvent effectuée dans les examens de SEP comme une séquence de référence anatomique.

La pondération T1 avec injection d'un contrastant (T1c), permet de visualiser les lésions actives comme sur la **Figure 2.13**. Ces lésions actives sont en hyperintensités, contrairement au reste des lésions en hypointensités. Les lésions actives sont des foyers qui sont inflammatoires au moment de l'examen IRM. Ces lésions entraînent la rupture de la barrière hématoencéphalique, ce qui fait pénétrer l'agent de contraste dans la zone inflammatoire uniquement.

La séquence T2 est la séquence de référence pour détecter les lésions de SEP. Cependant, la séquence FLAIR est, en pratique clinique, beaucoup plus utilisée. En effet, cette séquence atténue fortement le signal de l'eau libre (vaisseaux sanguins et LCS) tout en conservant les hyperintensités pour les lésions, ce qui résulte en un meilleur contraste et permet de délimiter plus simplement les lésions périventriculaires.

La séquence DP sert parfois en complément pour permettre une meilleure détection des lésions du tronc cérébral et du cervelet.

Localisations des lésions de SEP

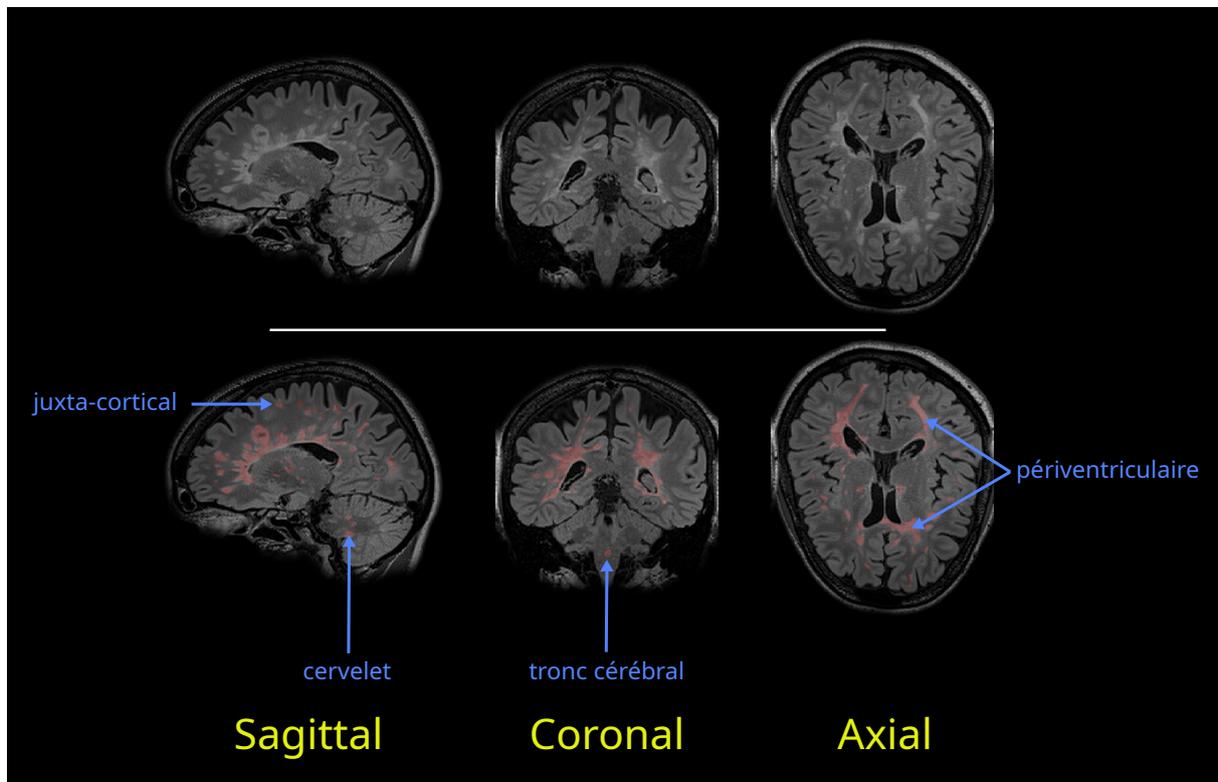


Figure 2.14 – Localisations de lésions de SEP dans l'encéphale : coupes IRM FLAIR d'un patient atteint de SEP en haut avec segmentation des lésions en rouge en bas.

Les lésions de SEP peuvent se situer à différents endroits du SNC (exemples sur la **Figure 2.14**). Il existe 5 zones particulières :

- la zone périventriculaire, dans la substance blanche du télencéphale proche des ventricules,
- la zone juxtacorticale, dans la substance blanche du télencéphale proche du cortex,
- la zone corticale, dans le cortex (très difficile à distinguer),
- la zone infratentorielle, sous la tente du cervelet, c'est-à-dire cervelet et tronc cérébral,
- la moelle spinale.

Les lésions de SEP sont majoritairement situées dans la substance blanche qui est le tissu contenant le plus de myéline. Ces lésions sont particulièrement visibles en imagerie FLAIR. Les lésions périventriculaires dites "en crête de coq" en coupe sagittale (voir **Figure 2.14**) sont particulièrement caractéristiques de la maladie.

2.3.3 Les formes

Un patient atteint de SEP peut traverser des épisodes de poussée durant de plusieurs jours à plusieurs semaines pendant lesquelles des signes neurologiques peuvent apparaître tels qu'une baisse de la vision, des troubles sensitifs ou moteurs, perte de l'équilibre, etc.

Quatre différentes formes de SEP ont été définies par les neurologues [4]. Ces formes sont majoritairement définies en fonction de la présence de poussée, de la rémission des symptômes ou de l'apparition de nouveaux signes neurologiques.

Syndrome cliniquement isolé

Le syndrome cliniquement isolé réfère au premier épisode de démyélinisation inflammatoire dans le SNC qui peut conduire à la SEP.

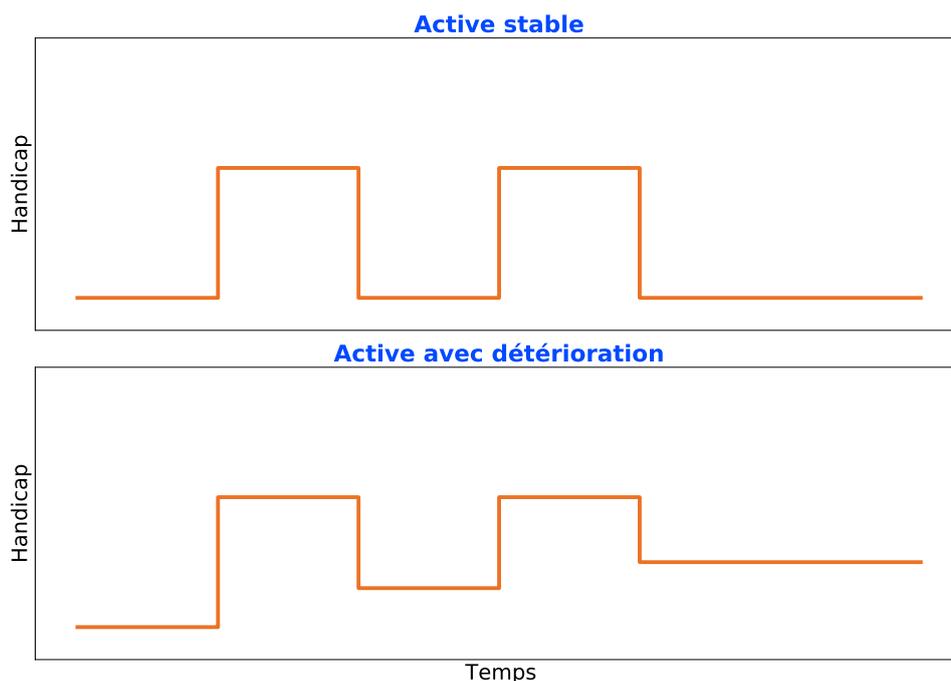


Figure 2.15 – Exemples d'évolution de forme récurrente rémittente.

Forme récurrente rémittente

La forme récurrente rémittente est caractérisée par des épisodes de poussée avec une récupération totale ou partielle sans progression apparente de la maladie (voir

Figure 2.15). Elle peut être stable sans augmentation du handicap après une poussée ou bien, peut se détériorer avec augmentation du handicap après une poussée. Cette forme peut être active avec des preuves de nouvelles rechutes, de nouvelles lésions rehaussées par agent contrastant en T1 (voir **Section 2.2.3**) ou des lésions qui s'élargissent où apparaissent en T2 (voir **Section 2.2.3**) en IRM. S'il n'y a pas de preuve de rechute la forme est dite inactive.

Forme progressive primaire

La forme primaire progressive réfère à la détérioration constante de la fonction neurologique dès l'apparition des symptômes sans que le patient n'ait précédemment développé une forme récurrente rémittente (voir **Figure 2.16**).

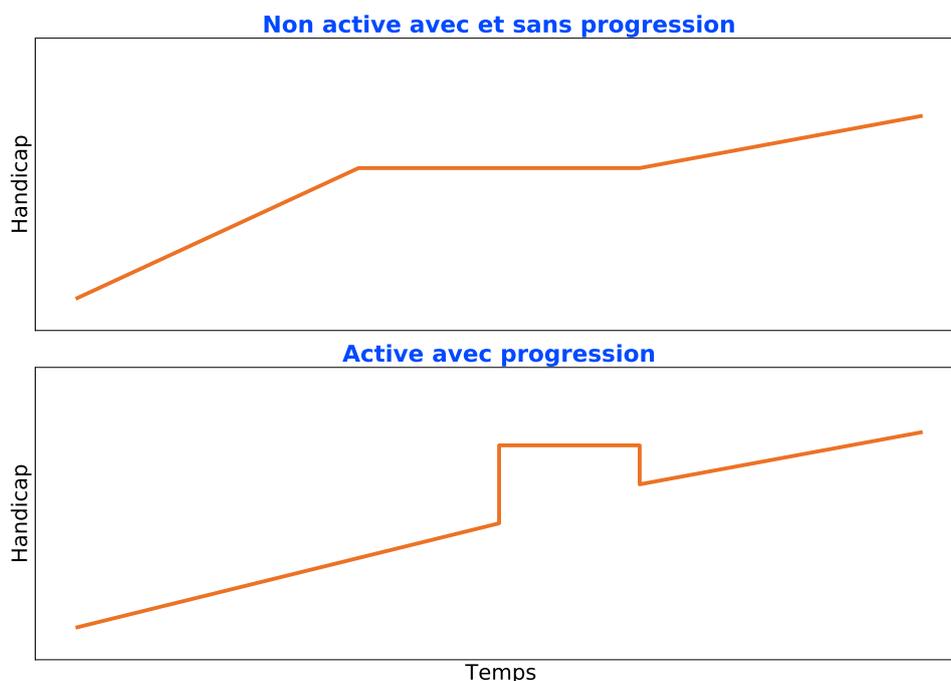


Figure 2.16 – Exemples d'évolution de forme progressive primaire.

Cette forme peut être avec progression s'il y a des preuves de détérioration de la maladie sur une mesure objective des changements (par exemple test de progression du handicap) ou sans progression. Comme la forme récurrente rémittente, elle peut être active ou inactive.

Forme progressive secondaire

La forme progressive secondaire suit la forme récurrente rémittente avec une détérioration constante de la fonction neurologique avec ou sans poussées (voir **Figure 2.17**). Cette forme aussi peut être avec progression avec ou sans activité.

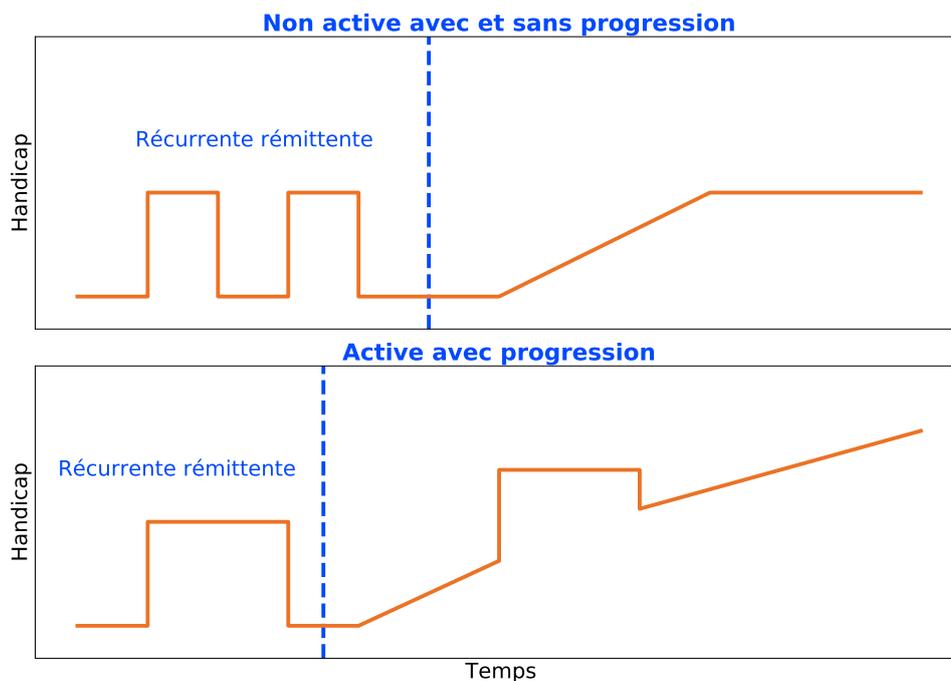


Figure 2.17 – Exemples d'évolution de forme progressive secondaire.

2.3.4 Le diagnostic et le suivi

L'examen IRM a un très grand rôle dans le diagnostic et le suivi de la SEP. Les consensus de médecins internationaux s'accordent sur le fait qu'une IRM cérébrale devrait être réalisée chez tous les cas de SEP suspectés.

L'examen IRM permet de mettre en évidence, de localiser et de voir l'apparition dans le temps des lésions de SEP. Deux critères sont particulièrement importants la dissémination spatiale (DIS) et la dissémination temporelle (DIT). La DIS se définit par la présence de lésions T2 dans plus de deux localisations alors que la DIT se définit par l'apparition de nouvelles lésions dans le temps.

Le diagnostic de la SEP se base sur les critères diagnostics de McDonald révisés [5] résumés dans le **Tableau 2.1**.

Présentation clinique	Données supplémentaires nécessaires pour le diagnostic de la SEP
Au moins 2 poussées et preuve d'au moins 2 lésions	Aucune
Au moins 2 poussées et preuve d'une lésion	DIS démontrée par une nouvelle poussée clinique sur un autre endroit du SNC ou par IRM.
1 poussée et preuve d'au moins 2 lésions	DIT démontrée par une nouvelle poussée ou par IRM ou Bandes oligoclonales spécifiques du LCS.
1 poussée et preuve d'une lésion	DIS démontrée par une nouvelle poussée clinique sur un autre endroit du SNC ou par IRM ET DIT démontrée par une nouvelle poussée ou par IRM ou Bandes oligoclonales spécifiques du LCS.

Table 2.1 – Critères diagnostics de McDonald révisés (2017) [5].

Le suivi de la maladie et notamment le diagnostique de la forme de la maladie s'appuie largement sur l'IRM en plus de signes cliniques.

Le suivi et la classification de la forme de la maladie sont particulièrement importants puisque le traitement médicamenteux donné au patient par le neurologue est conditionné par ces deux éléments [1]. Le traitement médicamenteux vise à ralentir la progression de la maladie et à protéger le plus possible du patient des rechutes ou de la progression du handicap, mais comme tout traitement il doit être soigneusement choisi pour donner au patient la meilleure balance bénéfice-risque en fonction de son profil clinique. L'analyse de l'examen IRM a donc un impact important dans la prise en charge du patient par le neurologue et par conséquent dans le ralentissement de sa maladie.

Le développement des aides aux diagnostics dans l'imagerie médicale et notamment les outils de segmentation automatique des lésions de SEP ont le potentiel de pouvoir fournir aux praticiens des métriques et des indices supplémentaires sur les examens cliniques qui pourront aider la prise en charge du patient vers une médecine de plus en plus personnalisée.

De nombreux algorithmes de segmentation automatique des lésions de SEP ont été développés par la communauté scientifique et plus de 100 articles scientifiques ont été publiés sur la segmentation des lésions de SEP faisant usage l'apprentissage profond [6].

2.4 Apprentissage profond et réseau de neurones convolutifs

L'apprentissage profond est une technique d'apprentissage machine qui a connu un fort essor et engouement les dix dernières années et qui repose sur les réseaux de neurones artificiels.

L'emploi de plus en plus massive de cette technique s'explique par les progrès matériels, notamment au niveau des capacités de parallélisation haute performance permises par les derniers processeurs graphiques. L'accessibilité des données croissantes avec leurs collectes de plus en plus simples, une disponibilité des ressources internet et l'accessibilité de la mémoire physique ont aussi joué un rôle important dans la démocratisation de ce type d'approche extrêmement gourmande en données. Le succès de l'apprentissage profond s'explique par un gain substantiel en performance dans un grand nombre de disciplines et de tâches variées.

L'apprentissage profond connaît cependant un certain nombre de limitations, notamment en ce qui concerne leur complexité. Les structures de données engendrées sont si complexes qu'il est difficile de comprendre son analyse malgré les performances. De par leur complexité, les algorithmes d'apprentissage profond sont extrêmement gourmands en temps de calcul et nécessitent des machines toujours plus performantes. Pour créer un modèle performant, il faut généralement disposer d'un grand nombre de données annotées qui sont coûteuses, potentiellement bruitées et assez représentatives de la réalité. Toutes ces contraintes sont autant de champs d'investigation et de recherches pour les chercheurs d'aujourd'hui.

2.4.1 Bref historique

L'origine des réseaux de neurones artificiels remonte aux années 1943 avec l'article de McCulloch et Pitts qui formalisent le fonctionnement d'un neurone biologique (voir **Section 2.1.1**) et créent le modèle du neurone formel [7]. Le neurone formel possède des entrées (dendrites) et calcule une réponse en fonctions de ses entrées et de ses poids synaptiques en sortie. C'est en 1947 que voit le jour la théorie du renforcement [8] stipulant que les neurones peuvent renforcer leurs liens et s'activer ensemble. Basé sur le modèle du neurone formel et la théorie du renforcement, le modèle du perceptron [9] est proposé par Rosenblatt *et al.* . Le perceptron est le premier modèle de neurone artificiels informatique capable d'apprendre par expérience en utilisant l'optimisation

convexe. Le premier réseau de neurones artificiels voit le jour en 1986 avec le premier perceptron multicouche [10] par Rumelhart *et al.* . En 2012, la recherche scientifique sur le sujet, l'accès à une base de données conséquente (ImageNet [11]) et le matériel informatique étaient assez avancés pour donner lieu à la création du AlexNet [12], qui dépassa significativement les performances de l'état de l'art (de plus de 15% sur le taux d'erreur) dans la classification d'images avec plus de 1000 catégories différentes. Depuis, le nombre de recherches tirant parti des réseaux de neurones artificiels a explosé avec toujours plus d'applications et de performances.

2.4.2 Le modèle du réseau de neurones

Le modèle classique du réseau de neurones est le réseau de neurones dense ou le perceptron multicouche. Dans ce type de réseau de neurones, le signal d'entrée passe par des couches denses qui réalisent des combinaisons linéaires entre tous les neurones de la couche précédente pour calculer la couche neuronale suivante. En pratique, les couches denses sont suivies d'une fonction d'activation qui permet d'introduire de la non-linéarité dans le réseau de neurones et ainsi pouvoir approximer n'importe quelle fonction continue [13].

La couche dense est chargée d'apprendre une combinaison linéaire entre tous les neurones de la couche précédente représentés par la matrice X en apprenant la matrice de poids W et la matrice de biais b . La sortie de la couche dense est activée par une fonction F_σ pour calculer la couche suivante Z avec σ une fonction d'activation non linéaire $\sigma : \mathbb{R} \rightarrow \mathbb{R}$, $x \mapsto \sigma(x)$ et F_σ la fonction qui applique σ terme à terme tel que $F_\sigma : \mathbb{R}^m \rightarrow \mathbb{R}^m$, $(x_i)_i \mapsto (\sigma(x_i))_i$. Le calcul de la couche neuronale Z par la couche dense est réalisé de la façon suivante :

$$Z = \begin{bmatrix} Z_{1,1} \\ \vdots \\ Z_{m,1} \end{bmatrix}, \quad W = \begin{bmatrix} W_{1,1} & \cdots & W_{1,m} \\ \vdots & \ddots & \vdots \\ W_{n,1} & \cdots & W_{n,m} \end{bmatrix}, \quad X = \begin{bmatrix} X_{1,1} \\ \vdots \\ X_{n,1} \end{bmatrix}, \quad b = \begin{bmatrix} b_{1,1} \\ \vdots \\ b_{m,1} \end{bmatrix},$$

$$Z = F_\sigma(W^T \cdot X + b)$$

$$Z = F_\sigma \left(\begin{bmatrix} W_{1,1} & \cdots & W_{n,1} \\ \vdots & \ddots & \vdots \\ W_{1,m} & \cdots & W_{n,m} \end{bmatrix} \begin{bmatrix} X_{1,1} \\ \cdots \\ X_{n,1} \end{bmatrix} + \begin{bmatrix} b_{1,1} \\ \vdots \\ b_{m,1} \end{bmatrix} \right).$$

En théorie, un réseau simple avec deux couches denses comme dans l'exemple de la **Figure 2.18** (avec un différent nombre de neurones) est suffisant pour approximer n'importe quelle fonction continue [13]. Cependant, en ajoutant des couches, il est possible de modéliser des fonctions plus complexes avec les mêmes ressources de calcul [14]. Dans un réseau de neurones, la première couche (généralement ignorée) est la couche d'entrée, elle est suivie par plusieurs autres couches cachées jusqu'à la couche de sortie comme illustrée dans la **Figure 2.18** qui représente un réseau de neurones artificiels avec deux couches denses. Un réseau de neurones est considéré comme profond à partir d'une couche cachée, mais en contient généralement plus.

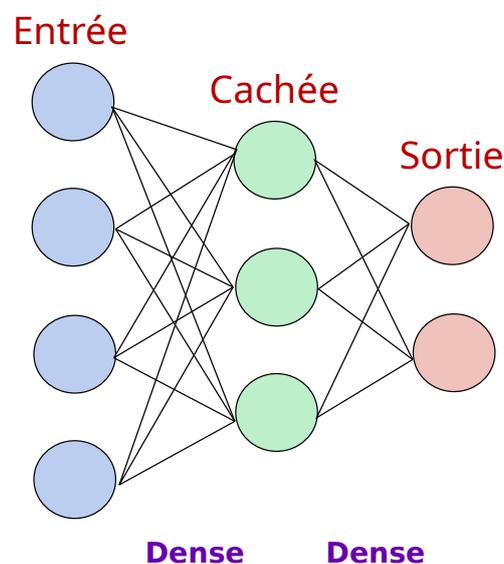


Figure 2.18 – Réseau de neurones dense ou perceptron multicouche simple. Les trois couches de neurones (d'entrée, cachée et de sortie) sont liées entre elles par des connexions denses.

L'AlexNet et la plupart des réseaux de neurones en vision par ordinateur sont des réseaux de neurones convolutifs. Ces réseaux de neurones sont particulièrement adaptés pour l'analyse d'image et sont le produit de la fusion des réseaux de neurones denses et des filtres de convolutions largement employés en vision par ordinateur.

2.4.3 Les réseaux de neurones convolutifs

Le produit de convolution

Le produit de convolution est défini comme une intégrale du produit de deux fonctions f et g après que l'une soit retournée et décalée à l'infinie et s'écrit en général

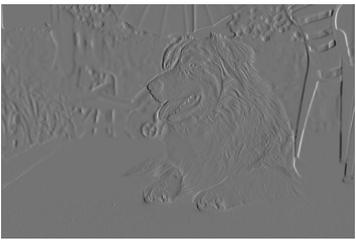
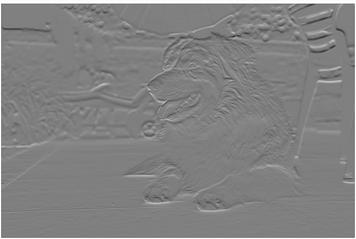
Image I	Filtre h	Caractéristique F
	$Sobel_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$	
	$Sobel_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$	

Table 2.2 – Exemples de filtrage par produit de convolution avec les filtres de Sobel.

$f * g$ et s'exprime comme :

$$(f * g)(x) = \int_{-\infty}^{+\infty} f(x-t)g(t)dt = \int_{-\infty}^{+\infty} f(t)g(x-t)dt \quad (2.3)$$

Le produit de convolution est utilisé dans de nombreux domaines et notamment en traitement d'images. En traitement d'images on considère l'image I de taille m, n sous sa forme matricielle et un filtre h de taille o, p impaires usuellement de même valeur entre 3 et 7.

$$I_{m,n} = \begin{bmatrix} I_{1,1} & I_{1,2} & \cdots & I_{1,n} \\ I_{2,1} & I_{2,2} & \cdots & I_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ I_{m,1} & I_{m,2} & \cdots & I_{m,n} \end{bmatrix}, \quad h_{o,p} = \begin{bmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,p} \\ h_{2,1} & h_{2,2} & \cdots & h_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ h_{o,1} & h_{o,2} & \cdots & h_{o,p} \end{bmatrix}.$$

Dans ce contexte, le produit de convolution d'une image I et d'un filtre h peut s'écrire sous la forme de l'**Equation 2.4**.

$$F_{x,y} = h * I_{x,y} = \sum_{i=1}^o \sum_{j=1}^p h_{i,j} I_{x+i-(o+1)/2, y+j-i-(p+1)/2}. \quad (2.4)$$

avec F le résultat de la convolution du filtre h et d'image I pour chaque pixel de I . F est communément appelé carte de caractéristiques.

La convolution d'une image et d'un filtre est largement utilisée en traitement d'images, car c'est une opération réalisée non pas sur un pixels, mais sur un voisinage

de pixel qui se rapproche du traitement visuel primaire. Le but du filtrage des images est d'obtenir des images de plus haut niveau, donnant des indications sémantiques sur l'image d'intérêt par exemple l'extraction des contours (*cf* **Tableau 2.2**).

Il existe de nombreuses opérations en traitement d'images qui recourent au produit de convolution avec différents filtres en fonction des caractéristiques que l'on souhaite extraire. Une approche en traitement d'images est de prendre ces cartes de caractéristiques pour résumer les images, de les concaténer, de les résumer et de les envoyer en entrée d'un modèle d'apprentissage machine dit conventionnel pour prédire une caractéristique de haut niveau d'une image. Ce genre de modèle conventionnel est rapide et interprétable, mais est limité par la définition de ses caractéristiques par un expert. Pour répondre à la question "Y-a-il une voiture sur la photo ?", ce qui constitue un problème de classification, il est difficile pour l'expert de fournir des éléments simples et concrets pour expliquer comment reconnaître une voiture. Bien sûr, on peut dire qu'une voiture possède des roues, mais cela pose le problème de reconnaître les roues qui n'est pas trivial [15]. De plus, dans l'image, les roues peuvent être masquées par un quelconque objet. Il est donc difficile de définir des attributs experts pour un certain nombre de tâches, notamment des tâches des visions. Les réseaux de neurones convolutifs (RNCs) résolvent ce problème en apprenant directement les filtres de convolutions pour réaliser leur prédiction.

Un RNC est un réseau de neurones qui contient des couches de convolutives, c'est-à-dire des couches dans lesquels des filtres de convolution sont appris pour effectuer la tâche de prédiction. Le plus souvent, plusieurs couches convolutives se suivent successivement, pour suivre le modèle biologique du traitement visuel en cascade effectué par le cortex visuel. Cela a pour but de créer des cartes de caractéristiques à partir des cartes de caractéristiques précédentes afin d'obtenir des analyses toujours plus haut niveau, vers des tâches visuelles toujours plus complexes. Les filtres de convolutions appris sont capables de débruiter l'entrée, et d'extraire des caractéristiques de toujours plus haut niveau vers la prédiction finale.

Les RNCs et les réseaux de neurones en général nécessitent une grande quantité de données et une architecture adaptée pour créer un modèle performant et efficace. De plus, les filtres de convolutions, étant successifs et appris par le RNC, sont très difficiles voire impossible à interpréter pour un humain. En effet, les réseaux de neurones créent leur propre représentation des données par entraînement sur celles-ci, cette représentation n'est en pratique jamais la même que celle d'un humain, ce qui rend ces réseaux de neurones difficilement explicables et leur vaut leur réputation de boîtes

noires.

En traitement d'images, un certain nombre de types de couches de réseaux de neurones sont populaires avec chacune des raisons et fonctions différentes.

Éléments usuels d'architectures des réseaux de neurones convolutifs

Couche convolutive

La couche convolutive est la couche de réseau de neurones qui apprend les filtres et effectue le filtrage des entrées par produit de convolution. La couche convolutive prend en entrée les cartes de caractéristiques ou l'entrée du réseau, calcule ses propres cartes de caractéristiques à partir de toutes ces entrées et les retourne en sortie. Dans un réseau de neurones convolutif, les couches convolutives sont suivies d'une fonction d'activation comme les couches denses pour introduire une source de non-linéarité aux réseaux de neurones.

Couche de sous-échantillonnage

Les couches de sous-échantillonnage aussi appelées *pooling* en anglais sont des couches qui sont très largement utilisées dans les RNCs. Ces couches sont la plupart du temps des fonctions fixes et non apprises. Ces couches permettent au réseau de neurones d'avoir une certaine invariance aux déformations et perturbations de l'entrée. Ces couches sélectionnent localement l'information à conserver et les agrègent. Les couches de sous-échantillonnage rendent une représentation plus robuste et diminuent la taille de l'information, ce qui diminue la charge de calcul [16]. Les deux couches de *pooling* les plus courantes sont les couches d'échantillonnage par maximum (*max pooling*) et d'échantillonnage moyen (*average pooling*). La couche d'échantillonnage par maximum sélectionne la valeur maximale locale et l'agrège avec le reste des valeurs maximales sélectionnées. La couche d'échantillonnage moyen moyenne les valeurs locales pour les agréger ensuite au reste des valeurs moyennées.

Couche de normalisation

Les couches de normalisation servent à normaliser l'information qui est calculée au sein du réseau de neurones. La couche de normalisation la plus commune est la normalisation par lot ou *batch normalization* [17]. La normalisation par lot consiste à estimer tout au long de l'entraînement la moyenne et l'écart type de la couche d'entrée et de sortir l'information normalisée (après avoir soustrait la moyenne et divisé par l'écart type). En pratique, la normalisation par lot permet une convergence plus rapide, de diminuer l'impact de l'initialisation et peut aider à obtenir de meilleures performances.

Couche d'abandon

La couche d'abandon, *dropout* en anglais transmet à la couche suivante l'information d'entrée dont une partie a été oubliée. En pratique, les couches d'abandon mettent aléatoirement des valeurs d'entrée à zéro avant de la transmettre à la couche suivante. Cette perturbation évite au réseau de neurones de surapprendre en supprimant aléatoirement des connexions au sein du réseau de neurones ce qui a pour effet d'éviter à la prédiction de ne reposer que sur quelques connexions [18].

Connexions raccourcis

Le plus fréquemment, une couche de réseau de neurones prend en entrée la sortie de la couche précédente et envoie sa sortie à la couche suivante. Cependant, dans certains cas, les réseaux de neurones sont connectés en entrée ou en sortie à plusieurs couches ou des couches plus lointaines.

Certains réseaux de neurones utilisent des connexions en saut ou *Skip-connections* ces connexions consistent à concaténer ou additionner les sorties d'une couche à la sortie de la couche distante. Ce type de connexion peut se faire à plus ou moins grande distance en fonction de la régularisation à effectuer.

Les connexions résiduelles ont été introduites dans l'architecture du Resnet [19]. Une connexion résiduelle connecte la sortie couche N_{-2} à la sortie de la couche N_0 en les additionnant. En pratique, l'emploi des connexions résiduelles permet de construire des réseaux de neurones plus profonds avec une amélioration des performances. En effet, on observe en général qu'à partir d'une certaine profondeur de réseau de neurones, on n'observe plus d'amélioration de performance. L'hypothèse avancée pour ce phénomène est que ces connexions accélèreraient la convergence et réduirait le taux d'erreur [19].

Plus récemment, les connexions denses sont des connexions en saut courtes par concaténation. Cela a pour effet d'améliorer la propagation du gradient à travers le réseau de neurones et de diminuer le nombre de paramètres à apprendre par rapport aux connexions résiduelles [20].

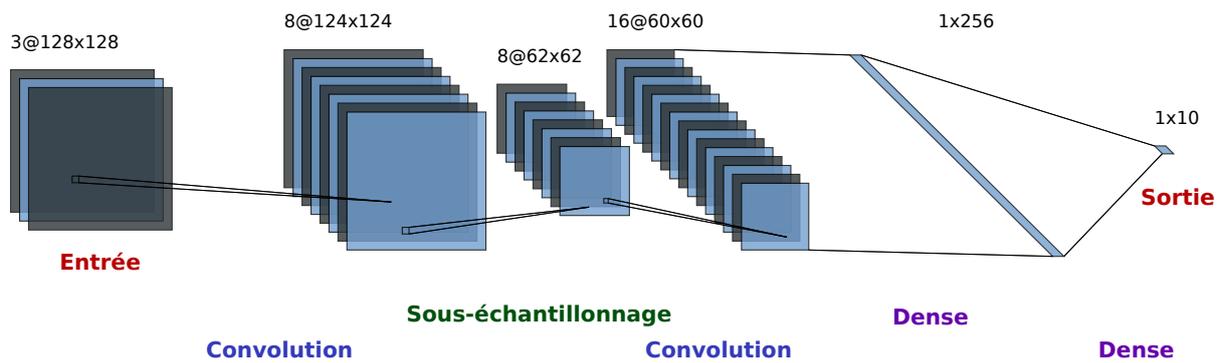


Figure 2.19 – Architecture classique simplifiée de RNC pour une tâche de classification. Cette architecture contient des couches convolutives, des couches de sous-échantillonnage et des couches denses.

Architecture classique de RNC

Les RNCs sont généralement constitués d'une alternance de couches convolutives et de sous-échantillonnage suivis par des couches denses jusqu'à la prédiction en sortie comme illustré dans la **Figure 2.19**. Cette architecture classique de classification est adaptable de plusieurs façons en ajoutant plusieurs couches de convolutions et de sous-échantillonnage ou en alternant avec des couches de régularisation comme de normalisation ou d'abandon. Les réseaux les plus profonds (avec de nombreuses couches convolutives) contiennent généralement des couches de régularisation et des connexions raccourcies pour toujours plus gagner en performance et en pouvoir de généralisation.

Apprentissage d'un réseau de neurones

L'entraînement d'un réseau de neurones est généralement paramétré par un certain nombre de valeurs fixées à l'avance appelées hyperparamètres. Il n'y a pas de paramétrage unique pour l'apprentissage d'un réseau de neurones et la nature complexe des réseaux de neurones rend difficile de connaître à l'avance le meilleur ensemble d'hyperparamètres et la meilleure façon d'apprendre la tâche d'intérêt. Le principe de l'apprentissage est d'apprendre de façon complètement autonome un réseau de neurones complexe à partir uniquement des données d'apprentissage fournies. Pour estimer les meilleurs choix architecturaux et d'apprentissage, une phase de test empirique est généralement nécessaire nécessitant des ressources de temps, de calcul et d'énergie.

Lors de la conception d'un algorithme d'apprentissage profond, il existe beaucoup de points de détails importants auxquels il faut penser et qu'il faut paramétrer.

Les données d'apprentissage

Le choix de l'ensemble de données, sa taille, la qualité de données et des annotations et sa diversité sont des facteurs importants pour entraîner le modèle sur des données les plus proches possible de la réalité et ainsi garantir une analyse la plus complète et générale possible.

Les données d'apprentissages sont en général séparées en trois ensembles de :

- l'ensemble d'entraînement qui est l'ensemble utilisé pour apprendre les données,
- l'ensemble de validation qui est l'ensemble utilisé uniquement pour ajuster les hyperparamètres et n'est pas appris par le modèle,
- l'ensemble de test qui est l'ensemble sur lequel le modèle n'a pas été entraîné et sur lequel les hyperparamètres n'ont pas été ajustés. Cet ensemble permet d'évaluer le modèle sur des données complètement extérieures à l'entraînement.

Dans certains cas, lorsque très peu de données sont disponibles, les ensembles de validation et de test sont parfois fusionnés. Les ensembles de validation et d'entraînement peuvent aussi être régulièrement mélangés lors de l'apprentissage pour fusionner la base d'entraînement et de validation et ainsi réduire le besoin en données.

Le prétraitement des données

Les données d'entrées sont, en général, prétraitées afin de les rendre les plus comparables possibles et les plus adaptées à la tâche d'apprentissage. Les prétraitements varient beaucoup en fonction du domaine d'applications et font appel à des connaissances spécifiques. Un des prétraitements les plus communs est de normaliser les valeurs d'entrées pour les ramener dans un intervalle proche de $[0; 1]$ ou $[-1; 1]$.

L'augmentation des données

L'augmentation des données consiste à altérer les données d'entrées aléatoirement afin d'entraîner le réseau de neurones avec un nombre artificiellement plus grand d'exemples différents. Ces transformations peuvent être l'addition de bruit ou des transformations qui ne détériorent pas la sémantique de la donnée d'entrées. L'augmentation des données vise à rendre l'apprentissage plus généralisable et plus robuste.

L'envoi des données

L'envoi des données se fait généralement par lot, ou *batch*. Le réseau de neurones apprend par lot, la taille du lot a donc une influence directe sur l'apprentissage. Si un lot est grand, chaque étape d'apprentissage sera générale. Si le lot est trop grand, le modèle risque d'apprendre une tâche trop généraliste évitant la convergence de l'apprentissage. Si le lot est trop petit, l'apprentissage ne se fera que sur des cas particuliers au risque de ne pas se généraliser.

Selon les applications, les données peuvent être envoyées entières ou partielles. Dans le cas des RNCs, pour le traitement d'images il est possible, en fonction de l'application, d'envoyer l'image entière ou bien uniquement une partie de l'image ou patch. Le recours aux patches peut contribuer à augmenter artificiellement les données et à diminuer les besoins en ressources matérielles pour l'apprentissage.

L'architecture du réseau de neurones

L'enchaînement et l'ordre des différentes couches sont importants à l'instar de la paramétrisation de chaque couche. Pour chaque couche apprise, le nombre de sorties doit être fixé influant directement sur le nombre de paramètres à apprendre pour la couche en question. Généralement, chaque couche dense ou convolutive d'un réseau de neurones est suivie d'une fonction d'activation non linéaire. Elle peut aussi être suivie de différentes couches de régularisation pour assurer la convergence de l'apprentissage.

L'initialisation du réseau de neurones

L'initialisation des poids d'un réseau de neurones conditionne la convergence de l'apprentissage. Les poids des réseaux de neurones sont généralement initialisés aléatoirement de manière uniforme pour faciliter la convergence [21]. Cependant, il est parfois judicieux d'initialiser les poids d'un réseau de neurones avec les poids d'un réseau de neurones déjà entraîné sur une tâche proche pour commencer l'entraînement avec un réseau déjà capable d'analyser l'entrée en question, c'est l'apprentissage par transfert. Cette pratique a pour but de diminuer le besoin en apprentissage du réseau de neurones et donc, de diminuer le besoin en données ou en ressources de calcul.

La tâche et le coût

Lors d'un apprentissage profond, le réseau de neurones artificiels est entraîné sur une ou plusieurs tâches spécifiques de façon supervisées. À chaque itération, les prédictions du réseau de neurones sont comparées avec la vérité terrain créée par un expert ou bien calculée par un algorithme ou encore être l'entrée elle-même. La comparaison entre la vérité terrain et la prédiction est réalisée en calculant le coût qui est une fonction dérivable mesurant la différence entre deux échantillons. La fonction de coût est capitale puisqu'elle doit rendre compte de façon la moins biaisée possible ce qui ne va pas dans la prédiction et ce qui doit être appris.

La tâche d'un réseau de neurones peut être multiple ou divisée en sous-tâche avec une ou plusieurs fonctions de coût.

L'optimisation de l'apprentissage

Lors de l'apprentissage d'un réseau de neurones, la comparaison entre la prédiction d'un lot et la vérité terrain est évaluée avec la fonction de coût. Le coût est lui-même dérivé en chaîne par rapport à chaque poids du réseau de neurones. Une fois le gradient de coût calculé pour chaque poids du réseau de neurones, chaque poids est corrigé par rapport à ce gradient avec un optimiseur. Le rôle de l'optimiseur est de rendre la descente de gradient la plus efficace possible. Le taux d'apprentissage ou *learning rate* est lui-même un des paramètres clefs de la plupart des optimiseurs puisqu'il conditionne la sensibilité de la correction des poids du réseau de neurones par rapport au gradient.

L'apprentissage profond, bien que capable d'apprendre par lui-même, nécessite donc, un certain nombre d'ajustements parfois empiriques pour pouvoir fonctionner de façon efficace.

Les réseaux de neurones artificiels et particulièrement les RNC peuvent être mis en place pour réaliser des outils de segmentation automatique des lésions de SEP pour aider le clinicien dans une tâche quotidienne vers un meilleur suivi du patient.

2.5 La segmentation automatique des lésions de sclérose en plaques

L'analyse des images IRM cérébrales des patients atteints de SEP est une tâche répétitive et chronophage effectuée en routine par les neuroradiologues. L'analyse des images est subjective et il a été observé qu'il y a une variation intra- et inter-radiologue quant à la segmentation des lésions de SEP [22], [23].

Un outil de segmentation automatique des lésions de SEP performant aide à réduire le temps d'analyse précieux des neuroradiologues, à donner des mesures reproductibles et de pouvoir fournir de nouvelles métriques jusqu'alors trop coûteuses à estimer.

De très nombreuses méthodes de segmentation automatique des lésions de SEP ont été proposées ces vingt dernières années. Ces méthodes se basent sur des modèles statistiques, des atlas, des modèles d'apprentissage machine et plus récemment en faisant usage de l'apprentissage profond [24]. L'apprentissage profond a été employé de façon extensive pour ce problème de segmentation et les autres méthodes, jugées moins performantes, sont peu à peu délaissées. Les dix dernières années, plus de cent méthodes utilisant les réseaux de neurones pour la segmentation des lésions de SEP ont été publiées avec des ensembles de données différents, différentes pondérations IRM, différentes architectures et méthodes d'apprentissage [6].

Plusieurs challenges ont été lancés par la communauté sur le sujet pour encourager la recherche dans le domaine [25],[26],[27]. Chacun a contribué à la recherche active dans le domaine en mettant à disposition, même *a posteriori* des images IRM dont les lésions de SEP sont segmentées par des radiologues.

Parmi les nombreuses méthodes d'apprentissage de segmentation des lésions de SEP, les contributions se focalisent souvent sur un gain en termes de performance, sur une métrique particulière.

Le travail réalisé durant ces trois ans vise à concevoir un outil performant de segmentation des lésions de SEP tout en restant le plus possible en adéquation avec les contraintes que l'on peut avoir dans des infrastructures médicales pour apporter au plus vite des solutions concrètes et fonctionnelles d'aide à la décision pour le diagnostic clinique.

3

Raffinement de l'architecture d'un réseau de neurones

Sommaire

3.1	État de l'art	55
3.2	Adaptation légère du U-net, le MPU-net	57
3.2.1	Données	58
3.2.2	Matériels et méthodes	60
3.2.3	Expérience	64
3.2.4	Réflexion sur les choix d'architecture	76
3.2.5	Conception du MPU-net	78
3.2.6	Discussion	80
3.3	Du MPU-net au MPU-next	81
3.3.1	Raffinement de l'architecture	82
3.3.2	Résultats et discussion	84
3.4	Adaptation du MPU-next pour la segmentation des nouvelles lésions de SEP	88
3.4.1	Contexte	88
3.4.2	Méthode proposée	90

Chapitre 3 – Raffinement de l'architecture d'un réseau de neurones

3.4.3	Résultats du challenge	93
3.5	Conclusion	95

LES réseaux de neurones artificiels sont employés avec succès dans de nombreux domaines. L'un des domaines où excellent ces modèles est en vision par ordinateur ou *computer vision*. Les RNCs se sont dressés comme référence pour des tâches de classification supervisée de photographie sur des milliers et dizaines de milliers de catégories. Les réseaux de neurones employés rivalisent sur leur architecture toujours plus complexe et régularisée en conséquence pour générer les meilleures performances.

Cependant, les RNCs entraînés pour ces tâches de classification haut niveau sont généralement très complexes et donc nécessitent une grande quantité de données et un coût en ressource de calcul très élevé.

Dans ce chapitre, guidés par l'hypothèse que la segmentation automatique des lésions de sclérose en plaques n'est pas aussi complexe que la classification d'image naturelle, nous proposons une architecture de RNC performante et légère qui ne nécessite ni grande infrastructure, ni des milliers d'exemples d'apprentissages. Le travail réalisé dans ce chapitre, met à disposition une architecture utilisable et entraînable plus simplement dans un environnement clinique et de recherche clinique vers une meilleure adoption des outils d'intelligence artificielle dans les hôpitaux.

Les travaux décrits dans ce chapitre ont fait l'objet des publications scientifiques suivantes : [28],[29],[30].

3.1 État de l'art

Le recours des réseaux de neurones artificiels et notamment les RNCs a le potentiel de donner les meilleures performances que les autres méthodes plus ou moins super-

visées. La première utilisation des réseaux de neurones artificiels pour la segmentation des lésions de SEP date de 1998 avec le travail de Goldberg-Zimring *et al.* [31]. Les auteurs se sont servis d'un réseau de neurones artificiels pour l'élimination des faux positifs sur leur méthode de segmentation. Puis, avec le développement de matériel de calcul toujours plus puissant et l'avènement conjoint des méthodes d'apprentissage profond, de plus en plus de méthodes ont été publiées en employant des réseaux de neurones convolutifs particulièrement adaptés pour les tâches en relation avec l'analyse automatique d'images.

Lors du challenge de segmentation des lésions de SEP MICCAI en 2016 [32], parmi les quatre meilleures méthodes [33],[34],[35],[36], seulement deux d'entre elles avaient recours à l'apprentissage profond, les autres faisant appel notamment à des méthodes de surface de niveau, un ensemble de règles définies et les forêts d'arbres décisionnels. McKinley *et al.* tirent avantage d'un réseau de neurones convolutif avec un encodeur et décodeur pour prédire le masque de segmentation des lésions alors que Valverde *et al.* se servent de réseaux de neurones pour prédire pixel à pixel si une hyperintensité FLAIR est une lésion.

Récemment, de nombreuses méthodes se sont développées en exploitant en majorité des approches d'apprentissage profond avec des RNCs pour la segmentation des lésions de SEP. Une architecture en particulier semble de plus en plus utilisée [6] avec succès: celle du U-net [37].

Le U-net a été développé à l'origine pour la segmentation des structures neuronales sur image de microscopie électronique et de cellules en microscopie photon à transmission. Le travail de Ronneberger a d'abord remporté le challenge ISBI 2015 pour le suivi des cellules, puis, au vu du succès et des performances du U-net, l'architecture a été de plus en plus employée et adaptée pour différentes tâches de segmentation notamment en imagerie médicale.

Le U-net a été adapté avec de plus en plus de succès et sert de modèle pour les dernières meilleures performances évaluées sur l'ensemble de données du challenge de segmentation des lésions de SEP ISBI de 2015¹ [38] :

- Le 2.5D Tiramisu Net [39],
- Le NN-Unet [40],
- Le Acu-net [41].

¹<https://smart-stats-tools.org/lesion-challenge>

C’est donc naturellement que nous avons choisi d’adapter le modèle d’architecture du U-net.

Cependant, le modèle de U-net et ses adaptations, y compris les meilleures, comptent des millions de paramètres d’apprentissage que le modèle doit apprendre pour effectuer sa tâche de segmentation. Plus un modèle doit apprendre de paramètres, plus il est susceptible de surapprendre et donc d’apprendre l’ensemble de données d’apprentissage [42] sans généraliser. Plus un modèle possède de paramètres, plus il est complexe, plus il nécessite d’exemples d’apprentissage pour généraliser. Seulement, en imagerie médicale et notamment pour la segmentation des lésions de SEP, il est difficile de réaliser un apprentissage avec un grand nombre d’images IRM annotées. En effet, les images médicales sont des données sensibles et la segmentation manuelle d’exemple d’apprentissage est longue, requiert un radiologue et est variable d’un expert à un autre (voir **Section 2.5**).

Nous avons donc travaillé à adapter une architecture de type U-net pour la segmentation des lésions de SEP [28]. Cette architecture se doit d’être performante et de pouvoir être entraînée sur peu de données et doit donc comporter un nombre limité de paramètres à optimiser, nous cherchons donc une architecture de type U-net légère.

3.2 Adaptation légère du U-net, le MPU-net

Comme illustré dans la **Figure 3.1**, le U-net original [37] possède une architecture composée de couches convolutives, de couches sous-échantillonnage par maximum, de couches de convolution transposée et des concaténations.

L’architecture originale du U-net, illustrée dans la **Figure 3.1**, est constitué d’un encodeur et d’un décodeur. L’encodeur rappelle une architecture classique de RNC avec des couches convolutives et de sous-échantillonnage diminuant la taille des cartes de caractéristiques tout en augmentant leurs nombres dans le but de compresser l’image en une information spatiale riche et compacte adaptée à la tâche de segmentation. Le décodeur partage une architecture assez proche de l’encodeur, mais est constitué de couches de convolutions transposées pour suréchantillonner la taille des cartes de caractéristiques, rendant la tâche de suréchantillonnage apprise et non fixée *a priori*. Le décodeur prend en entrée la sortie de l’encodeur, dense spatialement, et sort la segmentation du patch centré de l’image d’entrée. Le U-net a pour spécificité de faire des connexions entre l’encodeur et le décodeur à chacun de ses niveaux dans

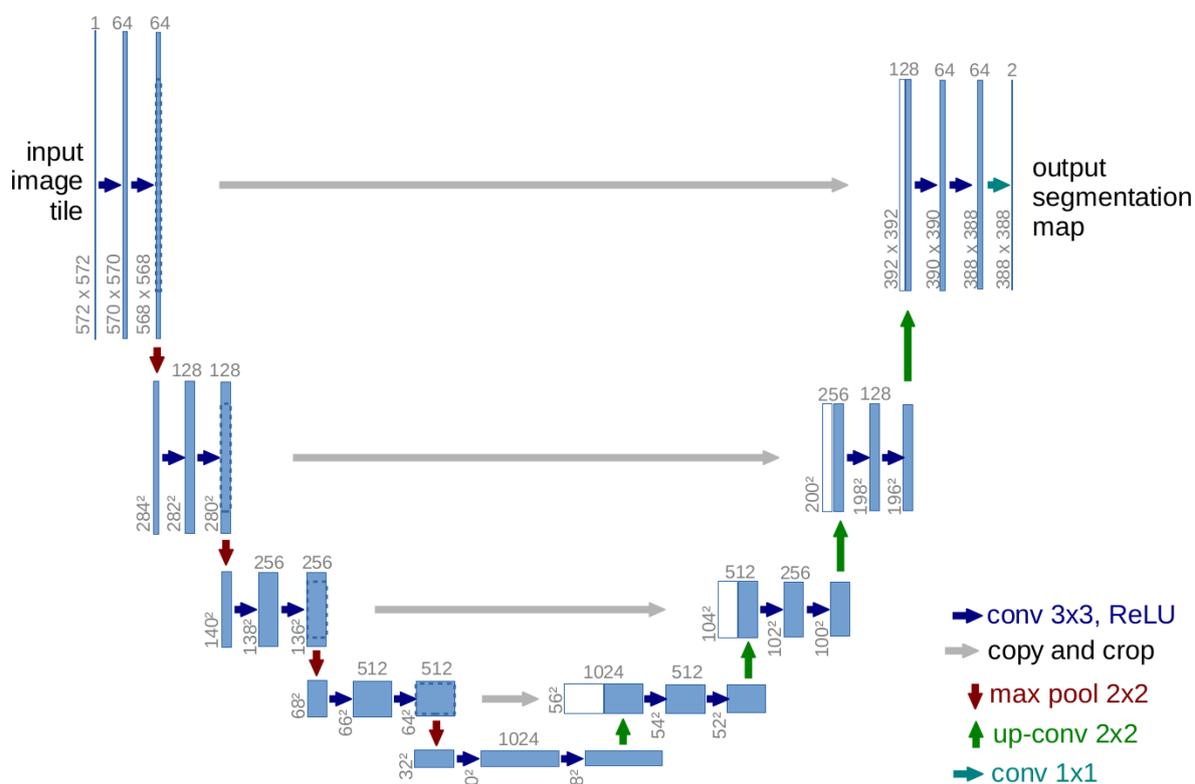


Figure 3.1 – Le U-net de Ronneberger issu de l'article original [37].

le but d'ajouter des détails spatiaux de meilleures résolutions dans le décodeur afin d'affiner sa segmentation et non pas reposer son analyse uniquement sur l'espace des caractéristiques latent généré par l'encodeur de faible dimension spatiale.

Le nom U-net provient tout simplement de la forme presque symétrique de l'architecture illustrée dans l'article de Ronneberger *et al.* [37].

Le travail développé ici consiste en l'adaptation de l'architecture du U-net pour obtenir un modèle à la fois simple et performant et partant d'un exemple minimal. Les différentes mutations appliquées à l'exemple minimal ont été effectuées pour évaluer l'impact de différents choix d'architecture.

3.2.1 Données

L'ensemble de données du challenge de segmentation des lésions de SEP de l'ISBI 2015 [26] est utilisé. Cet ensemble a été choisi principalement, pour s'évaluer sur l'ensemble de test².

²tableau des meilleurs score disponible à <https://smart-stats-tools.org/lesion-challenge>

L'ensemble d'apprentissage fourni par le challenge contient 21 examens IRM de 5 patients différents imagés chacun 4 à 5 fois avec un délai de 1 an environ entre chaque examen. L'ensemble de test quant à lui contient 61 examens IRM provenant de 14 patients différents eux aussi imagés plusieurs fois avec à chaque fois un an environ d'intervalle. Cependant, les segmentations sur l'ensemble de données de test sont gardées privées par les organisateurs. Nous avons donc, pour évaluer la pertinence des choix architecturaux, divisé l'ensemble d'apprentissage fourni par le challenge en deux sous-ensembles avec segmentations expertes:

- le sous-ensemble d'apprentissage avec 17 examens provenant de 4 patients différents,
- le sous-ensemble de validation avec 4 examens provenant du patient restant.

Les examens IRM consistent en des acquisitions T1, T2, DP et FLAIR. Pour l'ensemble d'apprentissage, les segmentations annotées par deux radiologues différents sur les images FLAIR sont fournies. Pour l'ensemble de test, les segmentations réalisées par ces deux mêmes radiologues ne sont pas disponibles et ne sont disponibles uniquement que pour l'évaluation en ligne.

Pour ce travail nous avons choisi de ne travailler qu'avec les images FLAIR de chaque examen pour :

- diminuer la complexité de la tâche,
- se rapprocher de la pratique clinique où les radiologues analysent principalement l'imagerie FLAIR pour l'évaluation des lésions voir **Section 2.3.2**,
- conserver uniquement la pondération la plus adaptée pour la segmentation automatique de lésions de SEP comme décrit par Feng *et al.* [43];

Prétraitement

Toutes les images ont été prétraitées pour le challenge [26]. Le prétraitement appliqué sur l'image FLAIR consiste en une correction de biais de champ avec l'algorithme du N4 [44], un alignement rigide sur l'atlas du MNI [45] suivi d'un rééchantillonnage à 1 mm^3 isométrique puis le masque du cerveau extrait sur le T1 [46], [47] est appliqué à l'image pour ne conserver que les structures de l'encéphale et enfin une dernière correction de biais de champ N4 est appliquée.

Pour utiliser les images dans cette étude, nous avons ajouté quelques étapes de prétraitement supplémentaires. Les images ont été rognées autour du cerveau afin d'éviter les problèmes de mémoire et de calculs inutiles. Pour rendre les intensités comparables, les histogrammes ont été mis en correspondance par la méthode de Nyul *et al.* [48] et les intensités ont ensuite été normalisées en retranchant la moyenne et en

divisant par l'écart type des intensités des voxels du cerveau.

3.2.2 Matériels et méthodes

Entraînement

En nous basant sur le travail de Snehashis *et al.* [49], les modèles sont entraînés sur des petits patches de taille $32 \times 32 \times 32$ voxels. Cette taille de patch permet de capturer les lésions ainsi que la texture environnante comme il est illustré dans la **Figure 3.2**.

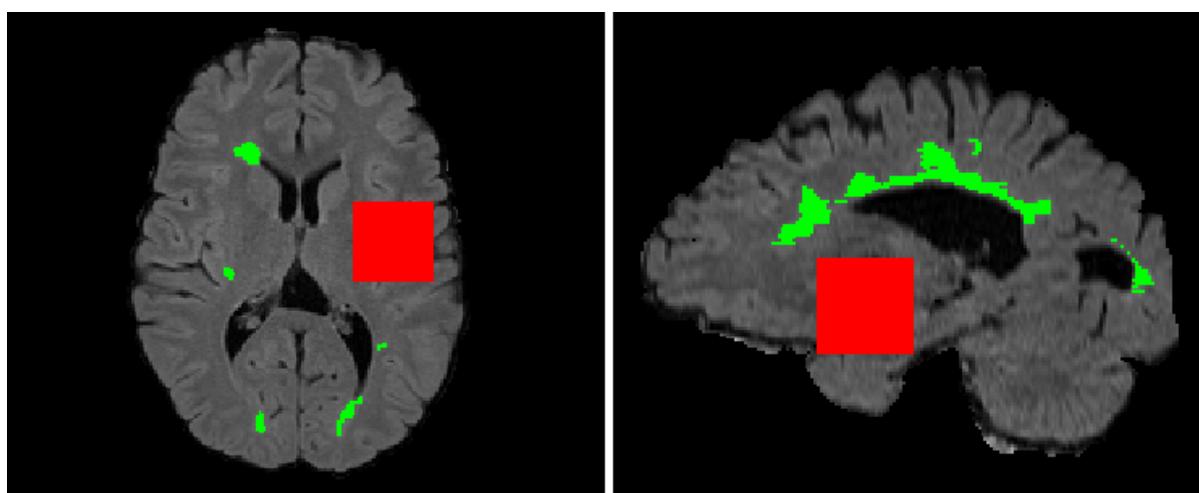


Figure 3.2 – La taille des patches en rouge et la taille des lésions en vert en vue axial à gauche et sagittale à droite.

Tous les entraînements sont réalisés sur des lots contenant 5 patches de chaque patient couvrant différentes zones du cerveau aléatoirement. Nous avons défini ici une époque comme composée de 100 lots qui couvrent de façon uniforme le cerveau. Les entraînements sont stoppés après 20 époques sans amélioration des performances sur l'ensemble de validation.

Durant les phases de test, la prédiction finale est la moyenne spatiale des masques prédits espacés de $8 \times 8 \times 8$ voxels pour avoir un recouvrement partiel des patches et d'éviter les effets de bords dus à l'emploi des patches.

Durant les différents apprentissages, les réseaux de neurones apprennent à prédire chacune des segmentations fournies par l'ensemble de données pour évaluer la capacité du réseau à mimer deux segmentations proches pour la même tâche.

Cadre de l'expérience

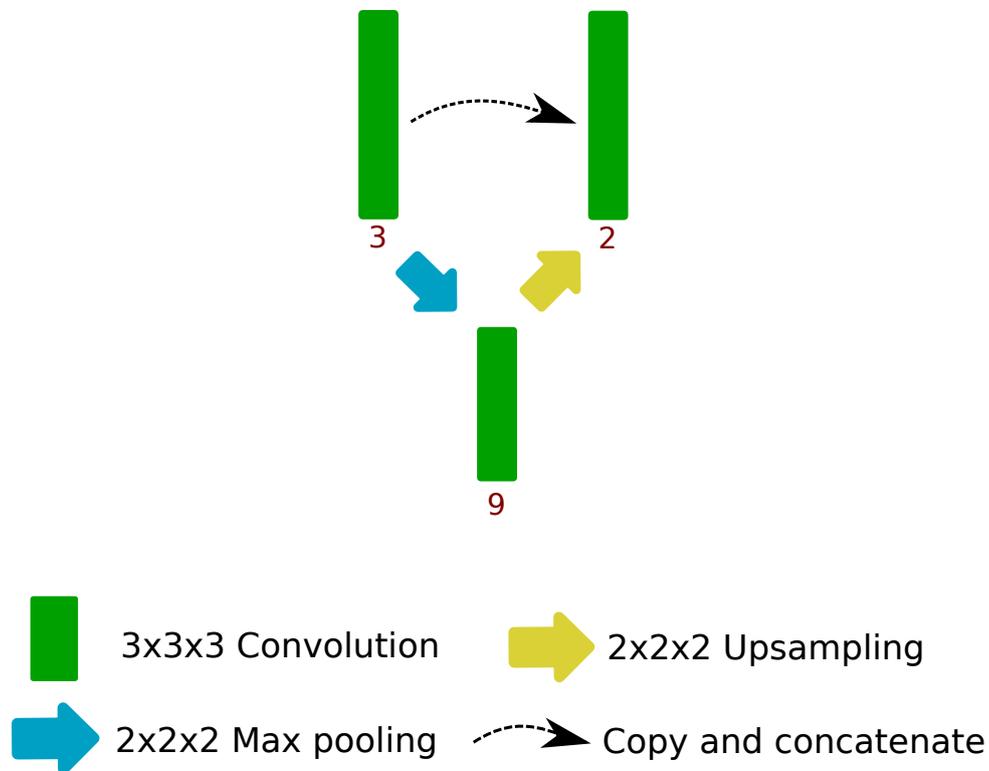


Figure 3.3 – Le modèle minimal d'architecture employé pour l'adaptation simple du U-net.

Comme il est expliqué précédemment dans la **Section 3.2**, l'expérience vise à changer l'architecture simple de référence illustrée dans la **Figure 3.3** pour justifier les choix d'architecture afin de concevoir un réseau de neurones léger pour la segmentation des lésions de SEP. Les différents choix architecturaux évalués concernent :

- le sous-échantillonnage et suréchantillonnage,
- la présence de connexions par concaténation,
- le nombre de niveaux du U-net,
- le nombre de cartes de caractéristiques,
- le nombre de couches convolutives,

et sont illustrés dans la **Figure 3.4**.

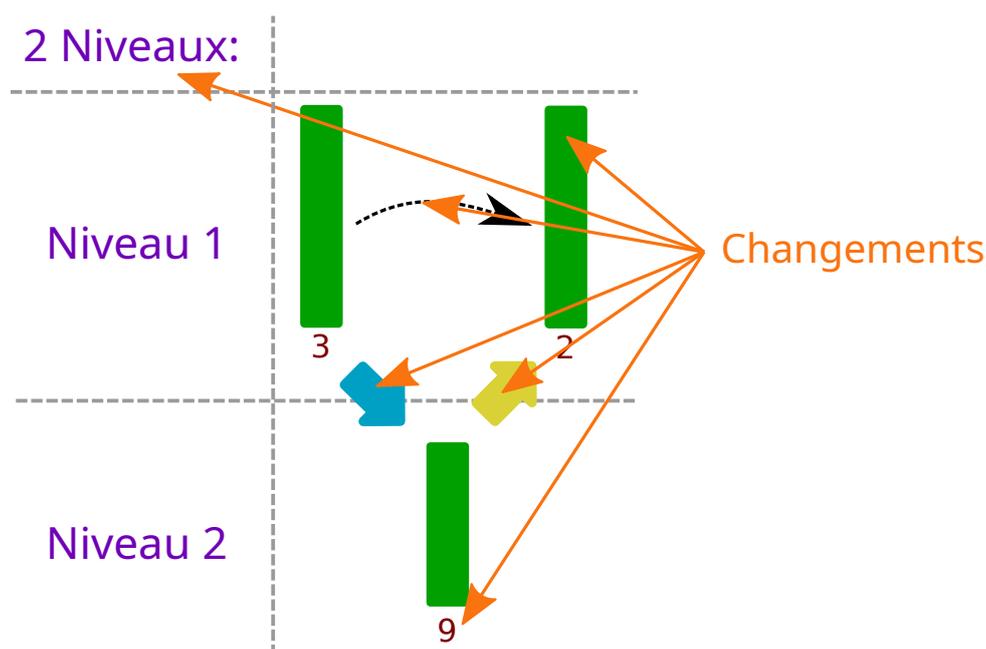


Figure 3.4 – Les différents changements appliqués à l'architecture simplifiée du U-net utilisée.

Un certain nombre de règles et de choix a été appliqué pour simplifier les apprentissages et limiter le nombre d'expériences à mener. Les couches de *Dropout* [18] ou de *Batch normalization* [17] n'ont pas été mises en place pour pouvoir observer les éventuels phénomènes de surapprentissage qui peuvent apparaître. Toutes les couches de convolutions sont suivies d'une activation *leaky rectified linear activation function* (*leaky ReLU*) et les paramètres sont initialisés aléatoirement de façon uniforme selon la méthode de Glorot et Bengio [21]. Nous avons aussi fixé des règles arbitraires pour limiter la complexité des architectures testées :

- le nombre de cartes de caractéristiques des couches de convolutions d'un même niveau dans l'encodeur ou le décodeur doit être même ;
- le nombre de cartes de caractéristiques des couches convolutives doit être croissant avec le niveau ;
- le nombre de cartes de caractéristiques doit augmenter rapidement avec le niveau jusqu'à 80, pour ensuite augmenter légèrement.

Pour chaque variation d'architecture testée, nous avons fait le choix d'entraîner plusieurs versions différentes. Chaque version a un nombre de niveaux différent comme illustré dans l'exemple de la **Figure 3.5**. Un niveau désigne le nombre d'étages dans notre architecture proche du U-net comme illustré dans la **Figure 3.4**. Quatre versions d'architecture ont été évaluées à chaque fois avec respectivement 2, 3, 4 et 5 niveaux. Le nombre de cartes de caractéristiques pour chaque niveau est donné dans le **Tableau 3.1**.

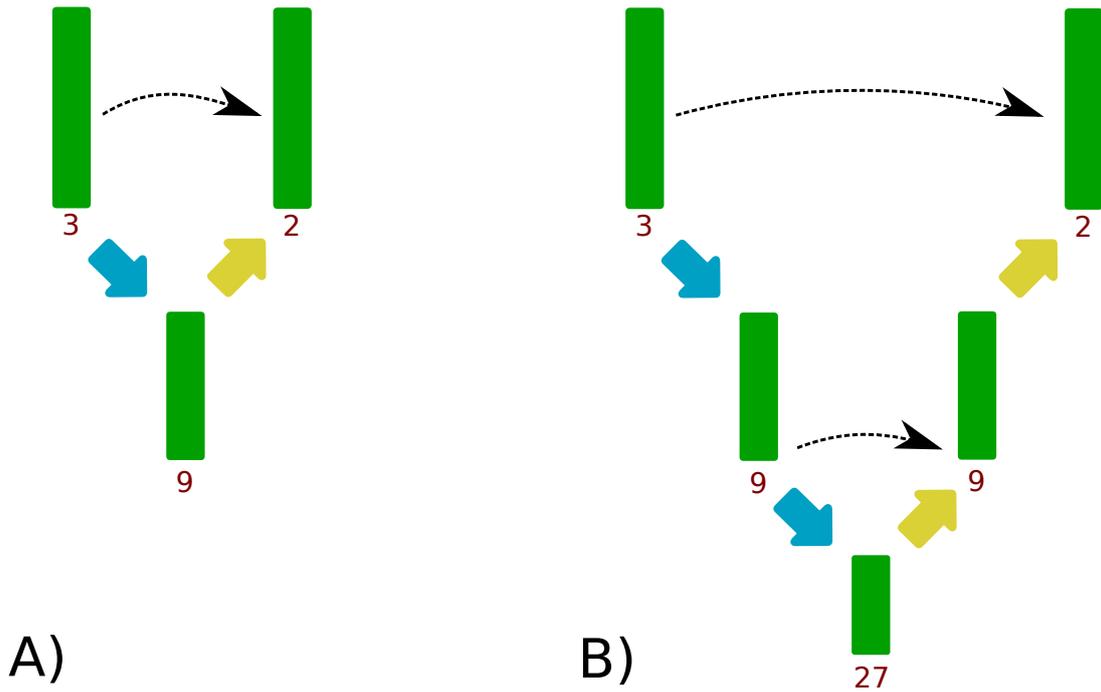


Figure 3.5 – Variation avec deux (a) et trois (b) niveaux de l'architecture de référence en **Figure 3.3**. Le nombre de cartes de caractéristiques de sortie est noté en rouge en dessous des couches convolutives.

Niveau 1	Niveau 2	Niveau 3	Niveau 4	Niveau 5
3	9	27	81	99

Table 3.1 – Nombre de cartes de caractéristiques pour chaque niveau.

Tous les modèles sont entraînés avec le coût de Dice par lot défini par l'**Equation 3.1**. Le coût est calculé sur l'entièreté du lot plutôt que comme la moyenne du coût de Dice pour limiter la contribution au coût des patches sans lésion.

$$\text{Coût Dice}(P, \hat{P}) = -2 \frac{\sum_{b=1}^B \sum_{h=1}^H \sum_{w=1}^W \sum_{d=1}^D \sum_{s=1}^S P_{c,b,h,w,d,s} \hat{P}_{c,b,h,w,d,s}}{\sum_{b=1}^B \sum_{h=1}^H \sum_{w=1}^W \sum_{d=1}^D \sum_{s=1}^S P_{c,b,h,w,d,s} + \hat{P}_{c,b,h,w,d,s}}, \quad (3.1)$$

où B est la taille du lot, H la hauteur, W la largeur D la profondeur de l'image, S le nombre de segmentations expert, $P \in \{0; 1\}$ la segmentation binaire des radiologues et $\hat{P} \in [0; 1]$ la prédiction du réseau.

Le coût de Dice est l'adaptation du score de Dice **Equation 3.2** pour l'entraînement des réseaux de neurones. C'est généralement cette fonction coût ou une variante qui

est utilisée pour l'entraînement des réseaux de neurones pour cette tâche. En effet, l'évaluation des performances des segmentations, notamment pour les lésions de SEP, est généralement calculée avec le score de Dice [43], [50]–[52] qui est une mesure de similarité entre 0 et 1. Cette mesure peut aussi être écrite comme la moyenne harmonique de la mesure de précision et de sensibilité (**Equation 3.2**). La sensibilité mesure la capacité à détecter un maximum de voxels appartenant à la lésion et la précision la capacité à détecter correctement ces voxels (**Equation 3.3**). Donc le score de Dice mesure la capacité du modèle à détecter un maximum de voxels appartenant aux lésions tout en contraignant à limiter les faux positifs.

$$Dice = \frac{2TP}{2TP + FP + FN}, \quad (3.2)$$

$$Dice = 2 \cdot \frac{Precision \times Sensibilite}{Precision + Sensibilite},$$

avec

$$Sensibilite = \frac{TP}{P}, \quad (3.3)$$

$$Precision = \frac{TP}{TP + FP},$$

où TP est le nombre de vrais positifs, FP le nombre de faux positifs et P le nombre de positifs.

3.2.3 Expérience

Pour chaque expérience, chaque modèle différent a été entraîné et évalué 5 fois. Les graphiques dans cette section montrent la performance moyenne de ces entraînements avec l'intervalle de répartition complet. La taille de l'intervalle permet d'évaluer la stabilité et la robustesse des différentes expériences.

Toutes les architectures testées sont issues des changements de notre modèle simplifié (voir **Figure 3.3**), les performances de chaque nouvelle architecture sont donc comparées avec les performances de cette architecture de référence.

Le sous-échantillonnage et suréchantillonnage

Pour évaluer les meilleurs choix de sous- et sur-échantillonnage, nous avons réalisé les trois expériences suivantes :

- sous-échantillonnage avec couche de *max-pooling* et suréchantillonnage interpolé aux plus proches voisins, couche simple d'*upsampling*, la référence en **Figure 3.3**,
- sous-échantillonnage avec des couches convolutives avec un pas de convolution de 2 et un sur-échantillonnage avec une convolution transposée de même pas en **Figure 3.6**,
- sans échantillonnage sur la **Figure 3.7**.

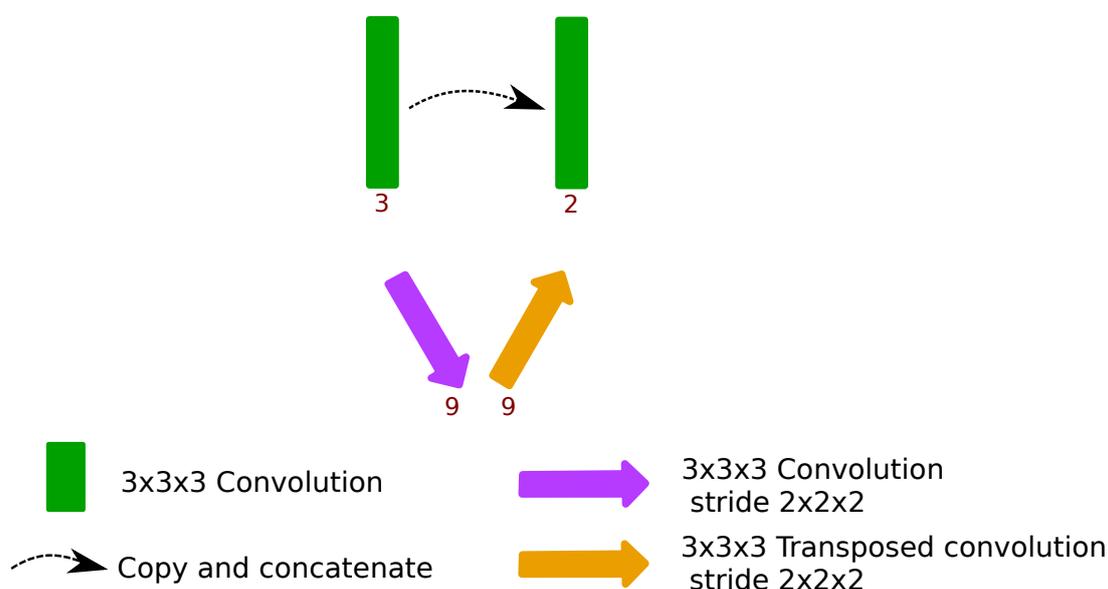


Figure 3.6 – La version avec deux niveaux de l'architecture avec des convolutions avec des pas de 2.

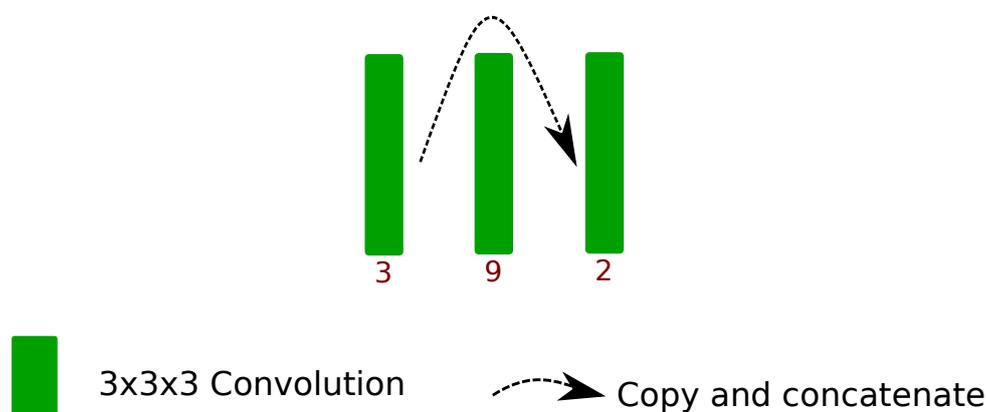


Figure 3.7 – La version avec deux niveaux de l'architecture sans échantillonnage.

Sur la **Figure 3.8**, nous observons que le couple *max-pooling* et *upsampling* (la référence) donne ici les meilleures performances. Pour chaque version, la référence donne un meilleur score de Dice d'environ 0.1 comparée à l'architecture sans échantillonnage. L'échantillonnage avec des convolutions de pas de 2 possède un intervalle de répartition plus petit et possède des performances très légèrement inférieures à la référence pour les deux dernières versions.

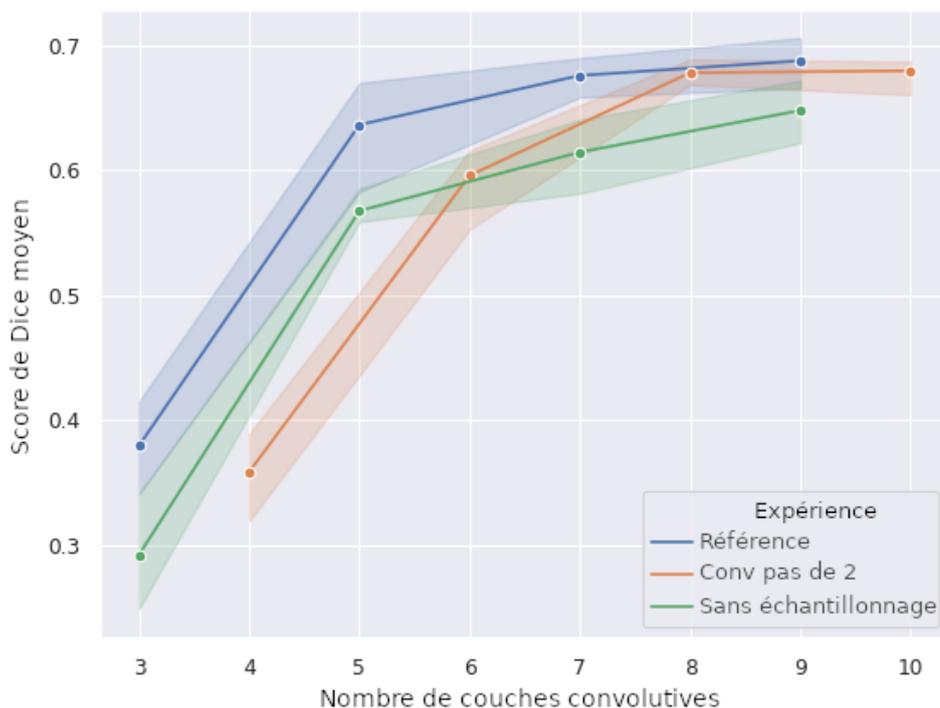


Figure 3.8 – Évaluation des différents échantillonnages.

Les connexions par concaténation

Dans le U-net, les connexions par concaténation sont effectuées entre les convolutions du même niveau dans l'encodeur et le décodeur. Pour étudier l'importance de ces connexions, nous avons étudié les choix suivants illustrés dans la **Figure 3.9** :

- sans connexion par concaténation,
- sans connexion par concaténation ni échantillonnage (linéaire).

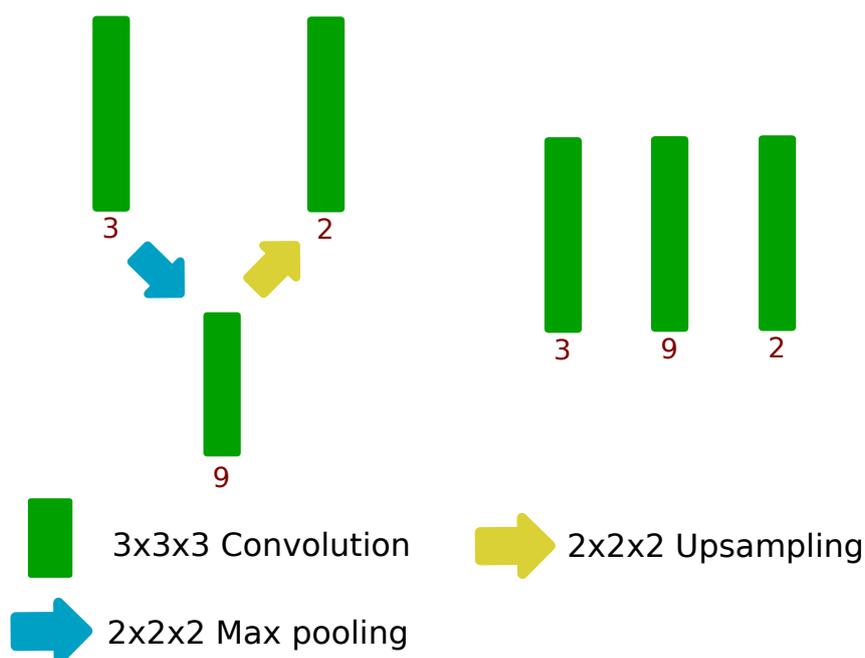


Figure 3.9 – Les versions à deux niveaux de l'architecture sans connexions à gauche et sans connexions ni échantillonnage à droite.

Nous observons dans la **Figure 3.10** que les deux premières versions l'architecture sans connexions donnent des performances proches de l'architecture de référence. Pour les deux dernières versions sans connexion, les performances s'effondrent à partir de 7 couches convolutives jusqu'à un score de Dice moyen de 0.4 dans un intervalle compris entre 0 et 0.52 pour la dernière version contre un score de Dice moyen de 0.79 dans un intervalle entre 0.67 et 0.71 pour la référence. Les performances de l'architecture linéaire (sans connexion, ni échantillonnage) sont inférieures aux performances de l'architecture de référence de 0.1 de score de Dice puis de 0.03 pour la dernière version. L'architecture linéaire tend donc à se rapprocher des performances de référence avec une complexité croissante sans montrer de signe d'effondrement des performances observées pour la version sans connexion avec couches d'échantillonnage. Si nous comparons les performances de l'architecture linéaire de la **Figure 3.10** et l'architecture sans échantillonnage, mais avec connexions de la **Figure 3.7**, nous observons des performances similaires.

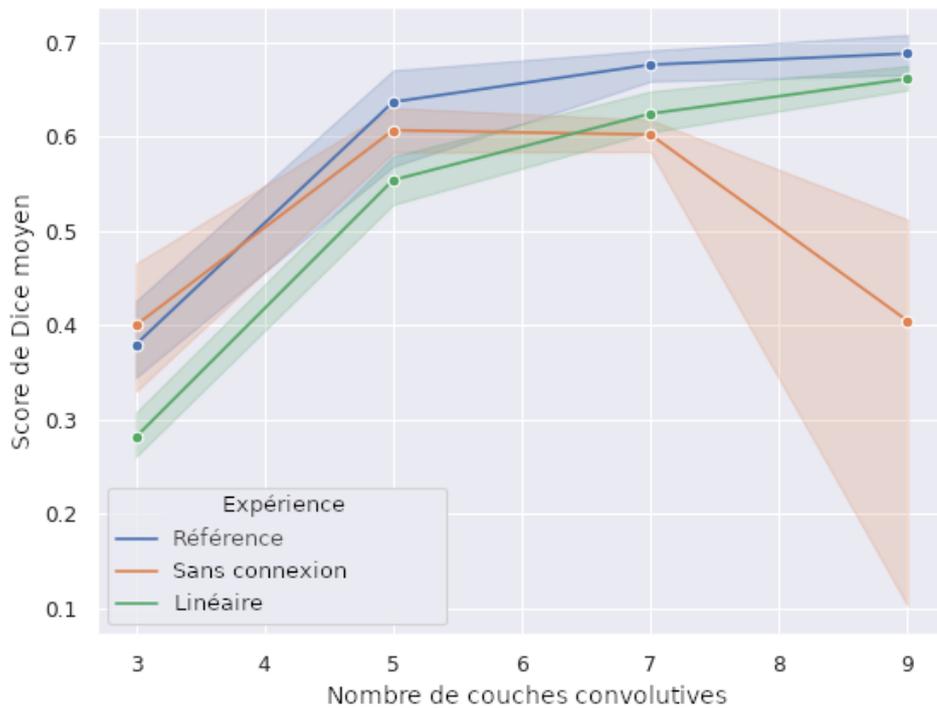


Figure 3.10 – Évaluation de l'apport des connexions par concaténation.

Augmentation de la complexité de l'architecture

La capacité d'analyse et la complexité d'une architecture donnée dépendent du nombre de paramètres d'apprentissage qui sont le nombre de poids totaux appris pour la tâche, mais aussi le nombre de couches convolutives qui influe sur le pouvoir de hiérarchisation et d'abstraction de l'analyse du RNC. Dans cette partie, l'impact de l'augmentation de la complexité est étudié en ajoutant plus de couches convolutives et en ajoutant plus de cartes de caractéristiques en sortie (et donc en augmentant directement le nombre de paramètres d'apprentissage).

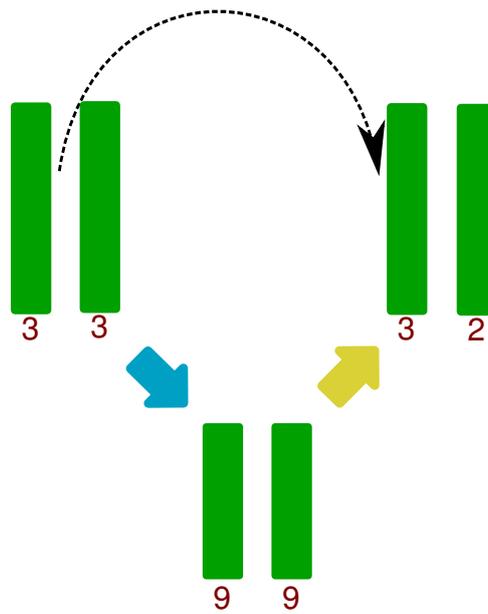
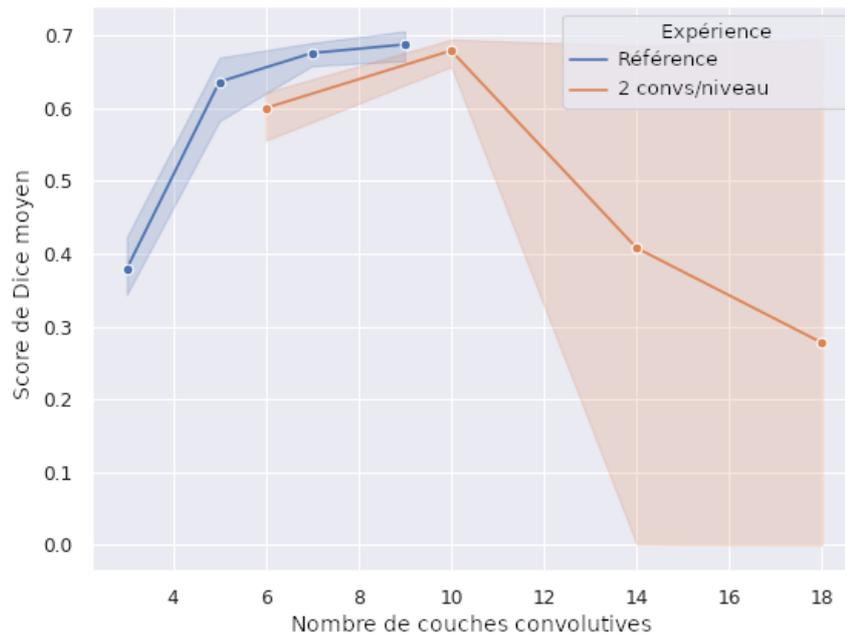


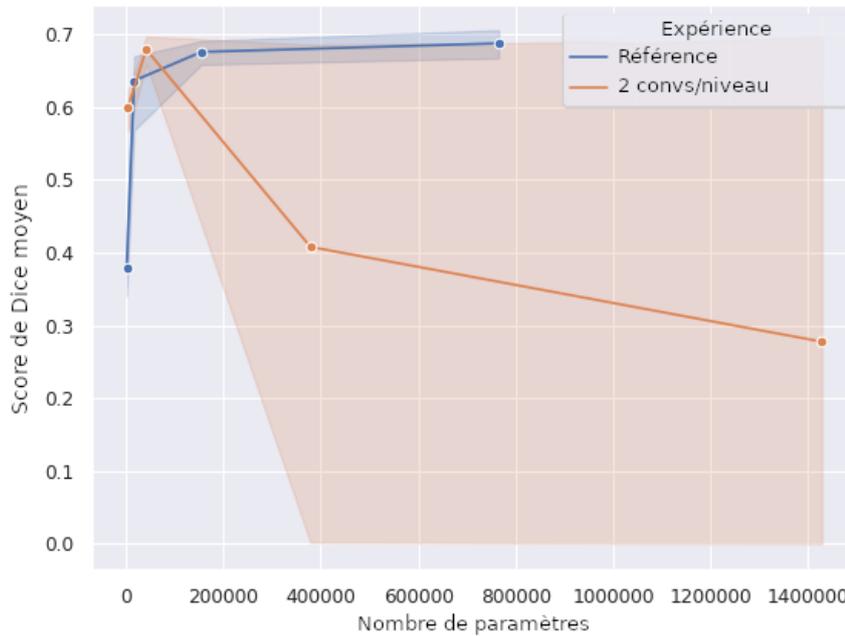
Figure 3.11 – La version à deux niveaux de l'architecture avec deux couches convolutives par bloc.

Plus de couches convolutives

À l'instar du U-net, nous avons étendu l'architecture avec deux couches convolutives par bloc comme illustré dans la **Figure 3.11**.



(a) Évaluation de l'impact de l'ajout d'une couche convolutive par bloc.



(b) Évaluation de l'impact de l'ajout d'une couche convolutive par bloc en fonction du nombre de paramètres.

Dans les **Figures 3.12a** et **3.12b**, l'ajout d'une convolution par bloc améliore les performances pour les deux premières versions comparées à l'architecture de référence.

En effet, la première version avec 6 couches convolutives et 4 403 paramètres donne un score de Dice moyen de 0.6 comparé à 0.38 pour la référence avec 3 couches convolutives et 1 472 paramètres. La deuxième version de l'architecture à deux convolutions par bloc est aussi meilleure que la deuxième version de référence avec un gain de 0.04 de score de Dice pour 5 couches convolutives additionnelles et 25 000 paramètres additionnels. Cependant, à partir de 14 couches convolutives et 379 000 paramètres, les performances moyennes s'écroulent et l'intervalle de performance s'élargit drastiquement avec deux couches convolutives par bloc. Ces résultats s'expliquent par le fait que, certains apprentissages parmi les 5 réalisés, ont appris un modèle qui n'a pas généralisé, avec un score de Dice proche de 0, alors que d'autres modèles appris avec la même architecture sont performants avec un score de Dice proche de 0.68.

Les résultats des **Figures 3.12a** et **3.12b** indiquent que l'ajout de couches convolutives apporte un gain de performance avec peu de paramètres d'apprentissage, cependant, trop de couches convolutives peuvent provoquer des instabilités sur les apprentissages allant jusqu'à la perte totale de la capacité de généralisation.

Plus de cartes de caractéristiques

Les couches de convolution du U-net original sont plus complexes (avec 64, 128, 256, 512, 1024 cartes de caractéristiques de sortie), comparé à l'architecture de référence de cette étude (avec 3, 9, 27, 81, 99 cartes de caractéristiques). Nous avons alors évalué l'impact de l'augmentation du nombre de cartes de caractéristiques seule sur l'architecture de référence. L'ajout de cartes de caractéristiques augmente le nombre de paramètres d'apprentissage sans changer la profondeur du réseau.

Deux variantes différentes avec plus de cartes de caractéristiques ont été évaluées appelées *Ref+* et *Ref++*, avec les nombres de cartes de caractéristiques pour chaque niveau de chacune des variantes détaillées dans le **Tableau 3.2**.

Nom	Niveau 1	Niveau 2	Niveau 3	Niveau 4	Niveau 5
<i>Ref+</i>	6	18	36	96	108
<i>Ref++</i>	9	24	45	108	112

Table 3.2 – Nombre de cartes de caractéristiques pour chaque niveau pour les variantes *Ref+* et *Ref++*.

Dans la **Figure 3.13**, les variantes *Ref+* et *Ref++* donnent de meilleurs résultats en moyenne que l'architecture de référence jusqu'à la dernière version où toutes les architectures convergent toutes vers un score de Dice moyen de 0.688. Les intervalles de performances des variantes *Ref+* et *Ref++* sont aussi moins étendus indiquant des apprentissages plus stables avec ces variantes. La variante avec les meilleurs scores

de Dice moyens est *Ref+* et le meilleur score de Dice moyen à 0.689 est atteint avec 4 niveaux à 7 couches convolutives .

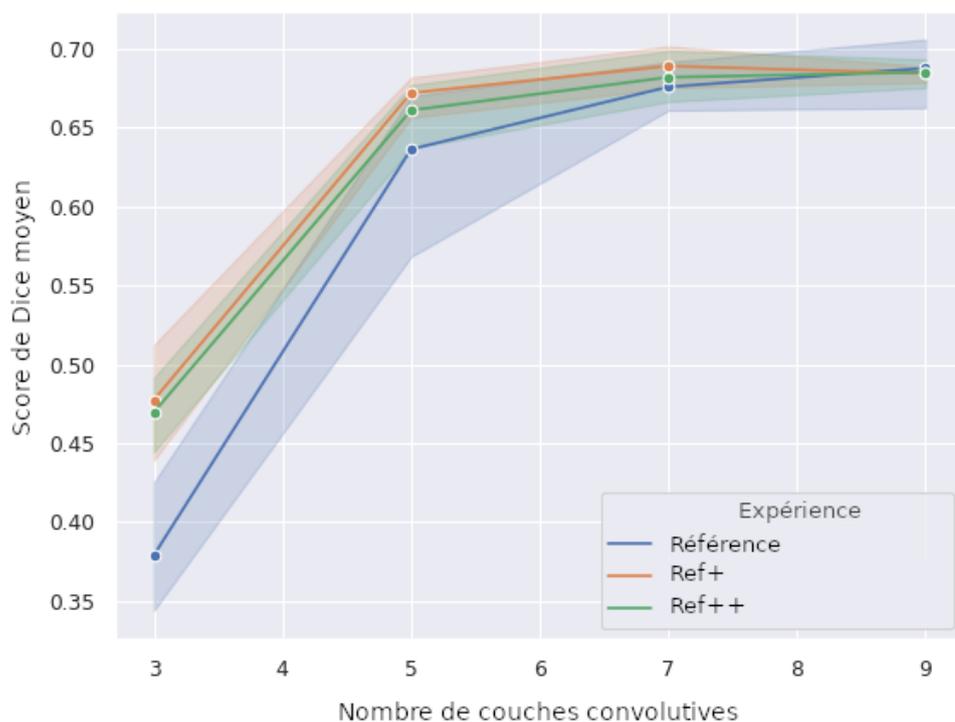
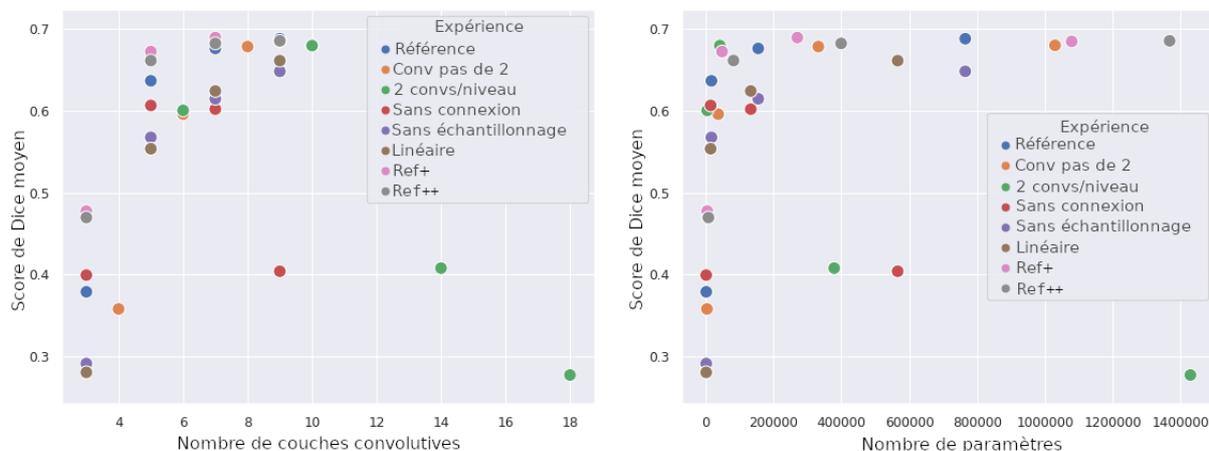


Figure 3.13 – Évaluation de l'impact de l'augmentation du nombre de cartes de caractéristiques.

Complexité et performances

Dans la **Figure 3.14a**, les performances augmentent avec le nombre de couches convolutives jusqu'à environ 7 couches convolutives, puis nous observons un plateau. Les architectures testées convergent toutes vers un score de Dice moyen proche de 0.65 excepté l'architecture sans connexion et l'architecture avec deux couches convolutives par bloc. La convergence de score de Dice apparaît à partir de 5 couches convolutives. Dans la **Figure 3.14b**, les résultats sont similaires avec le nombre de paramètres d'apprentissage. La convergence apparaît à partir de 100 000 paramètres. Le meilleur score de Dice moyen de 0.689 est atteint pour environ 270 000 paramètres avec 7 couches convolutives de l'architecture *Ref+* . Il apparaît sur la **Figure 3.14b** que l'architecture avec deux couches convolutives par bloc donne de meilleures performances que d'autres versions avec le même nombre de paramètres avant de s'effondrer avec des versions plus complexes en termes de nombre de paramètres et de couches

convolutives.



(a) Score de Dice moyen en fonction du nombre de couches convolutives.

(b) Score de Dice moyen en fonction du nombre de paramètres.

Figure 3.14 – Le score de Dice moyen de toutes les expériences.

Sensibilité et précision

Les modèles testés sont entraînés avec le coût de Dice et doivent donc apprendre un compromis entre sensibilité et précision (voir **Section 3.2.2**).

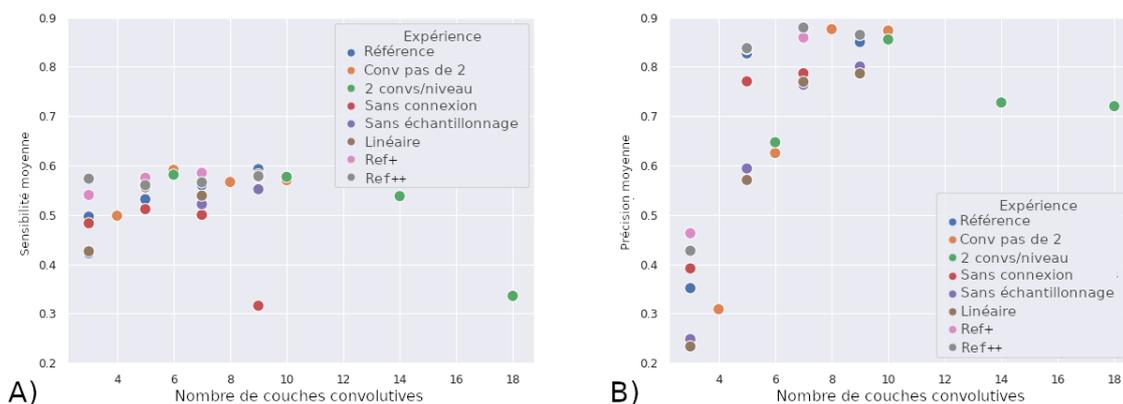


Figure 3.15 – Sensibilité (a) et précision (b) en fonction du nombre de couches convolutives.

Les expériences convergent vers une sensibilité d'environ 60% avec seulement 3 couches convolutives alors que la précision atteint 80% avec au moins 5 couches

convolutives (voir **Figure 3.15**). Nos expériences nous permettent de mettre en évidence les 2 points suivants :

- moins de couches convolutives sont nécessaires pour atteindre la meilleure sensibilité que la meilleure précision,
- la précision est privilégiée à la sensibilité en général pour les architectures performantes.

Visualisation de la segmentation

Dans cette partie, un exemple des segmentations réalisées par différents modèles sur une coupe sont présentés.

Sur les images FLAIR, les lésions apparaissent en hyperintensité. Dans l'exemple présenté dans les **Figures 3.16,3.17**, nous ne retrouvons que des lésions périventriculaires qui sont les plus représentées dans la SEP.

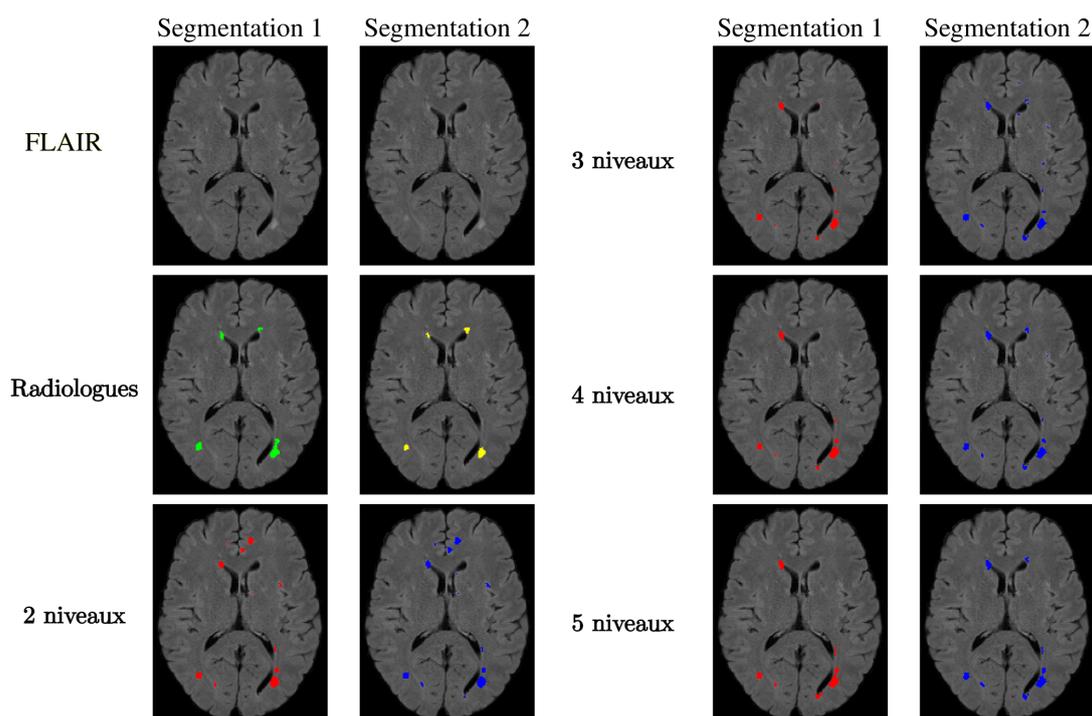


Figure 3.16 – Segmentation moyenne des versions de références.

Dans la **Figure 3.16**, toutes les segmentations des versions de référence sont proches exceptée la moins profonde, la version à deux niveaux qui a tendance à segmenter chaque hyperintensité sans distinction avec les structures anatomiques saines. Dans la **Figure 3.17**, les meilleures versions de chaque architecture testée convergent vers les mêmes prédictions dans cet exemple.

Pour chaque architecture testée les deux cartes de segmentation prédites (une pour

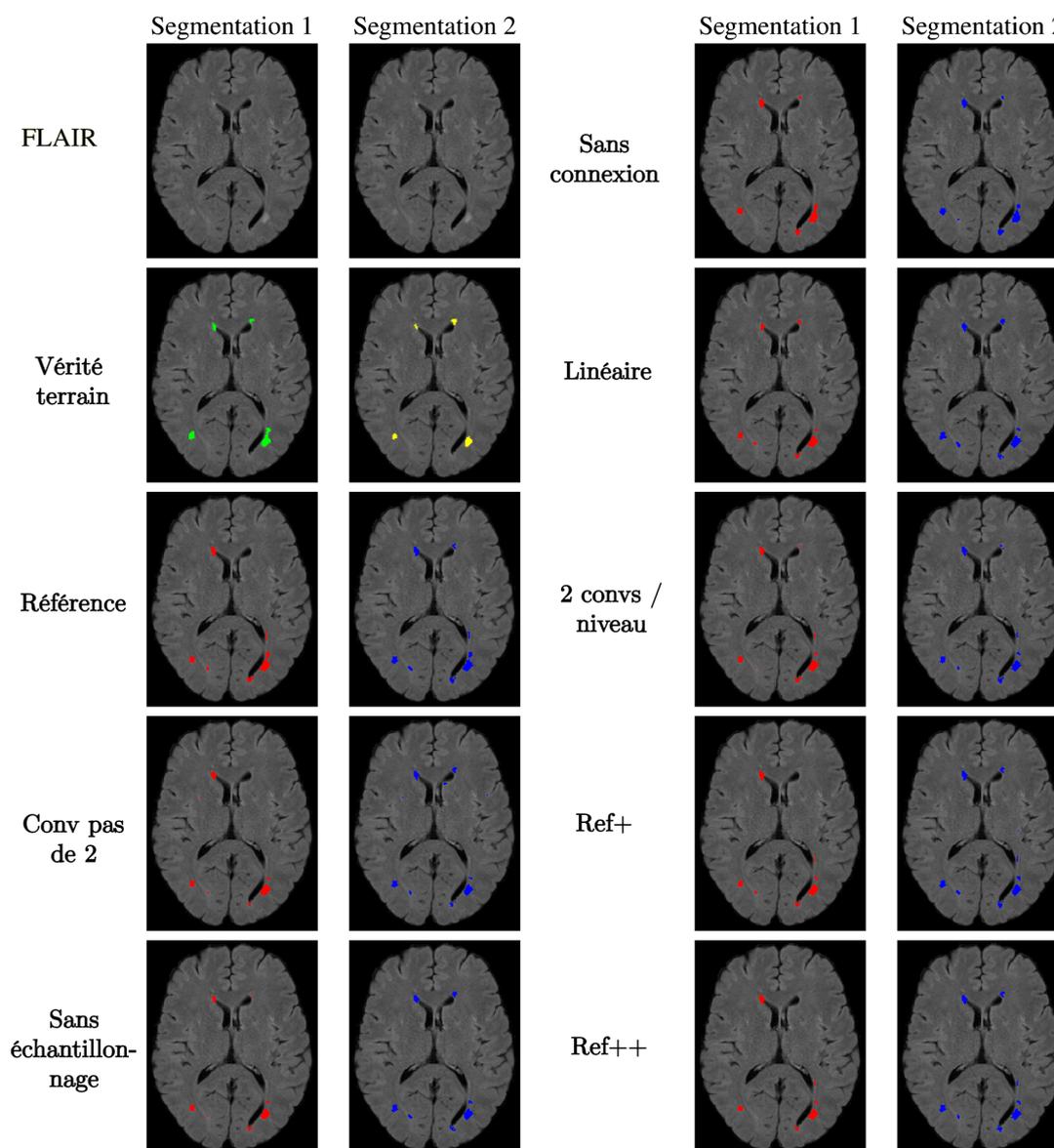


Figure 3.17 – Segmentation moyenne des meilleures versions de chaque expérience.

chaque segmentation réalisée par un radiologue différent), sont similaires bien qu'en réalité les segmentations sont légèrement différentes.

L'ensemble de prédictions observées dans les **Figures 3.16,3.17** surestiment légèrement proche de l'extrémité postérieure droite du ventricule. Dans l'ensemble de données et pour la SEP en général, les lésions périventriculaires sont plus représentées que les autres types de lésions. Il est possible que les modèles tendent à être plus sensibles aux hyperintensités proches des ventricules.

3.2.4 Réflexion sur les choix d'architecture

Échantillonnage

Nos résultats indiquent que le sous- et sur-échantillonnage et donc que la réduction d'informations locales améliore les performances. De façon empirique, nous observons que l'apprentissage de l'échantillonnage avec des convolutions de pas deux ne donne pas d'amélioration dans nos expériences. Le couple *max-pooling* et le suréchantillonnage interpolé aux plus proches voisins sont en pratique meilleurs. Ces transformations sont prédéfinies et n'ont pas besoin d'être apprises et simplifient donc le processus d'apprentissage. Le *max-pooling* est largement mise à profit dans les RNCs en général [53], notamment car il crée une invariance à la position. Le *max-pooling* diminue la taille de cartes de caractéristiques d'entrée, simplifie l'information et aide à accélérer la convergence en ne sélectionnant que les caractéristiques les plus fortes, ce qui apporte une meilleure généralisation[54]. Le suréchantillonnage interpolé aux plus proches voisins permet aussi éviter les problèmes d'artefact en échiquier causé par les convolutions transposées [55].

Connexion par concaténation

Un des éléments clefs du U-net est l'utilisation des connexions par concaténation entre chaque bloc de l'encodeur et du décodeur de chaque niveau. Ces connexions sont connues pour aider à la propagation du gradient dans les réseaux de neurones et ainsi permettre un meilleur apprentissage quand les architectures deviennent très profondes et permettent une convergence plus rapide [56],[57].

Sans sur- et sous-échantillonnage, les connexions par concaténation ne sont pas utiles, pire, elles perturbent même la segmentation. Cela s'explique par le fait que le *max-pooling* est aussi connu pour causer du surapprentissage [53],[58]. Dans le cas spécifique du U-net, les connexions par concaténation pourraient limiter ce problème de surapprentissage, et pourraient ainsi faire bénéficier des avantages du *max-pooling* en évitant ses limites.

Complexité et performance

Nous avons observé qu'un nombre minimum de couches convolutives et de paramètres d'apprentissage sont nécessaires pour converger vers un score de Dice autour de 0.65 pour la segmentation des lésions de SEP. Nous avons obtenu les meilleures performances avec environ 270 000 paramètres, ce qui est dans les nombres de paramètres

les plus bas (voir **Section Tableau 3.3**). Cette limite est fortement plus basse que ce que l'on peut retrouver dans des architectures très profondes et complexes employées, par exemple, dans la classification d'image naturelle. La segmentation des lésions de SEP sur des images FLAIR, en comparaison, est une tâche plus simple et donc requiert une architecture moins complexe. Les images sont assez similaires entre elles et avec les différentes phases de prétraitement, le problème d'échelle, de positionnement, d'illumination commun sur les images naturelles est limité voire inexistant.

Il est possible de réaliser, avec le même nombre de paramètres, des tâches plus compliquées dans les réseaux de neurones en augmentant le nombre de couches convolutives [14][59]. Cela explique le recours extensif aux architectures très profondes dans l'état de l'art. Nos observations empiriques sont en accord avec ces travaux, en effet, le doublement du nombre de couches convolutives permettent de bonnes performances avec seulement 40 000 paramètres comme il est illustré dans la **Figure 3.12b**. Pour certaines expériences, nous avons aussi identifié une baisse de performance quand le nombre de paramètres était trop haut, car le nombre de paramètres en apprentissage aussi est assujettit à la malédiction de la dimension [59]. Avec le même nombre de paramètres, l'ajout d'un grand nombre de couches peut aussi limiter les performances en raison du phénomène de disparition du gradient.

Dans la majorité des expériences, nous avons obtenu une convergence de performance à partir d'une certaine complexité de modèle avec un plateau. Ce phénomène est intéressant puisqu'il indique que les meilleures qualités de segmentation obtenues ici sont atteintes à partir d'une certaine complexité et qu'augmenter celle-ci est inutile voir contre-productif dans nos conditions expérimentales. L'ajout de certaines régularisations ou de lien peut cependant aider à bénéficier d'un apport de performance avec des architectures plus complexes [19] et cela reste une limite des expériences menées.

Roy <i>et al.</i> [49]	Notre meilleure architecture Figure 3.14	Valverde <i>et al.</i> [50]	Zhang <i>et al.</i> [60]	Aslani <i>et al.</i> [61]
~ 240k	~ 270k	~ 470k	~ 29M	< 75M

Table 3.3 – Nombre de paramètres d'apprentissage de certaines architectures utilisées.

Sensibilité et précision

Ici, les modèles les moins profonds sont sensibles, mais pas précis. La précision est améliorée avec des modèles plus complexes en maintenant la même sensibilité. La segmentation des lésions de SEP nécessite peu de couches convolutives pour atteindre un niveau acceptable de sensibilité, mais la précision vient avec plus

de profondeur selon notre analyse précédente comme observé dans la **Figure 3.15**. Pour toutes nos expériences, les RNCs apprennent un compromis entre la sensibilité et la précision en faveur de la précision, même si le score de Dice donne autant d'importance aux deux métriques.

L'impact d'une double vérité terrain

Chaque version de l'architecture testée prédit deux cartes de segmentation : une pour chacune des segmentations de chaque radiologue fourni avec l'ensemble de données. Il apparaît que les deux segmentations pour tous les prédicteurs sont presque identiques. Ainsi, les réseaux neuronaux testés établissent leur propre consensus et ne sont pas capables de reproduire le comportement de deux radiologues différents.

Après avoir observé et analysé les résultats dans la **Section 3.2.3**, nous avons décidé de concevoir et d'évaluer une architecture en tenant compte de l'ensemble de ce que nous avons appris.

3.2.5 Conception du MPU-net

L'objectif principal de cette étude est de concevoir une architecture avec une complexité aussi petite que possible. Le but est d'obtenir le meilleur compromis entre performance et nombre de paramètres.

Dans les expériences précédentes, nous avons rassemblé un certain nombre d'indices pour la conception d'une architecture compacte et performante proche du U-net. Nous avons alors conçu le MPU-net qui :

- utilise des connexions par concaténation et du sur- et sous-échantillonnage non appris,
- possède 4 niveaux avec trois couches convolutives par bloc,
- contient environ 30 000 paramètres, ce qui est suffisant compte tenu de la profondeur du réseau implémenté.

En fixant le nombre de cartes de caractéristiques comme présenté dans la **Figure 3.18**, nous avons une architecture avec 33 332 paramètres d'apprentissage et 21 couches convolutives.

Pour tester le MPU-net, nous l'avons entraîné sur l'intégralité de l'ensemble de données de segmentation des lésions de SEP disponible et avons soumis les segmentations de l'ensemble de test pour évaluation externe par les organisateurs du challenge de segmentation des lésions de SEP ISBI 2015³. La soumission en ligne donne un

³consultable sur <https://smart-stats-tools.org/lesion-challenge>

score pondéré avec différentes métriques [26].

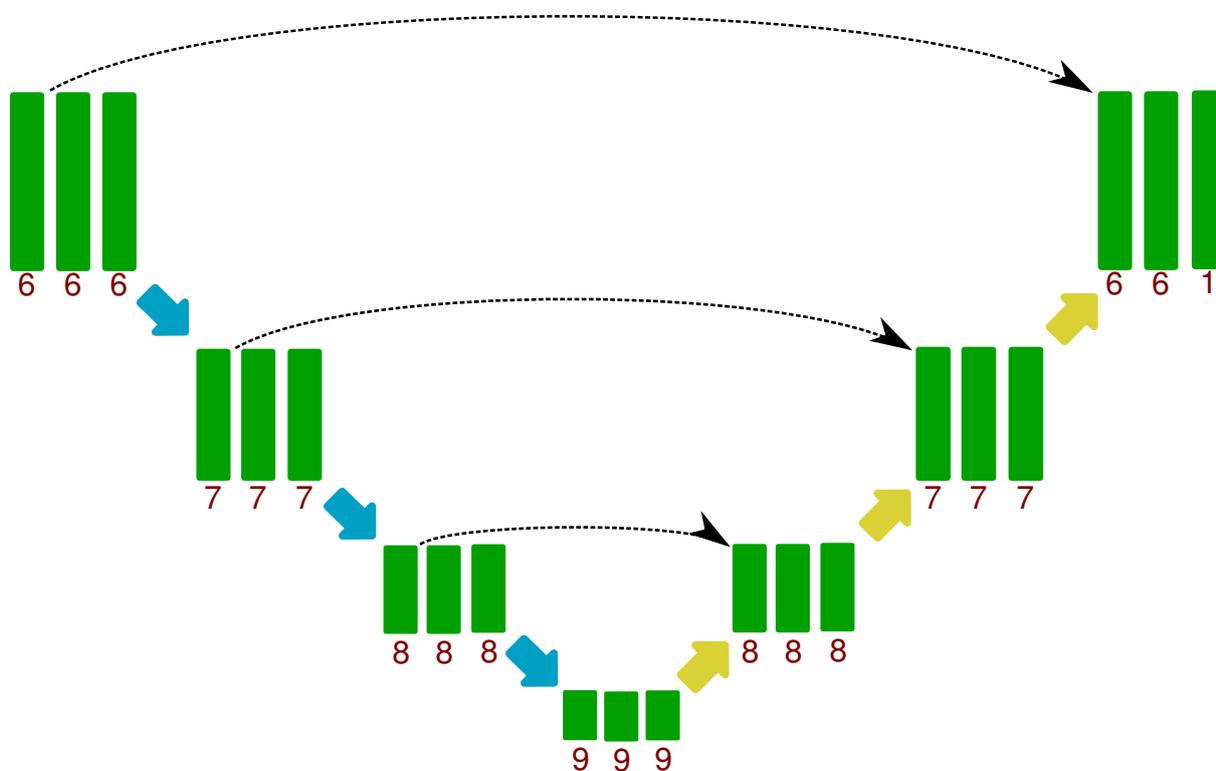


Figure 3.18 – L'architecture proposée le MPU-net avec 33 332 paramètres d'apprentissages et 21 couches convolutives.

Les segmentations produites par le MPU-net ont obtenu un score de 90.591, ce qui est un score comparable à la différence de segmentation entre deux radiologues [26].

Score	Référence	Année
93.21	Zhang <i>et al.</i> [39]	2019
93.087	Insensee <i>et al.</i> [40]	2020
92.637	Hou <i>et al.</i> [62]	2019
92.486	Hashemi <i>et al.</i> [63]	2018
92.118	Aslani <i>et al.</i> [61]	2018
92.076	Andermatt <i>et al.</i> [64]	2017
91.44	Valverde <i>et al.</i> [65]	2017
90.591	MPU-net [28]	2021
90.283	Maier <i>et al.</i> [66]	2016
90.07	Birenbaum <i>et al.</i> [67]	2016

Table 3.4 – Extrait du tableau des scores ouvert au challenge de segmentation des lésions de SEP ISBI 2015 en ligne <https://smart-stats-tools.org/lesion-challenge>.

Les publications référencées dans le **Tableau 3.4** mises en comparaison avec notre architecture utilisent toutes les pondérations fournies (T1, T2, FLAIR et DP), et sont des architectures très paramétrisées, régularisées et employant pour la plupart l'augmentation des données. Comparé aux publications référencées sur le tableau des scores de l'ISBI dans le **Tableau 3.4**, nous avons des performances acceptables en sachant que nous n'avons tiré parti que de la pondération FLAIR, entraînée uniquement sur la base d'entraînement, sans augmentation, ni couche de régularisation et sans tenir compte du temporel.

La prédiction des segmentations MPU-net se fait aisément avec des machines modestes. Par exemple, une segmentation complète de cerveau prend 12 secondes par cerveau sur un ordinateur portable avec 8 GO de RAM et une carte graphique Nvidia Quadro P1000. Ce modèle est facilement déployable dans un environnement clinique sans nécessiter une grosse infrastructure informatique ce qui facilite l'adoption de ce genre de technique d'aide pour les radiologues.

3.2.6 Discussion

Dans le but de trouver une architecture compacte pour la segmentation des lésions de SEP, nous avons mené une étude empirique pour déterminer la pertinence de certains des choix assumés dans la conception des architectures dans les RNCs en imagerie médicale et spécifiquement pour notre tâche de segmentation. Cette étude se concentre sur une adaptation simple du U-net et évalue l'influence de certaines modifications afin de fournir un aperçu de la raison pourquoi cette architecture est pertinente. Il apparaît que, pour cette tâche de segmentation de lésions, il existe une gamme assez large de complexité de modèle en termes de nombre de paramètres et de nombre de couches convolutives montrant presque toutes les mêmes performances. En d'autres termes, un modèle compact peut avoir des performances équivalentes à un modèle plus complexe pour notre tâche spécifique et cela devrait également être le cas pour d'autres tâches. De plus, un modèle trop complexe montre une diminution de la performance. La complexité minimale du modèle donne un aperçu de la complexité d'une tâche donnée. Comme la classification d'images naturelles est beaucoup plus complexe que la segmentation de lésions cérébrales, la complexité minimale des RNCs pour notre tâche est beaucoup plus faible que celle des modèles de classification de l'état de l'art, qui ont plusieurs millions voire milliards de paramètres.

Dans les conditions expérimentales données, le recours au *max-pooling* avec le suréchantillonnage aux plus proches voisins était la meilleure méthode, mais seulement avec l'emploi de connexions par concaténation. Ces connexions agissent comme des

régulariseurs pour éviter la perte de performance due à la complexité du prédicteur.

Enfin, avec toutes ces observations, il a été possible de concevoir une architecture très légère pour la segmentation des lésions de la SEP, le MPU-net, qui ne prend que des images FLAIR. Cette architecture réalise une bonne qualité de segmentation, comparable à celle des humains alors que sa conception et son entraînement sont en fait très simples puisqu'elle ne nécessite pas d'ingénierie ou de régulariseurs complexes et ne possède que 33 332 paramètres. Pour cause, cette architecture a été reprise au CHU de Poitiers pour la segmentation des lésions de SEP pour donner la possibilité aux radiologues de disposer d'une segmentation à titre indicatif.

La puissance d'une architecture donnée n'étant souvent pas expliquée ou partiellement expliquée en imagerie médicale, nous avons cherché à définir les raisons de notre choix d'implémenter une architecture U-net. Ce travail est une tentative de présenter, de clarifier et d'expliquer le processus de conception d'un RNC à travers l'exemple de notre intérêt spécifique pour la SEP

Dans ce travail, nous avons présenté certaines des raisons du succès de l'architecture U-net. Nous avons montré que pour cette tâche, avec suffisamment de couches convolutives, très peu de paramètres sont nécessaires pour obtenir une analyse de niveau proche de l'humain. La conception d'une architecture légère aussi efficace simplifierait l'utilisation des réseaux de neurones artificiels dans les hôpitaux en ne nécessitant pas de serveurs coûteux. Elle pourrait également contribuer à expliquer plus facilement les représentations apprises.

Ce travail a donné lieu à la conception d'une architecture simple, compacte et performante: le MPU-net. Comme détaillé plus tôt, cette architecture a été conçue dans le but de rester le plus simple possible. Cependant, avec l'ajout de quelques couches de régularisation il devrait être possible d'améliorer encore les capacités de régularisation du réseau de neurones convolutifs. Cela a donc mené à l'élaboration du MPU-next[29].

3.3 Du MPU-net au MPU-next

Le MPU-net (voir **Section** 3.2.5) est une architecture compacte possédant un nombre limité de cartes de caractéristiques en sortie de ses couches convolutives et qui possède uniquement 33 332 paramètres d'apprentissage. Cette architecture est compacte, mais aussi très simple, en effet, elle n'a recours à aucun type de régularisation. Nous proposons, ici, d'adapter le MPU-net pour profiter de meilleures performances en

ajoutant des couches de régularisations et en assemblant les cartes de segmentations prédites à différentes profondeurs.

3.3.1 Raffinement de l’architecture

La façon la plus évidente pour améliorer le MPU-net est d’ajouter des couches de régularisation comme les couches d’abandon ou *dropout* pour éviter le surapprentissage et de normalisation pour accélérer la convergence, diminuer l’impact de l’initialisation et obtenir de meilleures performances (voir **Section** 2.4.3). Nous proposons aussi d’améliorer notre architecture pour ajouter de multiples segmentations à plusieurs échelles, pour faire de l’apprentissage collaboratif à l’image du U-net++ [68].

Le modèle du MPU-net++

Nous avons choisi d’évaluer la pertinence d’ajouter de multiples niveaux de segmentation au sein du MPU-net en s’inspirant du U-net++ pour créer le modèle d’architecture du MPU-net++ dans la **Figure** 3.19. La segmentation prédite par le modèle du MPU-net++ est la somme de trois segmentations qui sont effectuées à différentes profondeurs du réseau. Le MPU-net++ est constitué d’une agrégation de plusieurs encodeurs et décodeurs mis en série pour produire une segmentation finale multiéchelle. Le réseau est donc formé de plusieurs segmenteurs entraînés en même temps durant un apprentissage collaboratif [69]. La généralisation du réseau est ainsi améliorée en général. Les dernières couches de chaque bloc sont connectées directement aux blocs suivants par des connexions par concaténations au sein du même niveau pour fusionner l’analyse du réseau à plusieurs échelles à tous les niveaux du réseau. Les connexions par concaténation au sein d’un réseau permettent au réseau de neurones d’apprendre des caractéristiques discriminantes et éviter les redondances au sein du réseau vers une diminution du nombre de paramètres d’apprentissage comme dans le DenseNet [20].

Les architectures sur le modèle du MPU-net++ possèdent 10 blocs en plus de 2 couches convolutives $1 \times 1 \times 1$ contre 7 blocs pour le modèle du MPU-net. Le modèle du MPU-net++ est donc plus profond que le modèle du MPU-net partageant les mêmes blocs.

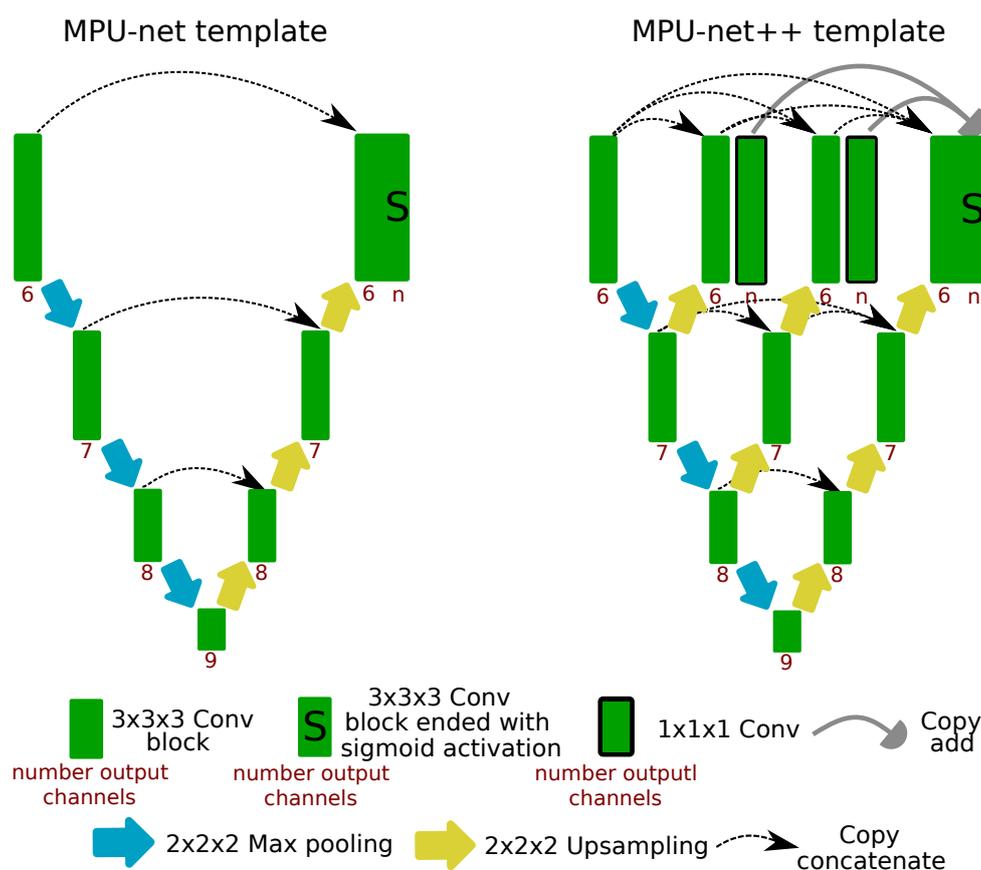


Figure 3.19 – Les modèles d'architectures du MPU-net et du MPU-net++. Le contenu des blocs est illustré dans la **Figure 3.20**.

L'ajout de couches de régularisation

Dans le MPU-net, un bloc est composé de trois couches convolutives chacune suivie par une activation *leaky ReLU* comme illustré dans la première colonne de la **Figure 3.20**. Nous avons évalué plusieurs adaptations de blocs pour cette étude. Dans la seconde colonne de la **Figure 3.20** une couche de normalisation par lot entre les couches convolutives et *leaky ReLU* ont été ajoutées ainsi qu'une couche d'abandon après la première activation *leaky ReLU* du bloc. De cette façon, les sorties des couches convolutives sont normalisées avant l'activation et la couche d'abandon n'agit qu'au début de la propagation de l'information dans le bloc. De plus, dans la troisième colonne de la **Figure 3.20**, inspirés par les blocs résiduels du ResNet [19] qui permettent à une architecture très profonde de gagner en performance, nous avons aussi testé la transformation de nos blocs en blocs résiduels dans l'encodeur en ajoutant une connexion par sommation entre la première couche convolutive et la sortie du bloc. Le modèle du MPU-net++ avec trois convolutions par niveau dans la deuxième ligne de la **Figure 3.20** atteint 32 couches convolutives contre 21 couches pour le modèle initial du MPU-net. Nous avons donc évalué une architecture avec supervision profonde

sur le modèle du MPU-net++, mais avec des blocs de deux couches convolutives au lieu de trois dans la troisième ligne de la **Figure 3.20** permettant de n'atteindre seulement que 22 couches convolutives.

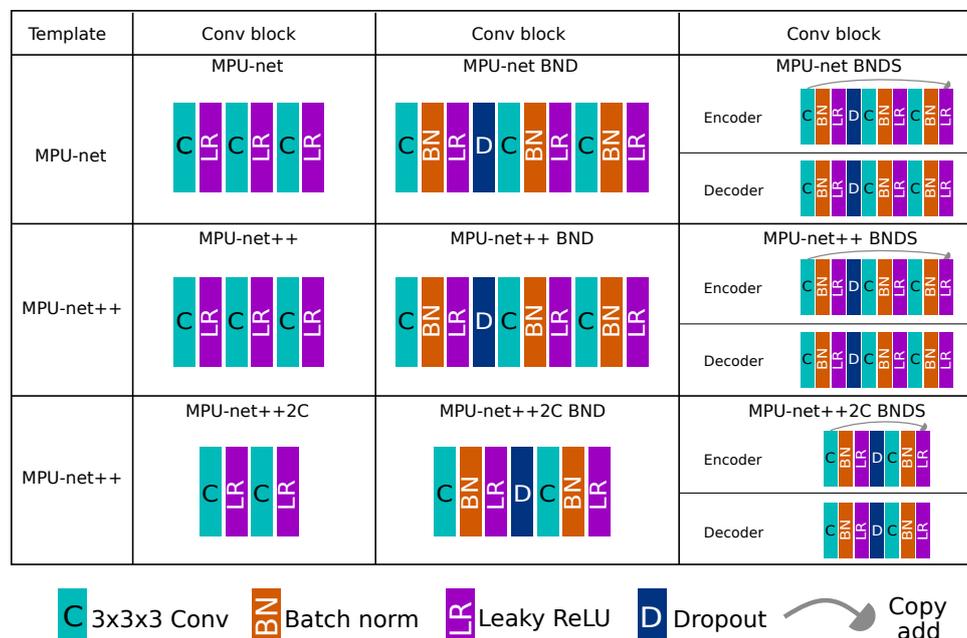


Figure 3.20 – Les différentes variations des blocs testés avec les modèles d'architectures présentés dans la **Figure 3.19**.

Nous avons donc, en tout, évalué 9 architectures différentes en testant les trois différentes variations de bloc sur le modèle du MPU-net, du MPU-net++ et du MPU-net++ à deux convolutions illustrées dans les **Figures 3.19** et **3.20**.

3.3.2 Résultats et discussion

Protocole d'apprentissage et de test

Pour rester consistant avec l'élaboration du MPU-net, nous avons repris exactement les mêmes images, prétraitées de la même façon avec les mêmes règles et paramètres d'apprentissage que dans l'analyse précédente à l'exception de l'extraction des patches et le nombre d'époques (voir **Section 3.2.2**). Pour les entraînements précédents, nous avons fixé le lot comme 5 patches pour chaque patient extraits aléatoirement dans une zone du cerveau et avons fixé une époque comme 100 lots. Ici, nous sommes revenus sur ces définitions pour agrandir la taille des lots pour apprendre sur des lots plus représentatifs de la diversité des textures du cerveau. Dans la **Section 3.2.2**, un lot correspond à 85 patches, nous avons revu, ici, la définition d'un lot comme 1024

patches et d'une époque comme étant 3800 patches extraits pour chaque patient contre 500 précédemment dans la **Section 3.2.2**. Chaque modèle a été entraîné durant 30 époques, période à partir de laquelle nous avons observé la convergence pour chacun des modèles.

Pour les performances de test, les patches ont été extraits de la même manière que dans l'analyse précédente (voir **Section 3.2.2**) avec un seuillage à 0.6 qui a été observé comme moins sensible aux faux positifs que le seuillage à 0.5 précédent.

Résultats

Les segmentations ont été évaluées avec le score de Dice. Les résultats sont présentés dans la **Figure 3.21** sous forme de boîte à moustaches. La boîte représente l'intervalle entre le second et le troisième quartile et les moustaches représentent la distribution du reste des valeurs exceptées les valeurs aberrantes en diamant. La médiane est représentée par le segment à l'intérieur de la boîte.

D'une façon générale, dans la **Figure 3.21** croisée avec la **Figure 3.20**, les boîtes à moustaches de la première colonne sans couche de régularisation ont un intervalle de répartition des scores plus grands que la seconde colonne avec régularisation et un score médian plus bas. Cela indique bien que les couches de régularisation ajoutées (normalisation par lot et couche d'abandon) apportent à la fois une meilleure stabilité et une meilleure généralisation. Les blocs résiduels (troisième colonne) par rapport aux blocs régularisés (seconde colonne) apportent un léger gain de score de Dice médian de l'ordre 0.02 avec l'architecture du modèle MPU-net (première ligne), alors que pour le modèle d'architecture du MPU-net++ (seconde ligne) il y a un gain important de score de Dice médian d'environ 0.13 pour la première segmentation et de 0.06 pour la deuxième segmentation. Il y a donc un plus gros apport pour la seconde ligne que pour la première. Cela s'explique par le fait que le modèle du MPU-net++ est plus profond que le modèle du MPU-net et que les blocs résiduels permettent aux architectures plus profondes d'améliorer encore leurs résultats plutôt que de perdre en performances. Cependant, pour la troisième ligne avec le modèle du MPU-net++ avec deux couches convolutives par bloc, les scores de Dice médians sont meilleurs avec les blocs régularisés (seconde colonne) qu'avec les blocs résiduels (troisième colonne). Notre hypothèse est que, dans ce cas, l'architecture de la troisième ligne est trop peu profonde pour bénéficier des apports des blocs résiduels qui, ici, perturbent la segmentation du réseau.

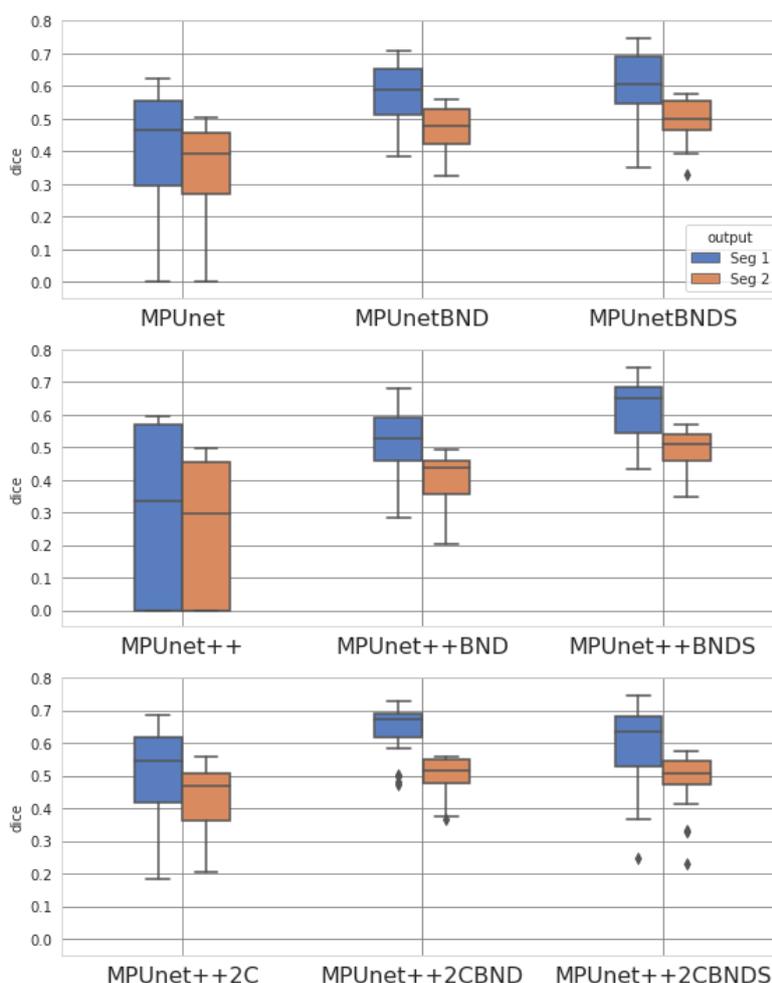


Figure 3.21 – Graphique en boîte à moustaches du score de Dice sur l'ensemble de test pour chaque architecture testée. La boîte représente le deuxième et troisième quartile et les moustaches contiennent le reste de la distribution des données exceptées les valeurs aberrantes représentées par les diamants.

Le MPU-net++ sans aucune régularisation (seconde ligne, première colonne de la **Figure 3.21**) possède un très grand intervalle de distribution de score de Dice de 0.57 et 0.46 contre 0.4 et 0.3 pour sa version régularisée (seconde ligne, deuxième colonne). Le MPU-net++ avec trois couches convolutives par bloc est si profonde qu'il nécessite obligatoirement des couches de régularisation pour atteindre une analyse reproductible et généralisable.

Dans la **Figure 3.21**, les meilleures performances sont atteintes, en général, sur la première carte de segmentation experte. En pratique, comme évoqué précédemment en 3.2.4, les deux cartes de segmentations prédites sont presque identiques et la première carte de segmentation détecte généralement plus de lésions avec des contours

plus larges ce qui donne plus d'importance pour les voxels segmentés sur cette carte avec la fonction de coût de Dice par lot **Equation 3.1**.

Le meilleur score de Dice médian avec le plus petit intervalle interquartile et les résultats les moins dispersés en général est atteint pour l'architecture du MPU-net++ avec deux convolutions par blocs avec régularisation non résiduels (troisième ligne, deuxième colonne). Pour cette version, le score de Dice médian atteint 0.68 et 0.52 pour les deux segmentations expertes avec un intervalle interquartile de 0.8 et 0.7. La meilleure adaptation du MPU-net est donc la version calquée sur le modèle du U-net++, avec 3 segmenteurs entraînés conjointement et des connexions par concaténation entre l'encodeur et tous les décodeurs d'un même niveau avec seulement deux couches convolutives par bloc espacées par une couche d'abandon et des couches de normalisation par lot. C'est une adaptation du MPU-net légèrement plus profonde avec des couches de régularisation, plus de connexions par concaténation et plusieurs décodeurs et segmenteurs. Nous considérons cette architecture comme la plus performante et lui attribuons le nom de MPU-next.

Par rapport au MPU-net conçu dans la **Section 3.2.5**, le MPU-next gagne dans la **Figure 3.21** 0.12 de score de Dice médian pour chaque segmentation avec un intervalle interquartile de 0.8 pour chacune des vérités terrain contre 2.5 et 1.9. C'est donc une version plus performante, plus robuste et plus reproductible. Le MPU-net possède 21 couches convolutives et 33 332 paramètres d'apprentissage alors que le MPU-next possède 22 couches convolutives et 37 935 paramètres d'apprentissage. Nous avons donc obtenu un gain substantiel en performance et robustesse avec l'ajustement de l'architecture pour un coût en complexité d'une couche convolutive et de 4 603 paramètres d'apprentissage, ce qui nous place toujours avec une complexité bien plus faible que de nombreuses architectures de la littérature du **Tableau 3.3**.

Nous avons montré que le MPU-next est une architecture performante pour la segmentation des lésions de SEP. Dans la section suivante, cette architecture est adaptée pour la segmentation automatique des nouvelles lésions de SEP qui sont importantes à détecter pour le diagnostic et le suivi de la maladie.

3.4 Adaptation du MPU-next pour la segmentation des nouvelles lésions de SEP

La détection des nouvelles lésions de SEP a un fort intérêt médical. L'apparition de nouvelles lésions fait partie des critères de McDonald [5] (voir **Section** 2.3.4) et est, en pratique, effectuée par les radiologues par comparaison visuelle des images IRM acquises à plusieurs mois ou années de différence. La détection des nouvelles lésions permet d'estimer l'évolution de la charge lésionnelle, de la progression de la maladie [70] et constitue une information de choix pour le neurologue pour orienter le traitement médicamenteux. La détection des nouvelles lésions ou de leur changement de volume est comme la segmentation des lésions de SEP une tâche répétitive, chronophage et sujette à la variation intra- et inter-expert [71].

3.4.1 Contexte

Les travaux réalisés ici ont été motivés par le challenge du MSSEG2 [72]⁴. Le challenge a pour but de rassembler des techniques proposées par différents groupes de recherche sur la détection des nouvelles lésions de SEP sans les lésions qui s'agrandissent à partir de deux images FLAIR acquises consécutivement séparées par 1 à 3 années.

En pratique, la détection automatique des nouvelles lésions a été assez peu adressée dans la littérature. Cela s'explique d'une part par la disponibilité de données, en effet, il est plus difficile de se procurer des images consécutives des mêmes patients plutôt que des images sans suivi. La segmentation des nouvelles lésions est aussi plus laborieuse, car elle nécessite la comparaison de deux images par le radiologue en même temps avec des changements de contraste et toutes les variations potentielles entre deux mêmes examens IRM. La détection des nouvelles lésions est aussi une tâche plus difficile que la détection des lésions, car il y a moins de nouvelles lésions que de lésion, qu'il faut pouvoir détecter les lésions avant de pouvoir détecter les nouvelles lésions et qu'il faut un prétraitement robuste pour rendre comparable des images successives. La segmentation automatique des nouvelles lésions de SEP, pourrait être assimilé à une soustraction des deux images FLAIR puisque les lésions sont en hyperintensité. Cependant, les erreurs d'alignement, les artefacts et les petites différences d'intensité rendent la soustraction des deux images très sujette à la détection de faux positifs [73].

⁴<https://portal.fli-iam.irisa.fr/msseg-2/>

Plusieurs méthodes et techniques ont été proposées pour détecter les nouvelles lésions ou l'élargissement des lésions dans la littérature. Les premières méthodes de détection automatique des lésions de SEP se sont basées sur l'analyse des intensités ou des déformations entre les images successives [74] avant de fusionner les deux approches dans les travaux de Cabezas *et al.* [23] et Salem *et al.* [75]. D'autres approches, basées sur la détection des lésions de SEP dans les deux images successives, ont aussi été élaborées avec les outils de segmentation toujours plus performants. Schmidt *et al.* [76] présentent une méthode pour corriger la segmentation des lésions de deux images FLAIR et T1 successives pour produire la carte des changements significatif des lésions en se basant sur leur outil de segmentation des lésions [77] et une analyse statistique des différences des images FLAIR alignées avec les intensités corrigées et normalisées. McKinley *et al.* [71] utilisent, quant à eux, leur RNC générant la segmentation des lésions avec une carte de confiance pour segmenter les lésions de deux examens consécutifs pour produire la carte des nouvelles segmentations en s'appuyant sur les changements significatifs entre les deux segmentations. Gessert *et al.* [78] font le choix d'employer une architecture proche du U-net avec deux encodeurs (un pour chaque image FLAIR à comparer) liés entre eux par des mécanismes d'attention pour générer la segmentation des nouveaux voxels appartenant aux lésions de façon complètement supervisée. Cependant, ces méthodes ne sont pas spécifiques aux nouvelles lésions et ne distinguent pas les nouvelles lésions des lésions qui s'agrandissent.

Pour le challenge du MSSEG2, nous avons choisi de concevoir une solution basée sur le MPU-next voir **Section** 3.3 préalablement développé [29] pour développer une architecture spécifique à la détection des nouvelles lésions [30]. La méthode de Gessert *et al.* [78] basée sur un U-net avec un encodeur par image FLAIR, a servi de modèle pour l'adaptation du MPU-next à la tâche. Inspirés par le travail de Schmidt *et al.* [76] et de McKinley *et al.* [71], nous avons voulu décomposer la segmentation de chaque image FLAIR pour ensuite fusionner ces informations pour créer la carte de segmentation des nouvelles lésions. Nous avons donc adapté le MPU-next dans l'idée de créer une carte de segmentation pour chaque image FLAIR et de comparer ces cartes dans un second temps pour réaliser la segmentation des nouvelles lésions uniquement de façon complètement supervisée.

Données

Les images FLAIR fournies par le challenge MSSEG2 ont été acquises dans différents centres radiologiques français avec différents scanners IRM, de fabricants différents, avec différents champs magnétiques et des paramètres d'acquisition FLAIR différents

par l’observatoire français de la sclérose en plaques (OFSEP). L’ensemble de données est constitué de 100 patients avec deux images FLAIR successives avec les segmentations de 4 neuroradiologues et leur consensus. Les images sont séparées en deux ensembles de données:

- 40 examens consécutifs d’apprentissages avec les segmentations fournies,
- 60 examens de test avec uniquement les images FLAIR consécutives.

Prétraitement

Les images ont été préalablement alignées par l’organisation du challenge en amenant chaque image FLAIR dans le même espace de représentation à “mi-chemin” entre les deux images. Puis, nous avons appliqué les prétraitements conseillés par l’organisation⁵ consistant en :

- L’extraction du cerveau par alignement sur atlas,
- la correction des inhomogénéités de champs avec l’algorithme N4 de Tustison [44].

Nous avons aussi appliqué une étape de ré-échantillonnage à $0.8 \times 0.8 \times 0.8 \text{ mm}^3$, car toutes les images n’étaient pas à la même résolution spatiale. Les histogrammes des images FLAIR ont été mis en correspondance par la méthode de Nyul *et al.* [48] et les intensités ont été normalisées par soustraction de la moyenne et division par l’écart type pour rendre les intensités comparables centrées réduites.

3.4.2 Méthode proposée

Nous avons décidé d’adapter notre MPU-next pour la tâche de segmentation des nouvelles lésions de SEP à partir des images FLAIR consécutives fournies et prétraitées.

Architecture

Nous avons conçu l’architecture de la **Figure 3.22** pour segmenter les nouvelles lésions de SEP. Elle est composée d’un bloc de segmentation et d’un bloc de raffinement. Le bloc de segmentation est conçu pour segmenter les lésions sur chaque image FLAIR séparément avec le même réseau, de façon siamoise. Le bloc de raffinement agrège les cartes de segmentations produites sur chaque image FLAIR pour prédire la carte des nouvelles lésions. En pratique, nous ne pouvons garantir le fonctionnement du réseau de neurones du fait de la complexité de ces modèles, de l’analyse interconnectée et

⁵Anima MS longitudinal processing script RRID:SCR 017072 <https://anima.irisa.fr>

hiérarchisée créant leur propre représentation des données basées uniquement sur l'apprentissage supervisé de la tâche.

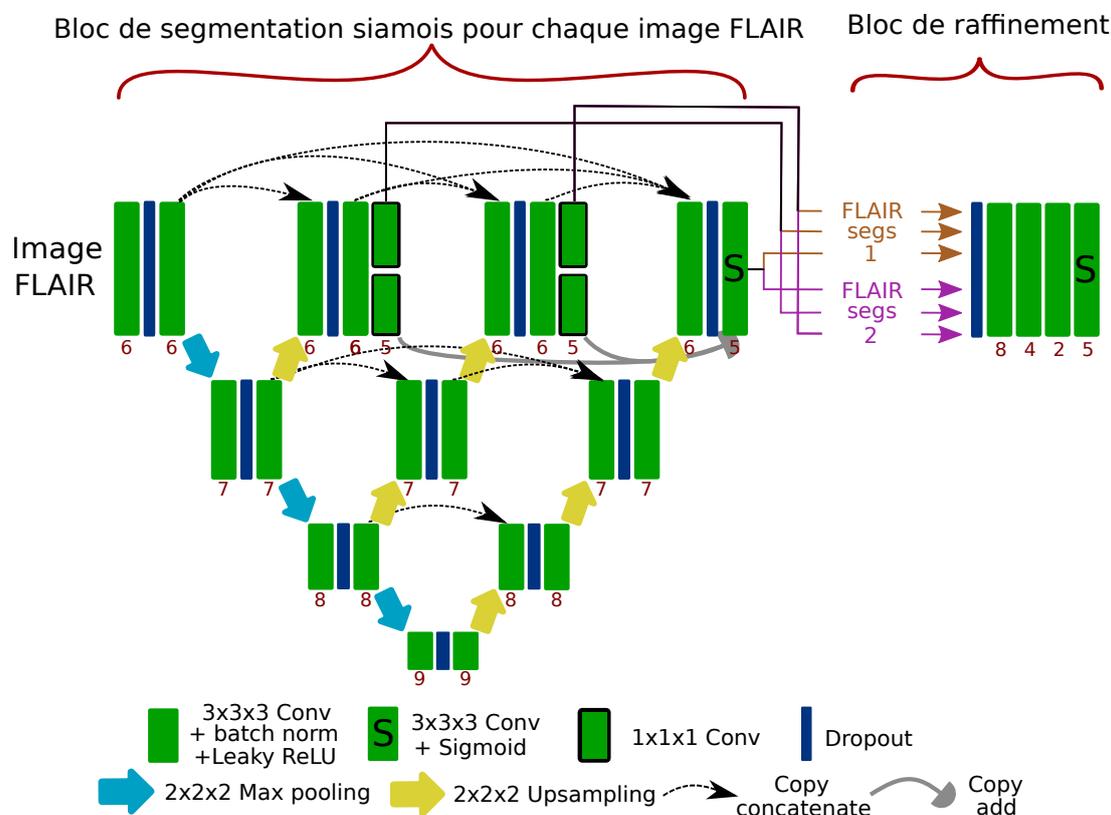


Figure 3.22 – Architecture proposée pour la segmentation des nouvelles lésions au challenge MSSEG2 : l'architecture contient un MPU-next siamois comme bloc de segmentation et bloc de raffinement convolutif.

Bloc de segmentation

Pour analyser de la même façon les deux images FLAIR d'entrée dans le but de segmenter les zones d'intérêt dans chacune des entrées, nous avons utilisé le MPU-next (voir **Section 3.3**) conçu pour être performant, robuste et léger avec le potentiel d'une meilleure généralisation pour les petits ensembles de données grâce à son nombre réduit de paramètres d'apprentissage (voir **Section 3.2.5**).

Bloc de raffinement

Le bloc de raffinement prend comme entrée les cartes de segmentations des différents décodeurs issus du bloc de segmentation pour chaque image FLAIR consécutive

en entrée. Ce bloc est composé d'une couche d'abandon suivie de quatre blocs composés d'une couche convolutive suivie d'une normalisation par lot puis d'une activation *leaky ReLU*.

Ce bloc est conçu pour affiner les différentes segmentations obtenues à différentes profondeurs du bloc de segmentation pour chaque entrée pour prédire les nouvelles lésions.

Apprentissage

Le réseau prend en entrée les deux images FLAIR consécutives et génère en sortie chaque segmentation fournie comme source de supervision (le consensus et les 4 segmentations expertes). L'entraînement avec les 5 segmentations fournies est réalisé dans le but de laisser le réseau de neurones à converger vers son propre consensus. Le réseau travaille avec des patches de $56 \times 56 \times 56$ voxels extraits dans le cerveau avec une probabilité de 80% de contenir au moins un voxel appartenant à une lésion. La taille de patch a été agrandie par rapport à la taille de patch de $32 \times 32 \times 32$ habituel pour capturer une plus grande partie de la texture environnantes les lésions. L'extraction des patches privilégiée autour des voxels de lésions a pour but d'équilibrer l'importance des nouvelles lésions qui occupent un volume extrêmement faible des images. La taille de lot d'apprentissage est fixée à 160 patches et une époque est définie pour contenir 3488 patches provenant de chaque patient d'entraînement. Le modèle est entraîné avec l'optimiseur Adam [79] avec un taux d'apprentissage à 0.004 pendant 40 époques à partir desquelles nous avons observé la convergence. Le taux d'apprentissage a été en pratique diminué de 10% après chaque époque pour affiner la convergence au fur et à mesure, en effet, nous avons observé que baisser le taux d'apprentissage au fur et à mesure améliorait l'apprentissage. Le coût de Dice a été adopté comme fonction de coût.

Test

Pour le test, nous n'avons conservé en sortie que la prédiction correspondant au consensus pour chaque patch et nous avons moyenné ces patches extraits régulièrement avec un recouvrement de 28 voxels dans chaque dimension. La carte de prédiction obtenue a été ré-échantillonnée vers la résolution originale et binarisée pour ne garder que les valeurs prédites à plus de 0.5.

Equipe	Score de Dice moyen	Equipe	Score F1 moyen
MedICL	0,507	Mediaire-B	0,541
LaBRI-IQDA	0,500	Empenn	0,532
SNAC	0,485	Mediaire-A	0,525
LaBRI-D&E	0,472	LaBRI-IQDA	0,517
NVAUTO	0,469	SNAC	0,514
LaBRI-lw	0,453	MedICL	0,500
New Brain	0,451	LaBRI-D&E	0,498
ITU	0,443	ITU	0,480
Mediaire-B	0,437	New Brain	0,477
Mediaire-A	0,432	NVAUTO	0,465
Empenn	0,424	LaBRI-lw	0,464
McEwan-IM	0,423	McEwan-IM	0,453
PVG	0,414	PVG	0,448
Neuropoly-1	0,411	SCA-SimpleUNet	0,446
IAMLAB	0,411	LYLE	0,441
LYLE	0,409	SCAN	0,433
Neuropoly-2	0,409	Neuropoly-1	0,424
SCAN	0,403	IAMLAB	0,414
SCA-SimpleUNet	0,400	Neuropoly-3	0,414
I3M	0,398	Neuropoly-2	0,410
Neuropoly-3	0,379	HufsAIM	0,405
The NoCoDers	0,365	The NoCoDers	0,382
Vicorob	0,358	Vicorob	0,369
HufsAIM	0,346	CMIC	0,361
CMIC	0,330	I3M	0,358
MIAL	0,309	MIAL	0,333
SCA-withPriors	0,224	LIT	0,242
LIT	0,214	SCA-withPriors	0,216
IBBM	0,155	IBBM	0,143

Table 3.5 – Les tableaux des scores moyens calculés sur l'ensemble de test par l'organisation du challenge MSSEG2: à gauche le tableau décroissant des scores de Dice et à droite le tableau décroissant des scores F1. Nos scores sont en **gras**.

3.4.3 Résultats du challenge

Vingt-neuf techniques ont été soumises au challenge MSSEG2 de septembre 2021 et sont toutes présentées dans le recueil du challenge [72] et les résultats ont été publiés [80] et sont résumés dans le **Tableau 3.5**.

Dans le **Tableau 3.5** des scores moyens sur la base de test, nous pouvons observer que nous avons atteint en moyenne 0,398 de score de Dice et 0,358 de score F1 pour la segmentation des nouvelles lésions et atteint respectivement les places de 20^e/29 et 25^e/29 pour le score de Dice et le score F1. Nous sommes donc dans les

deux cas dans la partie inférieure des scores. Pour comprendre le succès des autres méthodes, nous avons passé en revue les méthodes avec les meilleurs scores de Dice et F1.

Analyse des meilleurs scores

La méthode MedICL [81] (1^{er} score de Dice et 6^e F1) proposent une adaptation de leur 2.5D Tiramisu Net de segmentation des lésions de SEP [39] en utilisant la distance au moindre carré comme fonction de coût en augmentant artificiellement et aléatoirement les données pour rendre le modèle plus générique et plus robuste aux éventuels changements d'acquisition.

Les méthodes Mediaire A&B [82] (1^{er} et 3^e scores F1 et 9^e et 10^e scores de Dice) font usage d'un U-net 2D entraîné sur chaque plan (axial, sagittal, longitudinal) en même temps avec comme fonction de coût une adaptation de l'entropie croisée et le coût de Dice. Leur méthode emploie aussi l'augmentation artificielle de données. Les prédictions sont effectuées avec le même réseau sur les volumes découpés en axial, sagittal et longitudinal. Les prédictions de chaque plan sont fusionnées à la majorité. La différence en Mediaire-A et Mediaire-B est uniquement sur l'ensemble d'apprentissages. La méthode A est entraîné uniquement sur l'ensemble d'apprentissage du challenge alors que la méthode B est aussi entraînée sur ensemble d'images privé.

La méthode LaBRI-IQDA [83] (2^e score de Dice et 4^e score F1) ont recours à un U-net 3D entraîné avec le coût de Dice avec augmentation artificielle des données.

La méthode Empenn [84] (11^e score de Dice et 2nd score F1) utilise un U-net 3D entraîné avec le coût de Dice et l'entropie croisée avec augmentation des données. Ils se servent des images d'apprentissage du challenge en excluant 3 patients et en rajoutant 53 autres patients de leur base privée. Les images sont post-traitées pour éliminer les lésions trop petites.

En résumé, les meilleures méthodes résumées plus haut emploient globalement des réseaux de neurones artificiels de type U-net 2D ou 3D avec les coûts de Dice, d'entropie croisée et les moindres carrés. Ils mettent tous à profit l'augmentation artificielle des données pour tenter de rendre les modèles plus robustes aux bruits et aux différences d'acquisition. Une partie des meilleures méthodes disposent de données externes, mais pas l'entièreté.

Il serait intéressant de pouvoir soumettre notre architecture en ajoutant l'entropie croisée comme fonction de coût ou les moindres carrés et en ajoutant l'augmentation artificielle des données qui est présente dans toutes les meilleures méthodes.

3.5 Conclusion

Nous nous sommes largement inspirés du U-net, reconnu pour la segmentation, pour créer trois différentes architectures : le MPU-net et le MPU-next pour la segmentation des lésions de SEP et le MPU-next siamois pour la segmentation des nouvelles lésions de SEP.

Le MPU-net a été conçu en comparant les différents choix architecturaux qui donnent les meilleures performances avec des entraînements robustes. Cette architecture a été conçue pour être légère (environ 33 000 paramètres d'apprentissage et 21 couches convolutives) et a pour avantage de nécessiter moins de données d'apprentissage pour généraliser qu'un modèle plus complexe.

Le MPU-net a par la suite été amélioré pour créer le MPU-next qui rajoute des couches de régularisation et de multiples segmentations à différentes profondeurs du réseau qui sont agrégées pour réaliser la segmentation finale. Le MPU-next s'est montré plus performant et plus reproductible que le MPU-net avec seulement environ 37 000 paramètres et 22 couches convolutives.

Dans le but de segmenter les nouvelles lésions de SEP, nous avons conçu le MPU-next siamois. Les résultats de segmentation indiquent qu'un entraînement avec augmentation artificielle des données et un changement de fonction de coût auraient très probablement amélioré les performances de notre tentative de soumission au challenge MSSEG2 de segmentation des nouvelles lésions de SEP.

4

La réduction des données d'apprentissage

Sommaire

4.1	Apprentissage autosupervisé pour la segmentation des lésions de SEP	97
4.1.1	Contexte	98
4.1.2	Matériels et méthodes	99
4.1.3	Résultat et analyse	103
4.2	Nombre de patients nécessaire	106
4.2.1	Matériels et méthodes	107
4.2.2	Résultat et analyse	108
4.3	Apprentissage semi-supervisé de la segmentation des lésions	109
4.3.1	Introduction	109
4.3.2	Évaluation de l'ensemble de données	111
4.3.3	Conception d'une méthode semi-supervisée pour la segmenta- tion des lésions de SEP	114
4.4	Conclusion	120

LES images IRM ou plus généralement médicales sont des données sensibles et limitées en quantité. De plus, l'annotation de ces images est coûteuse en temps et nécessite des radiologues. L'apprentissage profond d'une tâche nécessite des échantillons représentatifs, variés et annotés afin d'être capable de généraliser son analyse. L'emploi des réseaux de neurones est donc réputé pour nécessiter beaucoup de données d'apprentissage et d'être limité par la disponibilité de celles-ci pour son recours en imagerie médicale.

Dans ce chapitre, le travail sur la réduction en données d'apprentissage pour la segmentation des lésions de SEP est présenté. Nous avons dans un premier temps essayé de réduire le besoin en images en mettant à profit l'autosupervision. En voyant que l'autosupervision n'était pas nécessaire pour améliorer les données, nous avons mené des travaux pour déterminer si nous n'employions pas déjà assez d'images. Enfin, après avoir déterminé un nombre suffisant de données d'apprentissage, nous avons essayé de réduire ce nombre en tirant parti de l'apprentissage semi-supervisé.

Les travaux décrits dans ce chapitre ont fait l'objet des publications scientifiques suivantes : [85],[29].

4.1 Apprentissage autosupervisé pour la segmentation des lésions de SEP

L'apprentissage autosupervisé est une méthode qui a le potentiel d'améliorer un apprentissage supervisé en disposant des données non annotées et l'apprentissage par

transfert (voir **Section 2.4.3**). Nous avons donc travaillé sur la mise en place d'un apprentissage autosupervisé pour la segmentation des lésions de SEP [85].

4.1.1 Contexte

L'apprentissage autosupervisé est un type d'apprentissage faiblement supervisé décrit par Doersch *et al.* [86]. L'apprentissage autosupervisé repose sur l'apprentissage par transfert. Lorsqu'il n'y a pas assez de données pour apprendre correctement un réseau de neurones, il est usuel d'appliquer l'apprentissage par transfert qui consiste à réentraîner en partie ou en totalité un réseau de neurones déjà appris sur une tâche le plus proche possible. L'idée sous-jacente est que l'extraction des caractéristiques apprises lors de deux apprentissages proches est la même ou très semblable. Ainsi, lorsque le réseau a convergé pour une tâche, il est moins coûteux en calcul et en exemple d'apprentissage de réapprendre ce réseau sur une nouvelle tâche proche plutôt que d'apprendre un réseau de zéro sur la nouvelle tâche. L'apprentissage autosupervisé consiste, dans un premier temps, en l'entraînement d'un réseau de neurones sur une tâche non supervisée (c'est-à-dire qui n'a pas besoin d'annotation), apparemment inutile pour apprendre le contexte. Puis, le réseau appris entier ou partiel est réutilisé (ou transféré) pour apprendre la tâche cible supervisée d'intérêt. Comme le réseau est déjà entraîné à analyser les images sur une tâche similaire, moins d'échantillons annotés sont nécessaires.

Doersch *et al.* [86], entraînent dans un premier temps, de façon non supervisée, un réseau siamois sur la classification de position relative de patches d'images adjacents. Cette tâche prétexte, sans intérêt à première vue, permet d'apprendre l'extraction de caractéristiques intéressantes à partir des images sans avoir besoin d'annotation. Dans un second temps, ils réemploient la partie siamoise de leur réseau appris sur la tâche prétexte pour l'apprendre sur la tâche d'intérêt supervisée de détection d'objet. Ils montrent qu'en préentraînant une partie de leur réseau sur la tâche prétexte non supervisée ils obtiennent de meilleures performances.

Partant de l'hypothèse que nous n'avons pas assez de données pour apprendre un réseau de neurones performant pour la segmentation des lésions de SEP, nous avons proposé une approche autosupervisée en s'inspirant de Doersch *et al.* [86]. L'avantage de l'autosupervision est que cela permet de préentraîner un réseau et donc de tirer parti des avantages de l'apprentissage par transfert sans pour autant contraindre son apprentissage à une architecture donnée avec un nombre d'entrées fixé. En pratique, l'apprentissage autosupervisé a très peu été mis à profit pour la segmentation

des lésions de SEP. Denner *et al.* [87] appliquent une forme d'autosupervision en apprenant conjointement la segmentation des changements lésionnels et l'approximation du champ de déplacement à partir des deux images d'entrées. Le travail de To *et al.* [88] consiste à apprendre de façon non supervisée un générateur de nouvelles lésions et d'apprendre un réseau de neurones à partir de ce générateur de façon autosupervisée, ce qui n'entre pas dans notre définition d'autosupervision. Nous présentons ici le travail sur la conception d'une méthode d'autosupervision pour la segmentation des lésions de SEP et l'évaluation de cette méthode [85].

4.1.2 Matériels et méthodes

Nous avons rassemblé plusieurs ensembles de données différents, spécifiques ou non pour la segmentation des lésions de SEP. Les images ont été séparées en patches 3D envoyés à un premier réseau de neurones dont la tâche prétexte non supervisée est de prédire la position (coordonnées x, y et z) du patch dans le cerveau. Dans un deuxième temps, une partie de ce réseau de neurones appris est réemployée dans une architecture proche du U-net pour apprendre la tâche supervisée d'intérêt de segmentation des lésions de SEP.

Données

Ensembles de données Pour cette étude, nous avons voulu travailler avec le plus de données possibles et avons donc rassemblé quatre ensembles d'images publiques différents sur la segmentation des lésions de SEP, à savoir, l'ensemble du MICCAI 2008 [25] et 2016 [32], de l'ISBI 2015 [26], et l'ensemble public de Lesjak *et al.* [89]. Deux autres ensembles publics ont été mis à profit : l'OASIS3 [90] avec des patients cognitifs normaux et en déclin et les données du MICCAI BraTS challenge 2017 [91] avec des tumeurs cérébrales. Plus d'informations sur les ensembles d'images sont présentées dans le **Tableau 4.1**. L'ensemble MICCAI 2016 a été conservé comme test extérieur pour évaluer la capacité de généralisation du modèle appris avec toutes les acquisitions disponibles.

Origines	N. examens	Acquisitions	SEP	Sous-ensemble
MICCAI 2008	20	T1, T2, FLAIR	OUI	Apprentissage
MICCAI 2016	15	T1, T1c, T2, FLAIR, DP	OUI	Test
ISBI 2015	21 (5 patients)	T1, T2, FLAIR, DP	OUI	Apprentissage
Lesjak <i>et al.</i>	30	T1, T1c, T2, FLAIR	OUI	Apprentissage
OASIS3 <i>et al.</i>	82	T1, T2, FLAIR	NON	Apprentissage
BraTS 2017 <i>et al.</i>	285	T1, T1c, T2, FLAIR	NON	Apprentissage

Table 4.1 – Les ensembles des données employés.

Prétraitement Les images ont été prétraitées pour être rendues les plus comparables possibles entre elles malgré des acquisitions différentes à plusieurs résolutions et interprétables par le réseau de neurones. Le prétraitement employé est basé sur celui du challenge de l'ISBI 2015 et est résumé dans la **Figure 4.1**. Après la première étape de correction de biais de champ par l'algorithme du N4 [44], les images sont alignées sur l'image FLAIR avec l'algorithme du FLIRT [92] sur lesquels les annotateurs se sont basés. Les histogrammes des images sont ensuite mis en correspondance par l'algorithme de Nyul *et al.* [48]. La masque du cerveau est ensuite calculé sur le T1 avec l'outil FSL BET [93] et appliqué à toutes les images avant alignement sur l'atlas du MNI [45] et ré-échantillonnage en $128 \times 128 \times 128$ voxels à $1.4 \times 1.7 \times 1.4 \text{ mm}^3$ pour limiter la taille des images tout en laissant les lésions visibles sur les images.

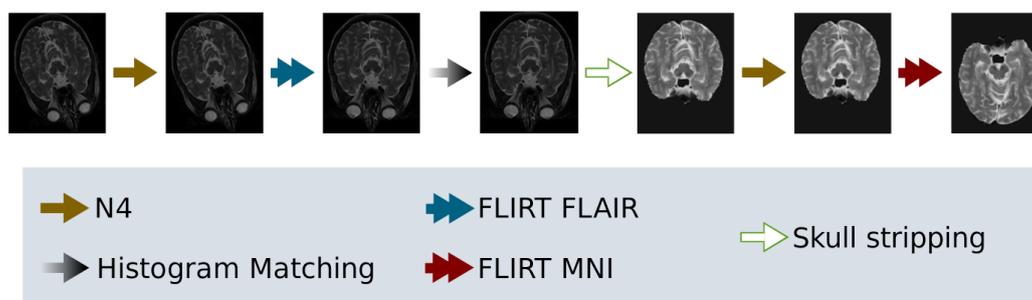


Figure 4.1 – Le prétraitement réalisé avec l'exemple d'une image T2.

Architecture

Pour ce travail, qui a commencé notre recherche sur le sujet, nous avons choisi une architecture déjà connue qui a fait ses preuves en segmentations, l'architecture proche du U-net de Isensee *et al.* [55] qui a été classée troisième pour la segmentation des tumeurs dans le challenge du BraTS 2017. C'est une architecture de type U-net avec

un encodeur et un décodeur avec des blocs résiduels et agrège plusieurs segmentations implémentant l'apprentissage collaboratif. Nous avons adapté l'architecture pour prendre en entrée les acquisitions 3D T1, T1c, T2, FLAIR et DP pour tirer parti de toutes les informations possibles pour prédire les segmentations. Pour l'entraînement sur la tâche prétexte non supervisée de régression des coordonnées des patchs, nous avons conservé l'encodeur de l'architecture en ajoutant des couches denses pour la régression des coordonnées des patchs comme décrit dans la **Figure 4.2**. Pour la tâche d'intérêt de segmentation des lésions de SEP nous avons conservé l'architecture entière avec encodeur et décodeur comme illustré sur la **Figure 4.2**.

Apprentissages

Les apprentissages ont été effectués sur des patchs de taille $32 \times 32 \times 32$ qui semble un bon compromis selon Roy *et al.* [49]. Nous n'avons pas employé d'augmentation artificielle des données pour diminuer le temps de calcul et avons jugé que les images provenant des différents ensembles présentent déjà une forme de diversité suffisante pour la généralisation de l'algorithme. Lorsqu'une séquence d'acquisition est manquante, nous l'avons remplacé par une image noire lors de l'apprentissage. Durant les apprentissages, certaines séquences d'entrée sont remplacées par des images vides aléatoirement pour forcer le réseau à baser son analyse sur toutes les entrées possibles et pour augmenter sa robustesse en cas de séquence manquante. C'est le principe du *sequence dropout* de Feng *et al.* [43]. Pour chaque lot d'apprentissages, 12 ensembles de patchs contenant au moins 75% de cerveau d'un même patient sont sélectionnés au hasard. Une époque est constituée d'un lot pour chaque patient d'apprentissage. Un tiers des examens d'apprentissages sont aléatoirement assignés dans l'ensemble de validation toutes les 60 époques et l'apprentissage est stoppé après 50 époques sans amélioration. Nous avons utilisé l'optimiseur Adam [79] avec un taux d'apprentissage fixé à 0.0001.

Les différentes étapes de la méthode sont résumées dans la **Figure 4.2**. L'apprentissage prétexte de régression des coordonnées des patchs est réalisé dans un premier temps, puis l'apprentissage supervisé de segmentation des lésions de SEP est effectué dans un second temps.

Tâche prétexte La tâche prétexte est apprise sur tous les sous-ensembles d'apprentissage détaillés dans le **Tableau 4.1**. Cependant, pour chaque époque, un ratio de 50% est respecté entre les images de patients atteints de SEP et les autres pour éviter de surapprendre sur les données provenant du BraTS et de l'OASIS3. La tâche non supervisée consiste à prédire la coordonnée du centre du patch extrait dans les axes x , y et z et le coût employé est l'erreur quadratique moyenne.

Tâche d'intérêt La tâche de segmentation des lésions de SEP est réalisé uniquement sur les ensembles de données de SEP avec le coût de Dice.

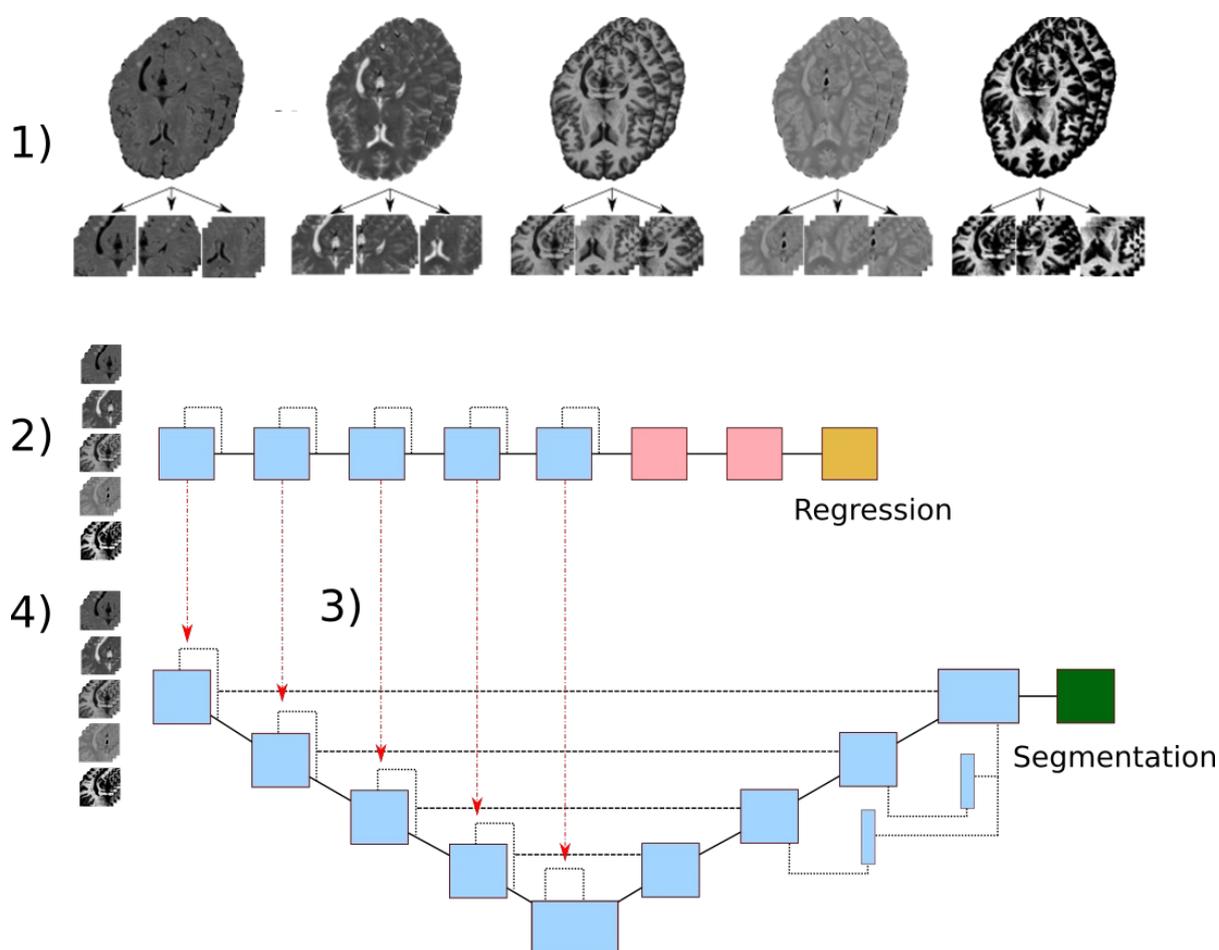


Figure 4.2 – Étapes d'apprentissage : 1) Les patches 3D de plusieurs séquences sont extraits à partir des volumes entiers. 2) Les patches sont envoyés au premier réseau neuronal pour l'apprentissage de la tâche de localisation. 3) Les poids des couches sont transférés au CNN de segmentation. 4) Apprentissage de la tâche de segmentation.

Expériences

Nous avons voulu évaluer l'apport de l'autosupervision et la profondeur optimale de réapprentissage pour notre tâche d'apprentissage des lésions de SEP. Pour ce faire, nous avons effectué les six expériences suivantes :

- VAN entraînement sur la tâche de segmentation avec initialisation de l'encodeur aléatoire,
- 0CF entraînement sur la tâche de segmentation avec l'encodeur préentraîné sur la tâche prétexte,
- 1CF entraînement sur la tâche de segmentation avec l'encodeur préentraîné avec la première couche gelée sur la tâche prétexte,
- 2CF entraînement sur la tâche de segmentation avec l'encodeur préentraîné avec les deux premières couches gelées sur la tâche prétexte,
- 3CF entraînement sur la tâche de segmentation avec l'encodeur préentraîné avec les trois premières couches gelées sur la tâche prétexte,
- 6CF entraînement sur la tâche de segmentation avec l'encodeur préentraîné avec les six premières couches gelées sur la tâche prétexte.

4.1.3 Résultat et analyse

Les résultats ont été évalués sur la tâche de segmentation des lésions de SEP sur les images de test (dans le **Tableau 4.1**) qui n'ont jamais été employées dans aucune des étapes d'apprentissage. Les prédictions sont le résultat de la moyenne spatiale de l'assemblage de la prédiction de 150 patchs extraits pour chaque examen. La prédiction subit ensuite un seuillage à 0,6 qui donne en pratique, ici, le meilleur score de Dice.

Pour mesurer la qualité de segmentation nous avons utilisé le score de Dice, la sensibilité et la précision calculés au niveau du voxel et détaillés dans l'**Equation 3.2**.

Expérience	Dice	Sensibilité	Précision
VAN	66,94%	60,63%	78,84%
0CF	67,63%	61,47%	79,30%
1CF	66,89%	59,79%	80,31%
2CF	64,44%	57,50%	77,53%
3CF	62,69%	55,86%	75,06%
6CF	57,93%	49,60%	75,03%

Table 4.2 – Moyennes des scores sur l'ensemble de test.

Les résultats moyens sont présentés dans le **Tableau 4.2** et le diagramme en boîte montrant la distribution des résultats dans l'ensemble de test est présenté dans

la **Figure 4.3**. L'évaluation a été réalisée sur l'ensemble de données du MICCAI 2016 acquis avec un scanner IRM et des protocoles différents des bases d'entraînement. La segmentation de référence pour l'évaluation est le consensus de la segmentation entre différents radiologues. La bonne qualité de la segmentation pour toutes les versions indique que l'architecture du CNN et la méthode d'apprentissage sont pertinents pour notre tâche de segmentation. Le meilleur score de Dice et la meilleure sensibilité sont obtenus avec la méthode 0CF et la meilleure précision est atteinte avec la méthode 1CF.

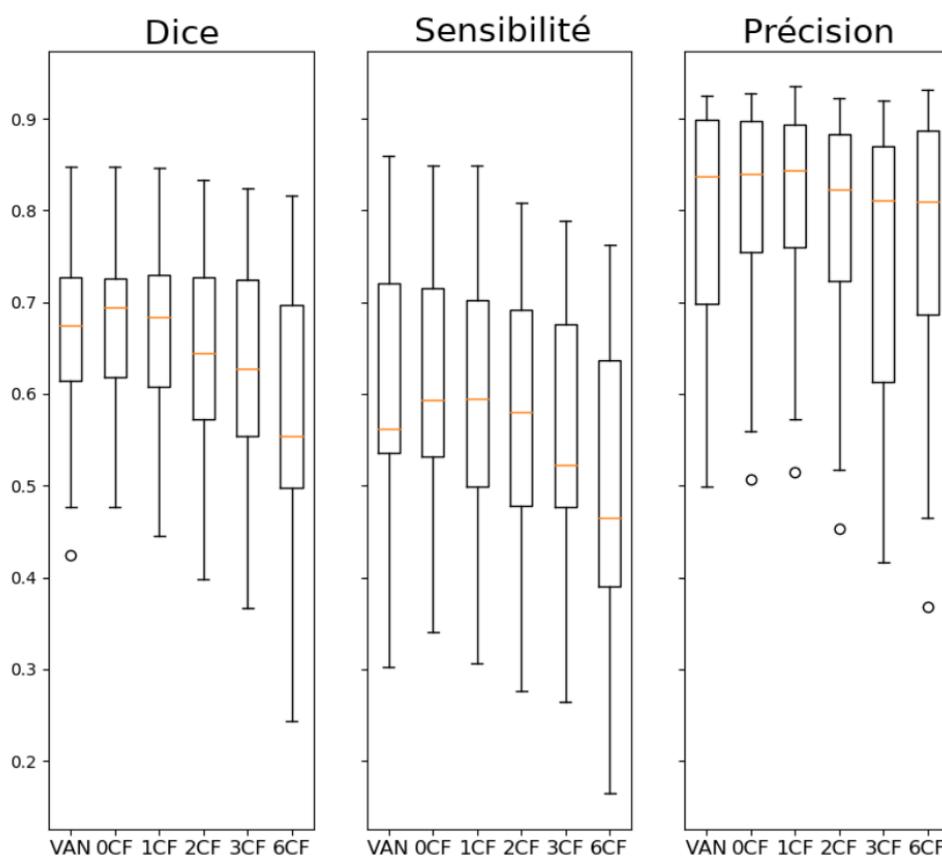


Figure 4.3 – Diagramme en boîte à moustache des scores obtenus sur l'ensemble de test. La médiane est représentée par les lignes orange, l'intervalle interquartile par la boîte, les moustaches symbolisent l'intervalle de répartition des données et les points sont les valeurs aberrantes.

La **Figure 4.3** montre que l'augmentation du nombre de couches gelées diminue le score de Dice et la sensibilité et augmente la dispersion des résultats. La précision est également diminuée avec le nombre de couches gelées, mais dans une moindre mesure.

Dans la **Figure 4.3** et **Tableau 4.1**, l'expérience OCF, avec autosupervision et réapprentissage total du réseau est légèrement meilleure que l'entraînement VAN initialisée aléatoirement. Le gel des premières couches entraîne rapidement une baisse du score de Dice, de sensibilité et précision. Seule la version OCF a surpassé pour chaque mesure la méthode VAN. Notre hypothèse initiale était que la tâche prétexte, non supervisée et la tâche d'intérêt supervisée sont assez proches pour partager les premières couches convolutives, comme cela est souvent le cas en vision par ordinateur. Les résultats indiquent cependant que ce n'est pas le cas et que les toutes premières couches doivent être réapprennent pour une meilleure segmentation. Les deux tâches ne nécessitent donc pas la même extraction de caractéristiques dès les premières phases d'analyses.

Le diagramme en boîte de la **Figure 4.3** montre que le gel des couches convolutives a diminué le score de Dice et la sensibilité dès le début. La perte de précision n'apparaît qu'après le gel des deux premières couches. Ces résultats suggèrent que le gel des premières couches convolutives affecterait principalement la sensibilité. Ceci est donc en faveur de l'hypothèse selon laquelle les premières couches du RNC capturent une grande partie des informations utiles de l'image dans les toutes premières couches. Cela suggère également que la sélection et la collecte de la plupart des caractéristiques discriminantes sont réalisées progressivement après la première couche.

La **Figure 4.4** illustre une visualisation de la segmentation prédite avec notre technique. Nous pouvons voir sur cette image que l'augmentation du nombre de couches gelées diminue la taille de la zone segmentée et augmente la taille de la zone correspondant au faux positif dans l'ellipse au bord frontal gauche du ventricule. Cela peut révéler un sous-apprentissage, car la zone périventriculaire contient généralement des lésions.

Nos résultats confirment l'idée que la régression des coordonnées et la segmentation des lésions sont des tâches vraiment différentes, même si elles sont effectuées avec des images similaires. Une des raisons peut être que le nombre d'échantillons d'apprentissage est trop important pour observer l'avantage du préapprentissage. Les observations concordent avec les constatations de Yosinski *et al.* [94] qui décrivent que la spécialisation du réseau de neurones augmente avec la profondeur et que l'apprentissage par transfert apporte la plupart du temps des améliorations même après réapprentissage dès la première couche.

Nous proposons une nouvelle façon de mettre en œuvre et d'employer les méthodes d'autosupervision en imagerie. Nos conclusions soutiennent l'idée que l'adoption de l'autosupervision pour des tâches d'analyse d'image médicale ne sont pas aisées et nécessitent des recherches plus approfondies pour définir les limites de la similarité et de la hiérarchie des tâches de vision. Cependant, nous pensons que les techniques d'autosupervision pourraient être davantage exploitées et que le fait de trouver une

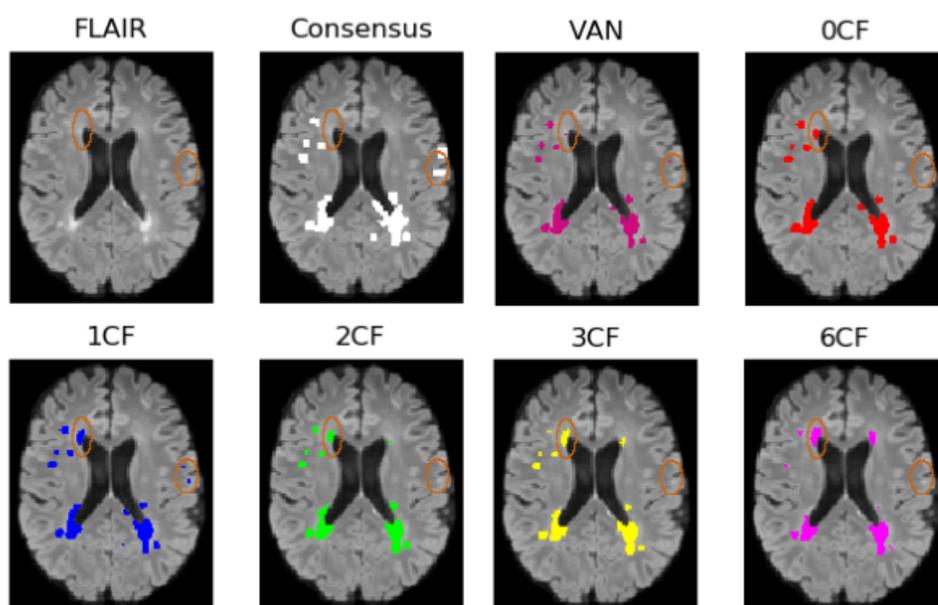


Figure 4.4 – Visualisation des différentes segmentations sur une coupe. Les principales zones de différence sont entourées par des ellipses orange.

bonne première tâche non supervisée à apprendre peut conduire à des améliorations dans le domaine de l'imagerie médicale.

Ce travail a été le premier travail effectué sur le sujet. Il a permis de poser un certain nombre d'hypothèses et d'orienter par la suite notre recherche. Puisque l'apprentissage n'a bénéficié que très peu de l'autosupervision, nous avons réalisé que la tâche de segmentation des lésions de SEP n'était peut-être pas si complexe que nous le pensions. Ce travail nous a donc amenés à déterminer si nous ne pouvions pas diminuer la complexité du modèle et l'apport en données sur une tâche *a posteriori* moins ardue que prévu.

4.2 Nombre de patients nécessaire

Nous avons choisi de mettre au point une architecture performante dans un premier temps, puis d'étudier le nombre de patients nécessaire dans un second temps dans l'article [29]. En effet, plus l'architecture est complexe, plus il faut d'exemples d'apprentissage en théorie pour apprendre le modèle généralisable. Le travail présenté a recours à notre modèle léger et performant, le MPU-next développé dans la **Section 3.3**.

L'expérience menée ici a été conçue pour estimer le nombre de patients nécessaire pour apprendre notre modèle du MPU-next pour la segmentation des lésions de SEP. Nous avons donc entraîné notre modèle séparément et plusieurs fois avec un nombre différent d'examens et avons comparé les performances moyennes. Nous avons choisi de changer l'ensemble de données et de mettre à profit les images de Lesjak *et al.* [89], car c'est l'ensemble publique de segmentation de lésions de SEP contenant le plus d'examens.

4.2.1 Matériels et méthodes

Données

Ensemble de données Nous avons choisi de changer l'ensemble de données et employer les images de Lesjak *et al.* [89], car c'est l'ensemble publique de segmentation de lésions de SEP contenant le plus d'examens avec 30 examens segmentés provenant de patients différents. Les images sont acquises de façon sagittale avec une résolution de $0,8 \times 0,47 \times 0,47 \text{ mm}^3$. Nous avons choisi de retirer de l'étude un patient qui présentait une lésion très volumineuse et assez atypique pour de la SEP.

Prétraitement Les images sont livrées prétraitées avec le masque du cerveau calculé sur le T1 avec l'outil Robex [95] appliqué sur le FLAIR et une correction d'inhomogénéités de biais de champ N4 [44]. Nous avons uniquement conservé le FLAIR et rééchantillonné les images à une résolution de $0,8 \times 0,8 \times 0,8 \text{ mm}^3$ pour permettre aux convolutions de partager la même importance dans les trois dimensions spatiales. Pour rendre les intensités comparables nous avons mis en correspondance les histogrammes [48] et centré réduit les valeurs d'intensité des images.

Apprentissage

L'apprentissage est effectué avec des lots de 644 patches de taille $32 \times 32 \times 32$ extraits aléatoirement dans le cerveau des différents examens. Une époque est constituée de 104 400 patches et l'entraînement est arrêté après 30 époques qui sont suffisantes pour la convergence des modèles. L'optimisation du modèle utilise Adam [79] avec le taux d'apprentissage fixé à 0.004 avec le coût de Dice.

Pour déterminer le nombre de patients nécessaire, nous avons réalisé 5 entraînements différents avec 23 patients d'apprentissage et 6 patients de test sélectionnés aléatoirement à chaque fois pour limiter l'influence des éventuels cas particuliers de

l'ensemble de données.

Test

Les résultats sont calculés sur les segmentations résultantes de la fusion moyenne des prédictions du modèle effectuées sur les patches extraits avec un recouvrement de 8 voxels dans chaque direction. Les segmentations sont binarisées au seuil de 0,6 à l'instar du travail de la **Section 4.1**.

4.2.2 Résultat et analyse

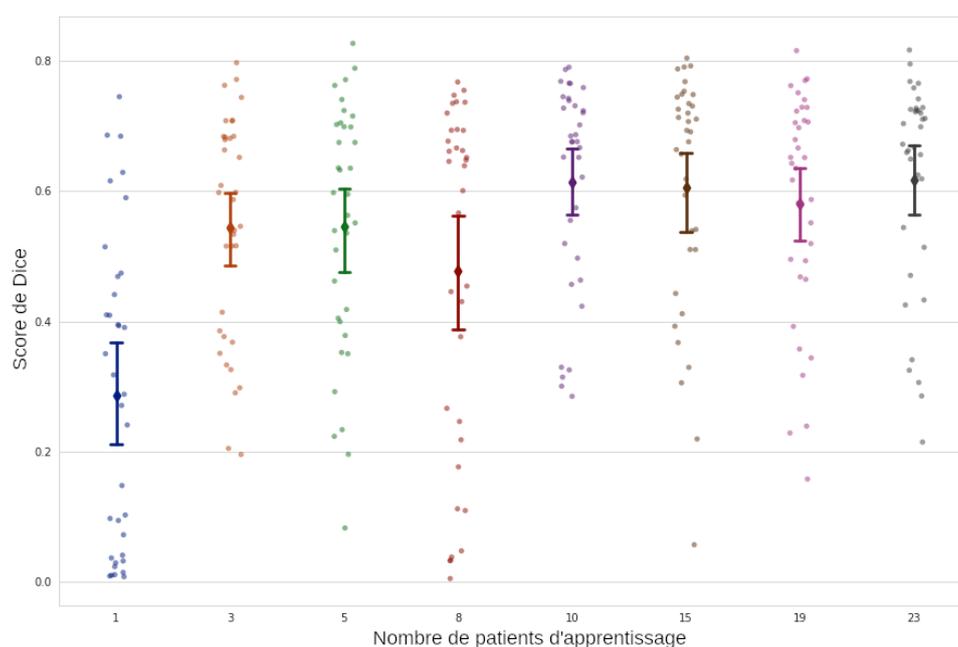


Figure 4.5 – Le score de Dice en fonction du nombre de patients d'apprentissage. Chaque point représente un patient testé pour chacune des 5 répétitions de l'expérience. Les moustaches représentent l'écart type centré autour de la moyenne.

Dans la **Figure 4.5**, nous pouvons voir qu'apprendre d'un seul patient n'est pas une bonne stratégie puisque le score de Dice moyen est faible (0,3), mais est presque doublé avec 3 patients apprenants. Nous n'avons pas vu de différence significative entre l'apprentissage avec 3 ou 5 patients. Étonnamment, apprendre avec 8 patients réduit les performances par rapport à 5 patients avec un écart type plus large de presque 0,2. De 10 à 23 patients d'apprentissage, les scores de Dice sont proches de 0,6 avec un écart-type proche de 0,1.

Dans nos conditions expérimentales, apprendre le MPU-next avec seulement 10 patients équivaut à apprendre avec 23 patients avec un score de Dice proche de 0,6. L'apprentissage avec 3 patients donne un score de Dice moyen inférieur, mais tout de même supérieur à 0,5. Le fait que l'apprentissage avec 8 patients soit pire qu'avec 5 patients en moyenne est certainement lié au choix aléatoire des patients qui peut avoir été une mauvaise combinaison dans certains ensembles d'entraînements ou de tests.

Nos résultats montrent qu'avec un faible nombre de paramètres le MPU-next est capable d'apprendre la segmentation des lésions de SEP de façon performante avec seulement 10 patients d'apprentissage. Il est même possible de baisser le nombre de patients d'apprentissage à 3 avec une légère perte de performances. Ce travail est un travail préliminaire pour donner un cadre et essayer de diminuer ce nombre avec un apprentissage faiblement supervisé.

4.3 Apprentissage semi-supervisé de la segmentation des lésions

Le travail précédent présenté dans la **Section 4.2** nous a montré que le MPU-next réalise des segmentations performantes à partir de 10 patients d'apprentissage. Nous proposons, ici, une méthode pour diminuer encore l'apport en données en faisant usage de l'apprentissage semi-supervisé qui nécessite en théorie moins de données annotées qu'une méthode supervisée.

4.3.1 Introduction

L'apprentissage automatique distingue les tâches supervisées et non-supervisées. L'apprentissage supervisé consiste en ensemble de données x ayant une valeur y correspondante. Le but est, alors, de construire un classifieur ou un régresseur qui estime la sortie y pour une entrée x qui n'a jamais été vue. Pour l'apprentissage non supervisé, il n'y a pas de sortie y fournie. À la place, le modèle doit déduire une structure sous-jacente de l'entrée x . Par exemple, en regroupement non supervisé, le but est de rassembler les entrées x en différents groupes par similarité [96].

Les méthodes d'apprentissage semi-supervisé ont pour but de rassembler l'apprentissage supervisé et non supervisé. Généralement, les méthodes d'apprentissage semi-supervisé ont pour but d'améliorer les performances d'un des deux apprentissages (supervisé ou non) en mettant à profit l'information généralement associée à l'autre. Par exemple, pour un problème de classification, des données non annotées peut être ajoutées pour aider l'apprentissage [96].

L'apprentissage semi-supervisé est un type d'apprentissage pertinent pour l'analyse d'analyse médicale puisqu'il a le potentiel d'améliorer les apprentissages sans avoir besoin d'annotations supplémentaires et donc de réduire la dépendance en une tâche d'annotation chronophage nécessitant un radiologue.

L'apprentissage semi-supervisé a été employé en imagerie médicale avec succès avec différentes méthodes. Par exemple, Chen *et al.* dans [97] donnent une méthode de semi-supervision multitâche. Ils adoptent une architecture avec un encodeur et deux décodeurs : un encodeur pour la segmentation des tumeurs cérébrales des images IRM (supervisé) et l'autre encodeur pour la reconstruction du cerveau et de la tumeur (non supervisé). Ils montrent qu'ils améliorent leurs performances apprenant avec des images annotées (sur les deux encodeurs) et avec des images non annotées (sur l'encodeur non supervisé). Baur *et al.* [98] présentent une méthode de segmentation des lésions de SEP semi-supervisée en apprenant conjointement l'apprentissage de la segmentation des lésions sur les données annotées et la minimisation de la différence entre deux entrées similaires dans l'espace latent. Bai *et al.* [99] améliorent légèrement leur méthode de segmentation cardiaque en apprenant à partir d'exemple annotées mais aussi d'exemples que leur modèle annoté lui-même au cours de l'apprentissage. Nie *et al.* [100], quant à eux, apprennent un modèle générant la segmentation de la prostate d'un côté avec une carte de confiance de l'autre. Ils peuvent, ainsi, durant l'entraînement, entraîner leur réseau de neurones sur des images annotées pour apprendre la segmentation avec un indice de confiance, prédire la segmentation avec la carte de confiance sur les images non annotées et apprendre sur les images non annotées à partir de la segmentation obtenue avec un grand indice de confiance. Nous nous sommes inspirés de Nie *et al.* [100] et Bai *et al.* [99] pour concevoir une nouvelle méthode de segmentation semi-supervisée pour la segmentation des lésions de SEP. Nous avons adapté la méthode de Nie *et al.* pour éliminer l'apprentissage d'un réseau générant la carte de confiance pour conserver uniquement un réseau de segmentation. Nous proposons de générer la carte de confiance en utilisant le *Monte-Carlo Dropout* [101] et en incrémentant peu à peu l'importance donnée à l'apprentissage réalisé sur les images sans annotations.

Avant de mener notre étude sur l'apprentissage semi-supervisé, nous avons voulu résoudre le problème que nous avons rencontré dans la **Section 4.2.2** concernant le tirage aléatoire des patients d'entraînement. En effet, dans la **Figure 4.5**, l'apprentissage avec 8 patients est en moyenne moins bon que l'apprentissage avec 5 patients, ce qui n'est pas cohérent. Notre hypothèse soulève la présence de meilleurs examens que d'autres pour apprendre la segmentation des lésions de SEP, ainsi, le tirage aléatoire de l'ensemble d'apprentissage est plus ou moins représentatif et

adapté pour l'apprentissage de la tâche d'intérêt. Nous avons alors mis au point un protocole pour mettre en évidence les dissimilarités qui existent entre les différents exemples et leurs valeurs d'apprentissage.

4.3.2 Évaluation de l'ensemble de données

Pour montrer la représentativité des différents examens annotés, nous avons entraîné le même modèle sur chaque examen séparément et évalué la performance du modèle appris sur le reste des examens. Ainsi, il est possible de représenter la qualité d'exemple d'un examen en particulier.

Protocole

Données Nous avons employé l'ensemble de données de Lesjak *et al.* [89] qui contient 30 examens de patients différents. L'ensemble d'images a été prétraité de la même façon que dans la **Section** 4.2.1.

Apprentissage En suivant le travail déjà effectué dans la **Section** 4.2.1, nous avons entraîné le MPU-next (voir **Section** 3.3) avec des patches de taille $32 \times 32 \times 32$ extraits aléatoirement dans le cerveau. Une époque est constituée de 25 600 patches extraits de l'examen d'apprentissage et l'entraînement est arrêté après 30 époques. L'optimisation est menée à bien grâce à Adam [79] avec un taux d'apprentissage à 0.004 et la fonction de coût utilisée est le coût de Dice.

Pour tous les apprentissages, c'est le même modèle, dont les poids ont été initialisés de façon uniforme [21], qui a été entraîné. Chaque apprentissage ne dispose uniquement que d'un examen comme source de supervision et les performances sont évaluées sur tous les autres examens de l'ensemble de données (29). Nous avons effectué un apprentissage par examen et avons donc en tout effectué 30 apprentissages différents.

Test Les cartes de segmentations sont générées de la même façon que dans la **Section** 4.2.1 avec un seuil de de binarisation à 0.5 pour rester sans *a priori*.

Métriques Pour l'évaluation des segmentations deux métriques ont été sélectionnées :

- le score de Dice pour l'évaluation de la segmentation (au niveau du voxel),
- le score F1 pour l'évaluation de la détection de lésions.

Le score F1 est analogue au score de Dice défini dans l'**Equation 3.2** mais au niveau lésionnel. En effet, le F1 score s'écrit :

$$F1 = 2 \cdot \frac{Precision_L \cdot Sensibilite_L}{Precision_L + Sensibilite_L}, \quad (4.1)$$

où $Precision_L$ est la proportion de lésions détectées sur le nombre de lésions prédites et $Sensibilite_L$ est le nombre de lésions détectées sur le nombre de lésions. Une lésion est dite détectée s'il y a recouvrement entre la lésion prédite et la vraie lésion. Le score F1 est calculé comme pour le challenge de MICCAI 2016 de segmentation des lésions de SEP [27].

Résultats et analyse

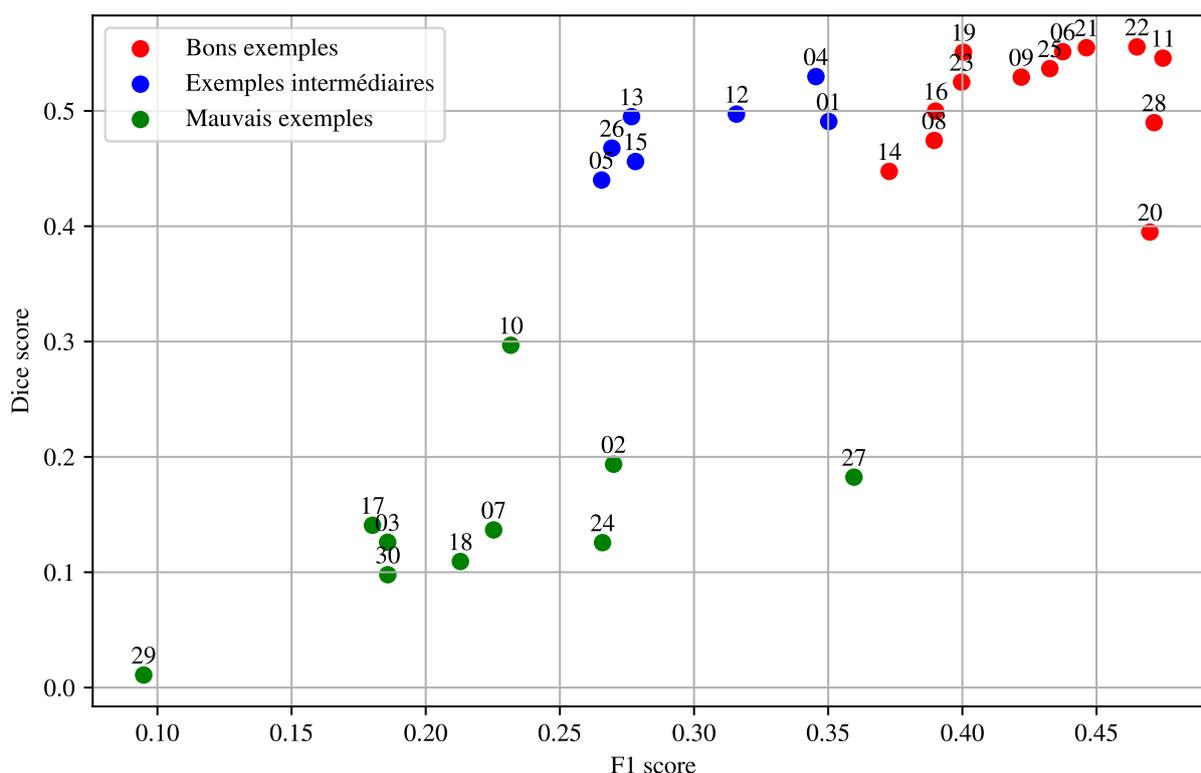


Figure 4.6 – Les scores de Dice et F1 moyens avec le même modèle appris séparément sur chaque examen et testé sur les examens restants. Les nombres sont les identifiants des patients. Nous avons découpé le nuage de points en trois classes pour classifier les exemples.

Lorsque l'on apprend le même MPU-next sur un examen à la fois et évaluons les performances de ce modèle sur le reste des examens, nous obtenons la **Figure 4.6**. La **Figure 4.6** met en évidence l'existence de différents profils de patients qui mènent à l'apprentissage d'un modèle plus ou moins performant. Nous pouvons remarquer qu'il existe trois profils différents de patients :

- les mauvais patients qui ont un score de Dice moyen assez faible inférieur à 0,3,
- les patients intermédiaires avec un score de Dice moyen supérieur à 0,3 et un score F1 moyen inférieur à 0,35
- les bons patients avec un score de Dice moyen supérieur à 0,3 et un score F1 moyen supérieur à 0,35.

Les mauvais patients seuls entraînent des modèles qui ont de mauvaises segmentations et détection des lésions. Les entraînements avec des patients intermédiaires donnent une bonne segmentation mais une moins bonne détection des lésions. Les patients intermédiaires apprennent donc des modèles qui ont tendance à segmenter les grandes lésions (beaucoup de voxels) aux dépens des petites. Les bons patients quant à eux apprennent des modèles qui réalisent des segmentations performantes avec une bonne détection des lésions. Ce sont donc des modèles qui se focalisent moins sur les grandes lésions que les modèles appris avec des patients intermédiaires.

Nous avons analysé les profils lésionnels de chaque patient et avons pu associer certaines caractéristiques à chaque classe. Les mauvais patients d'apprentissage sont des patients qui ont généralement très peu de lésions et des lésions très petites. Les patients intermédiaires sont des patients avec de grosses et/ou nombreuses lésions périventriculaires avec ou sans présence d'autres lésions. Les bons patients ont en général beaucoup de lésions assez peu volumiques et réparties à plusieurs localisations dans la substance blanche, en juxtacortical, en périventriculaire, dans le cervelet et dans le tronc cérébral. Les lésions périventriculaires sont globalement assez petites ou alors en présence d'autres grosses lésions dans les autres localisations.

Les différentes classes de patients ont des profils lésionnels cohérents sachant que le coût de Dice (réalisé au niveau du voxel) a été utilisé pour l'entraînement des modèles. En effet, avec peu de lésions, il y a une faible source de supervision et donc un modèle appris peu performant. Avec une surreprésentation des voxels périventriculaires, les modèles apprennent surtout à segmenter ces grandes lésions pour optimiser le coût de Dice. Si le profil lésionnel du patient est assez équilibré avec des lésions assez petites à différentes localisations, le modèle apprend à détecter toutes les lésions, puisque les lésions périventriculaires ne sont pas surreprésentées au niveau voxelique.

Cette étude nous a permis de choisir un ensemble de test équilibré pour les tests de la méthode semi-supervisé et pour orienter le choix d'apport de supervision.

4.3.3 Conception d'une méthode semi-supervisée pour la segmentation des lésions de SEP

Dans la **Section** 4.2.2, nous avons mis en évidence que 10 examens d'apprentissage suffisaient à entraîner notre MPU-next. Plus tôt dans la **Section** 4.3.2, nous avons montré qu'il existait des bons et de mauvais exemples d'apprentissage. L'entraînement avec un bon patient peut apprendre des modèles qui ont des performances de segmentations acceptables avec plus de 0,5 de score de Dice et un score F1 supérieur à 0,45. Nous cherchons maintenant à améliorer les performances de segmentation en limitant la quantité de supervision en concevant une méthode d'apprentissage semi-supervisée.

En nous basant sur le MPU-next et le travail de Bai *et al.* [99], nous proposons une approche semi-supervisé simple à mettre en place. Bai *et al.* apprennent un réseau de segmentation en même temps qu'un réseau de neurones prédisant la confiance de prédiction. Ils apprennent les deux réseaux sur des images annotées. Ils emploient, lors de l'entraînement, la prédiction de leur réseau de segmentation sur des images non annotées comme source de supervision lorsque leur réseau de confiance estime une forte confiance de prédiction. Cette méthode nécessite donc l'apprentissage simultané de deux réseaux de neurones. Notre méthode de semi-supervision estime la confiance de prédiction sans recours à un autre réseau de neurones avec le *Monte Carlo Dropout* [101].

Méthode

L'apprentissage est réalisé avec l'entrée X_s constituée de patchs d'image FLAIR avec le masque de segmentation P_s correspondant pour la partie supervisée. Sinon, l'ensemble non supervisé est constitué uniquement de l'entrée X_u constituée de patchs d'image FLAIR.

Coût supervisé Le coût supervisé est simplement le coût de Dice $Cost\ Dice(P^s, \hat{P})$ détaillé dans l'**Equation** 3.1 où \hat{P} est la prédiction du réseau.

Coût non supervisé La méthode proposée consiste à tirer parti des prédictions générées par le modèle comme source de supervision. Seulement, toutes les prédictions générées par le réseau de neurones n'ont pas le même degré de certitude. Inspirés par Nair *et al.* [101], nous avons mise à profit le *Monte Carlo dropout* pour obtenir une prédiction avec un niveau de confiance sur les données non annotées X^u sans recours à un autre réseau de neurones, donnant une méthode plus simple et souple que Bai *et al.* [99]. Le *Monte Carlo dropout* consiste à laisser le *dropout* activé lors du test et de répéter n fois la prédiction. Ainsi, n cartes de segmentations différentes sont générées avec une mise à zero aléatoire des cartes de caractéristiques. Il est ainsi possible d'obtenir une valeur d'incertitude de la prédiction en calculant la variance de ces n cartes de segmentations. Nous avons choisi d'employer l'écart-type de prédiction $std(\hat{P}^{u1}, \dots, \hat{P}^{un})$ comme mesure d'incertitude pour pondérer l'apprentissage non supervisé et la valeur moyenne de prédiction $moy(\hat{P}^{u1}, \dots, \hat{P}^{un})$ pour décider de l'inclusion du voxel comme source d'apprentissage en conservant le coût de Dice. L'apprentissage non supervisé est masqué en fonction de la valeur moyenne de prédiction calculée avec le *Monte Carlo Dropout* avec la fonction :

$$Mask^u(P) = \begin{cases} P & \text{si } moy(\hat{P}^{u1}, \dots, \hat{P}^{un}) \in [0; 0,5 - i] \cup [0,5 + i; 1] \\ 0 & \text{sinon} \end{cases}, \quad (4.2)$$

où i définit le seuil autour de 0,5 à partir duquel une prédiction moyenne est incluse dans l'apprentissage. La pondération par rapport à l'écart type calculée grâce au *Monte Carlo Dropout* s'écrit :

$$W^u = \begin{cases} std(\hat{P}^{u1}, \dots, \hat{P}^{un}) & \text{si } moy(\hat{P}^{u1}, \dots, \hat{P}^{un}) < 0,5 - i \\ 1 - std(\hat{P}^{u1}, \dots, \hat{P}^{un}) & \text{si } moy(\hat{P}^{u1}, \dots, \hat{P}^{un}) > 0,5 + i \\ 0 & \text{sinon} \end{cases}. \quad (4.3)$$

Le coût non supervisé sur le modèle du coût de Dice s'écrit alors :

$$Coût\ Dice^u(\hat{P}^{u1}, \dots, \hat{P}^{un}, \hat{P}) = -2 \frac{\sum_{b=1}^B \sum_{h=1}^H \sum_{w=1}^W \sum_{d=1}^D \sum_{s=1}^S Mask^u(W_{c,b,h,w,d,s}^u \hat{P}_{c,b,h,w,d,s})}{\sum_{b=1}^B \sum_{h=1}^H \sum_{w=1}^W \sum_{d=1}^D \sum_{s=1}^S Mask^u(W_{c,b,h,w,d,s}^u + \hat{P}_{c,b,h,w,d,s})}. \quad (4.4)$$

Coût total Le coût total est calculé par la somme pondérée du coût supervisé et le coût non supervisé. En début d'apprentissage, le modèle n'est pas encore assez mature pour donner des prédictions fiables. Pour limiter les effets de bords dus à

ce phénomène, la pondération de la tâche non supervisé augmente progressivement avec le nombre d'époques jusqu'à avoir autant d'importance que le coût supervisé. Le coût total semi-supervisé s'écrit :

$$Cost_{ss} = Cost\ Dice(P^s, \hat{P}) + Cost\ Dice^u(P^{\hat{u}1}, \dots, P^{\hat{u}n}, \hat{P}) \times \min(1, \log(episode) \times \lambda^u), \quad (4.5)$$

où l'hyperparamètre λ^u pondère le coût non supervisé.

Données Nous avons au recours au même ensemble de données et le même pré-traitement que dans la **Section 4.2.1**

Apprentissage Les apprentissages sont réalisés avec le MPU-next sur des patchs de taille $32 \times 32 \times 32$ par lot de 2048. Une époque est constituée de 98 304 patchs supervisés et le même nombre de patchs non supervisés pendant 40 époques. Tous les apprentissages sont initialisés uniformément avec les mêmes valeurs. Le *Monte Carlo Dropout* est effectué avec $n = 15$ prédictions et le seuillage autour de 0,5 est fixé avec $i = 0,15$ pour masquer les valeurs moyennes de prédictions entre 0,35 et 0,65. Le sous-ensemble de test contient 14 examens et a été équilibré pour contenir des proportions de chaque classe de patient (classifiées dans la **Section 4.3.2** dans la **Figure 4.6**) similaires à l'ensemble de données employé avec les patients 01, 06, 07, 10, 13, 16, 17, 18, 19, 20, 22, 26, 27, 28. L'optimiseur Adam [79] avec un taux d'apprentissage de 0.004 est utilisé.

Test et évaluation Le test et l'évaluation sont exécutés de la même façon que dans la **Section 4.3.2**.

Pour évaluer l'apport de notre méthode de supervision, nous avons lancés plusieurs tests avec des multiples examens différents comme source de supervision et de non-supervision en faisant varier la pondération de la tâche non supervisée λ^u .

Expériences et résultats

Pour débiter nos expériences, nous avons entraîné notre architecture sur tous les patients d'apprentissages pour avoir une référence avec laquelle comparer les résultats avec la semi-supervision.

Examen supervisé	Examen non supervisé	λ^u	Dice	F1
Tous	Aucun	0	0,5833	0,4615

Table 4.3 – Résultats de référence sans semi-supervision avec tous les patients d'apprentissage supervisés.

Sans supervision avec tous les patients Les résultats de référence sont présentés dans le **Tableau 4.3**.

Source de supervision Nous avons choisi de n'exploiter qu'un examen comme source de supervision et trois examens comme source non supervisée. Dans cette première expérience, nous avons évalué les performances du modèle appris sur trois patients supervisés différents appartenant aux classes de patients bons, intermédiaires et mauvais définis dans la **Section 4.3.2**.

Examen supervisé	Examen non supervisé	λ^u	Dice	F1
11	25,05,30	0,	0,5491	0,4855
		0,1	0,5883	0,4829
		0,15	0,5920	0,4848
15	25,05,30	0,	0,5037	0,3205
		0,1	0,4864	0,2768
		0,15	0,4786	0,2682
03	25,05,30	0,	0,1105	0,1961
		0,1	0,2506	0,1958
		0,15	0,2928	0,2172

Table 4.4 – Résultats moyens de semi-supervision en faisant varier l'examen de supervision et la pondération de la tâche semi-supervisée λ^u . Les bons patients sont en rouge, les intermédiaires en bleu et les mauvais patients vert pour respecter le code couleur de la **Figure 4.6**.

Dans le **Tableau 4.4**, sans semi-supervision ($\lambda^u = 0$) on vérifie avec le nouvel ensemble de données de test les tendances observées dans la **Section 4.3.2**.

Avec l'examen 11 (bon exemple) en supervision, nous augmentons le Dice avec la semi-supervision avec 0,5883 et 0,5920 de score de Dice pour $\lambda^u = 0,1$ et $\lambda^u = 0,15$ respectivement contre 0,5491 sans semi-supervision. Le score F1 est peu impacté dans cet exemple avec 0,4829 et 0,4848 pour $\lambda^u = 0,1$ et $\lambda^u = 0,15$ contre 0,4855 sans semi-supervision. Avec l'examen 11 comme source de supervision, le score de Dice augmente avec la pondération sur la tâche non supervisée et le score F1 reste stable. Après comparaison avec le **Tableau 4.3**, avec un seul patient sans semi-supervision le score F1 est amélioré avec 0,4855 contre 0,4615 lorsque le modèle est appris sur tous les patients, cependant le score de Dice est plus faible avec le patient 11 uniquement

avec 0,5491 contre 0,5833. Avec la semi-supervision, il y a maintient le score de Dice moyen en améliorant le score F1.

Avec l'examen 15 (exemple intermédiaire) en supervision, les scores de Dice et F1 diminuent progressivement avec la semi-supervision. Le score de Dice passe de 0,5037 sans semi-supervision à 0,4786 avec $\lambda^u = 0,15$ et le score F1 de 0,3205 à 0,2682. Les performances sont dans tous les cas inférieures aux performances avec l'entraînement supervisé sur tous les patients d'apprentissage du **Tableau 4.3**.

L'entraînement avec le patient 03 (mauvais exemple) comme source de supervision quant à lui bénéficie de l'apport de la semi-supervision en passant de 0,1105 de score de Dice à 0,2928 avec $\lambda^u = 0,15$ et de 0,1961 de score F1 à 0,2172. Les performances sont dans tous les cas inférieures aux performances avec l'entraînement supervisé sur tous les patients d'apprentissage du **Tableau 4.3**.

Les résultats du **Tableau 4.4** est cohérent avec l'analyse de la **Section 4.3.2** sur la classification des patients. L'augmentation du score de Dice sans changement du score F1 avec la semi-supervision en apprenant avec un bon patient indique que le modèle bénéficie d'un apport de segmentation avec la source non supervisée sans mieux détecter les lésions. Il est probable que la source de supervision soit suffisante pour obtenir une bonne détection des lésions et que la source non supervisée permette d'améliorer la segmentation des larges lésions pour augmenter le score de Dice. La semi-supervision baisse les performances si l'exemple d'apprentissage est intermédiaire. Notre hypothèse est que l'apprentissage avec l'examen intermédiaire, contenant essentiellement de grosses lésions périventriculaires, apprend surtout à segmenter les grandes lésions et n'est pas assez généraliste pour fournir des segmentations avec une valeur ajoutée pour les examens non supervisés et apprendre par lui-même. L'augmentation des performances avec la semi-supervision avec un mauvais exemple en supervision montre que le modèle appris avec de petites et rares lésions apprend avec les examens non supervisés à segmenter plus de lésions et donc s'améliore.

Pour étudier l'apport de la source de non-supervision, nous avons changé les exemples non supervisés tout en gardant le meilleur patient non supervisé, le patient 11.

Source non supervisée Pour l'expérience du **Tableau 4.5** nous avons conservé le patient 11 (bon exemple) comme source de supervision et changé la source non supervisée comme étant trois patients de chaque classe.

Dans le **Tableau 4.5**, l'apprentissage avec trois mauvais exemples en non supervisé entraîne une baisse de score de Dice de 0,5491 en complètement supervisé à 0,4348 avec $\lambda^u = 0,4348$ et une baisse de score F1 de 0,4855 à 0,3391.

L'apprentissage avec trois patients intermédiaires comme source non supervisée permet une amélioration du score de Dice jusqu'à 0,5806 et une légère baisse du score F1

Examen supervisé	Examen non supervisé	λ^u	Dice	F1
11	02,24,30	0,	0,5491	0,4855
		0,1	0,4620	0,3391
		0,15	0,4348	0,3108
	04,05,12	0,	0,5491	0,4855
		0,1	0,5790	0,4856
		0,15	0,5806	0,4768
	25,21,09	0,	0,5491	0,4855
		0,1	0,5748	0,4860
		0,15	0,5785	0,4817

Table 4.5 – Résultats moyens de semi-supervision en faisant varier les examens non supervisés et la pondération de la tâche semi-supervisée λ^u . Les bons patients sont en rouge, les intermédiaires en bleu et les mauvais patients vert pour respecter le code couleur de la Figure 4.6.

vers 0,4768.

Avec trois bons exemples non supervisés le modèle améliore sa segmentation des lésions avec un score de Dice à 0,5748 pour $\lambda^u = 0,1$ et 0,5785 pour $\lambda^u = 0,15$ et maintient globalement le score F1 moyen.

Les résultats du **Tableau 4.5** montrent que les mauvais exemples d'entraînement diminuent les performances d'apprentissage lorsqu'ils sont ajoutés comme source non supervisée. Les examens intermédiaires ou bons comme source de non-supervision améliorent tous deux le score de Dice avec assez peu d'effet sur le score F1. La source non supervisée permet dans ces deux cas une amélioration de la segmentation sans améliorer la détection des lésions.

Ce travail montre qu'avec uniquement un bon patient d'apprentissage seul, il est possible d'améliorer la détection des lésions par rapport à l'entraînement avec 16 patients avec une segmentation légèrement moins bonne. L'apprentissage semi-supervisé avec un bon patient segmenté et trois patients non segmentés (des trois classes, ou uniquement bons ou intermédiaires) permet d'améliorer la segmentation des lésions de SEP tout en maintenant une bonne détection. La meilleure segmentation des lésions avec un score de Dice moyen de 0,5920 est obtenue avec un bon patient comme source de supervision et un patient de chaque classe comme source non supervisée avec $\lambda^u = 0,15$. La meilleure détection des lésions avec un score F1 de 0,4860 est obtenu avec un bon patient supervisé et trois bons patients non supervisés.

4.4 Conclusion

Guidés par l'hypothèse que nous n'avons pas assez d'exemples d'apprentissage pour segmenter les lésions de SEP, nous avons mis au point une méthode autosupervisée pour l'apprentissage des lésions de SEP. L'autosupervision nous a permis d'augmenter légèrement nos performances de segmentation, mais pas autant qu'espéré.

Nous avons donc souhaité savoir si le nombre d'exemples d'apprentissage était vraiment une limitation pour notre tâche. Nous avons alors entraîné notre modèle léger, le MPU-next, avec plusieurs nombres de patients différents et avons constaté que 10 patients seulement suffisaient à réaliser un bon apprentissage. Cependant, apprendre avec 8 examens donne de moins bonnes performances qu'un apprentissage avec 5 patients en moyenne. Nous avons alors émis l'hypothèse que ce phénomène viendrait tout simplement de la source d'apprentissage et qu'il existerait de meilleurs exemples d'apprentissages que d'autres.

Nous avons ensuite conçu une méthode de segmentation semi-supervisée dans l'espoir de diminuer encore la dépendance aux données et principalement à l'annotation manuelle. Cependant, ayant un doute sur la qualité d'exemple de nos images, nous avons évalué la qualité d'exemple de chacun des examens de notre ensemble de données. Nous avons confirmé qu'il existe bel et bien plusieurs qualités d'examens pour apprendre la segmentation des lésions de SEP et avons pris en compte ce facteur pour le test et la validation de notre méthode semi-supervisée. L'évaluation de notre méthode montre que l'apprentissage avec un bon patient donne une meilleure détection des lésions qu'un apprentissage avec 16 patients avec une qualité de segmentation légèrement moindre. Avec notre méthode semi-supervisée, nous avons montré qu'avec un seul bon examen segmenté et trois patients non segmentés nous améliorons à la fois la détection des lésions de SEP mais aussi la segmentation de ces lésions.

Nous avons donc, maintenant, une idée claire d'architecture et des données nécessaires pour la segmentation et détection des lésions de SEP. Dans le prochain chapitre, plusieurs méthodes d'amélioration de la segmentation et détection des lésions de SEP sont mises en œuvre vers des modèles toujours plus performants.

5

Vers un meilleur apprentissage pour la segmentation des lésions de SEP

Sommaire

5.1	La taille des données en entrées pour la segmentation des lésions de SEP	122
5.1.1	Contexte	123
5.1.2	Matériels et méthodes	124
5.1.3	Résultats et analyse	125
5.2	Adaptation de la fonction de coût vers une meilleure détection des lésions	129
5.2.1	Analyse du problème	129
5.2.2	Matériels et méthodes	130
5.2.3	Résultats et analyse	134
5.3	Conclusion	141

DANS les deux chapitres précédents, nous avons travaillé sur la conception d’une architecture légère et adaptée pour la segmentation des lésions de SEP et sur l’étude du nombre d’images et d’annotations nécessaires pour apprendre un modèle performant. Ces deux facteurs permettent d’entraîner des modèles de segmentation des lésions de SEP de façon plus accessible. Cependant, pour apprendre un modèle performant, il convient aussi d’adapter les apprentissages au mieux à la tâche. Dans ce chapitre, nous nous focalisons sur l’amélioration des entraînements vers de meilleures performances.

Le travail présenté dans ce chapitre s’intéresse dans un premier temps à l’influence de la taille physique et voxelique des échantillons d’apprentissage qui est souvent mise sous silence, puis nous présentons notre travail sur la pondération de la fonction de coût pour améliorer la détection des lésions, et notamment les plus petites qui sont généralement oubliées par les modèles de segmentation appris malgré leur importance.

5.1 La taille des données en entrées pour la segmentation des lésions de SEP

Le choix des hyperparamètres dans l’apprentissage profond a un impact important sur la convergence de l’apprentissage et la performance du modèle final. Nous voulons, ici, étudier l’influence de la taille du patch et de la résolution de l’image pour l’apprentissage d’un modèle sur la segmentation des lésions de SEP. Cette étude donne un aperçu de la quantité de texture cérébrale environnante dont un réseau neuronal artificiel a besoin pour atteindre de bonnes performances et de la résolution la plus adaptée à la segmentation automatique des lésions dans notre contexte. L’objectif est de définir à quel point l’échantillonnage et la taille des patches ont une influence sur les performances de

segmentation des lésions. Nous avons testé empiriquement plusieurs tailles de patches et plusieurs échantillonnages d'images IRM et comparé les différents modèles appris.

5.1.1 Contexte

Pour la segmentation des lésions de SEP, les méthodes d'apprentissage profond de la littérature utilisent une entrée en 2, 2,5 ou 3 dimensions pour la segmentation des lésions de SEP [24], [39]. L'approche 2D a l'avantage de nécessiter moins de mémoire lors de l'apprentissage et de l'inférence, mais ne bénéficie pas ou peu de la nature 3D des images dans lesquelles la texture 3D environnante peut contenir des informations pertinentes. Récemment, l'approche 2.5D est apparue [39]. Elle consiste à prendre en entrée des tranches empilées pour produire en sortie la segmentation de la tranche centrale. L'approche 2,5D tire avantage de la nature volumétrique des images IRM avec un coût de mémoire supplémentaire négligeable. Nous avons choisi tout au long de notre travail l'approche 3D qui tire pleinement parti du travail sur un volume, mais qui nécessite trop de mémoire pour être entraînée sur des cartes graphiques avec des images complètes, en particulier avec une résolution d'acquisition inframillimétrique. La plupart des approches 3D travaillent avec des patches 3D d'images IRM plus ou moins larges avec ou sans rééchantillonnage.

Dans la littérature, la taille des patches 3D employés varie beaucoup avec des tailles de $15 \times 15 \times 15$ [35], $32 \times 32 \times 32$ [85], $96 \times 96 \times 96$ [83], par exemple. La sélection de la taille du patch 3D est rarement expliquée ou justifiée dans de nombreuses approches si ce n'est que par la taille en mémoire. L'étude de la taille des patches dans la performance de la segmentation de la SEP a été abordée par Roy *et al.* [49], mais uniquement sur des patches 2D avec au maximum une résolution de 35×35 pixels sans considération de la résolution de l'image IRM ou de la taille du lot. Nous proposons ici d'étudier l'impact de la taille du patch sur les performances du pour la segmentation de lésions de SEP avec le MPU-next dont le champ réceptif est bien plus large que la taille des patches compatibles avec la mémoire des processeurs graphiques dédiés les plus récents. La taille du lot et le nombre de patches extraits ont été soigneusement définis afin d'obtenir une quantité équivalente d'informations dans une époque et dans un lot, quelle que soit la taille du patch. L'impact du rééchantillonnage des images IRM, qui est une étape de prétraitement couramment appliquée, est étudié afin d'évaluer l'impact de la taille du patch physique et de trouver le meilleur couple résolution/taille du patch pour un entraînement efficace.

Nom	Résolution
<i>Iso min</i>	$0,47 \times 0,47 \times 0,47 \text{ mm}^3$
<i>Natif</i>	$0,47 \times 0,8 \times 0,8 \text{ mm}^3$
<i>Iso max</i>	$0,8 \times 0,8 \times 0,8 \text{ mm}^3$

Table 5.1 – Nom des rééchantillonnages effectués.

5.1.2 Matériels et méthodes

Le travail est effectué avec les images de l'ensemble de données de Lesjak *et al.* [89] avec le MPU-next de la Section 3.3 avec plusieurs tailles de patches et rééchantillonnages.

Données

Comme pour les travaux présentés dans les Sections 4.2 et 4.3, nous avons eu recours à l'ensemble de données de Lesjak *et al.* uniquement. Comme précédemment, nous avons travaillé avec les images FLAIR uniquement dont la résolution spatiale d'acquisition est à $0,8 \times 0,47 \times 0,47 \text{ mm}^3$.

L'ensemble de test est constitué de 14 examens choisis aléatoirement et les 16 autres examens sont utilisés comme exemples d'apprentissage.

Les images sont rééchantillonnées, rognées pour ne contenir que le cerveau puis les intensités sont mises en correspondance avec la méthode de Nyul *et al.* [48] avant d'être centrées réduites. Nous avons entraîné et évalué le MPU-next avec trois rééchantillonnages détaillés dans le Tableau 5.1 :

- *iso min* avec suréchantillonnage dans l'axe x avec une résolution isovolumétrique,
- *natif* sans rééchantillonnage,
- *iso max* avec sous-échantillonnage dans les axes y et z avec une résolution isovolumétrique.

Apprentissages

Les apprentissages ont été réalisés avec l'architecture du MPU-next (voir Section 3.3) avec les poids initialisés avec les mêmes valeurs de façon uniforme [21]. Nous avons recours à Adam [79] avec un taux d'apprentissage de 0,004 comme optimiseur.

Taille patch	Taille lot	Patch/pat.	N. lots	Vox/lot	Vox/époque
32^3	2048	6400	50	$6,72 \times 10^7$	$3,36 \times 10^9$
64^3	256	800	50	$6,72 \times 10^7$	$3,36 \times 10^9$
96^3	76	228	48	$6,72 \times 10^7$	$3,23 \times 10^9$
128^3	32	100	50	$6,72 \times 10^7$	$3,36 \times 10^9$

Table 5.2 – Choix de la taille du patch, de la taille du lot et du patch par patient afin d’avoir un nombre équivalent de voxels par lot et par époque.

Tailles de patches Nous avons testé quatre tailles de patch différentes : 32^3 , 64^3 , 96^3 et 128^3 . Le changement de la taille des patches induit un changement du nombre de voxels dans un lot si le nombre de patches est conservé. De la même façon, lors du changement de taille de patch, le nombre de voxels extraits change. Pour avoir des entraînements les plus comparables possibles, nous avons ajusté la taille des lots pour contenir un même nombre de voxels avec chaque taille de patch différente. Le nombre de patches extraits par patient, le nombre de lots dans une époque sont aussi ajustés pour avoir le même nombre de voxels par époque. Les différents paramétrages pour avoir des lots d’apprentissages comparables malgré la taille de patch sont indiqués dans le **Tableau 5.2**.

Les patches sont tous extraits aléatoirement avec le centre dans le masque du cerveau. Le coût de Dice de l’**Equation 3.1** est employé, car il est calculé au niveau du voxel et n’est pas moyenné pour chaque patch, ce qui rend les lots équivalents pour la fonction de coût.

5.1.3 Résultats et analyse

Tous les résultats présentés sont calculés sur l’ensemble de test. Chaque mesure affichée est moyennée sur les 14 examens de test. Au cours de l’inférence, la même taille de patch que pour l’entraînement est adoptée. Lors de l’inférence, les patches sont régulièrement échantillonnés sur les images avec un chevauchement d’un quart de la taille des patches dans chaque direction x , y et z . Après prédiction, la carte de prédiction résultante est rééchantillonnée à la résolution native et ensuite binarisée au seuil de 0,6. Nous avons évalué les performances des modèles après 30 époques, car nous avons observé une convergence à ce stade.

Nous utilisons comme métrique le score de Dice et F1 pour mesurer respectivement les performances de segmentation et de détection lésionnelle comme dans la **Section 4.3**.

Taille de patch et résolution

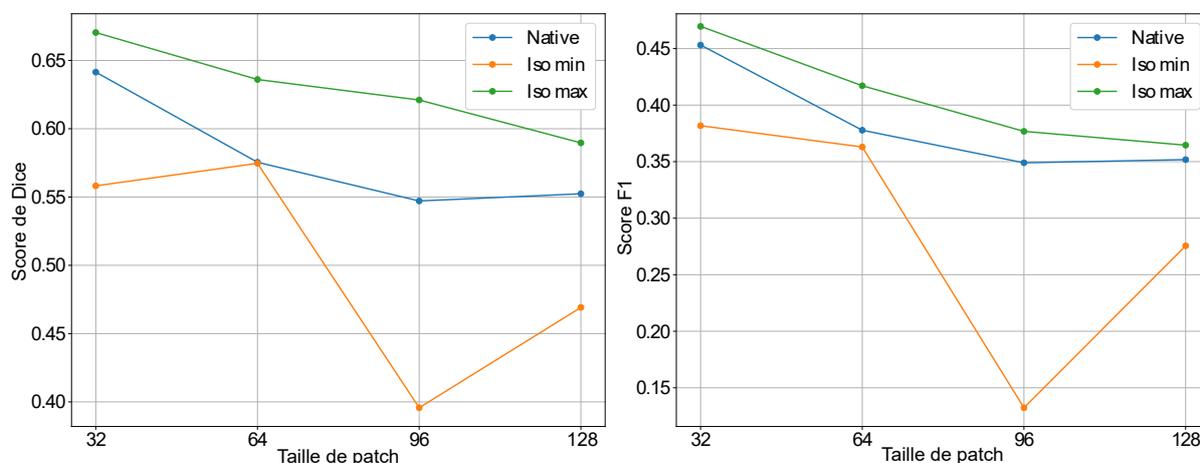


Figure 5.1 – Performances du modèle en termes de score de Dice (à gauche) et de score F1 (à droite) en fonction de la taille du patch avec différentes résolutions : *Natif* ($0,8 \times 0,47 \times 0,47 \text{ mm}^3$), *Iso min* ($0,47 \times 0,47 \times 0,47 \text{ mm}^3$) et *Iso max* ($0,8 \times 0,8 \times 0,8 \text{ mm}^3$).

Dans la **Figure 5.1**, nous pouvons voir que le suréchantillonnage à la plus petite résolution spatiale, *iso min*, baisse en moyenne le score de Dice et F1 par rapport à la résolution *natif*. Cela indique que le suréchantillonnage perturbe l'analyse et qu'il vaut mieux travailler avec une résolution non isométrique qu'une résolution isométrique suréchantillonnée dans notre cas.

La plus grande résolution spatiale, *iso max*, est meilleure en détection et segmentation pour chaque taille de patch dans la **Figure 5.1**. Le sous-échantillonnage isovolumétrique, *ie* des plus gros voxels cubiques, permet d'obtenir un modèle plus performant pour la segmentation et les détections des lésions de SEP. Il est intéressant de noter que malgré la perte d'information due à l'interpolation, de meilleures performances sont obtenues par rapport à la résolution d'acquisition.

La plus petite taille de patch évaluée dans notre protocole dans la **Figure 5.1**, $32 \times 32 \times 32$, est celle avec laquelle nous avons les meilleurs scores de segmentation et de détection puis une baisse de performance est observée avec l'augmentation de la taille des patchs pour les deux meilleures résolutions. Pour la résolution *iso min*, les performances varient avec un minimum de score de Dice et F1 pour une taille de patch de $96 \times 96 \times 96$ à respectivement 0,4 et 0,13 et un maximum de score de Dice à 0,57 avec une taille de patch de $64 \times 64 \times 64$ et un maximum de score F1 à 0,38 avec une taille de patch de $32 \times 32 \times 32$.

Les meilleurs scores de Dice à 0,672 et F1 à 0,471 sont atteints avec la résolution *iso max* et une taille de patch de $32 \times 32 \times 32$. Pour synthétiser, la meilleure segmentation et détection des lésions est obtenue avec la plus grande taille de voxel et avec la plus

petite taille de patch, le meilleur apprentissage est donc obtenu en compressant les voxels dans des petits patches. Nous avons identifié plusieurs raisons à ce phénomène :

- les lésions sont très rarement de taille inframillimétrique, une résolution trop fine apporte plus d'informations que nécessaire à l'analyse et la rend plus complexe,
- l'analyse d'un petit patch est plus locale et donc plus concentrée localement sans éventuelle perturbation des textures éloignées présente dans les plus grands patches,
- l'entraînement avec de plus gros patches contient plus de voxels en dehors du cerveau avec notre méthode d'extraction, ce qui peut mener à entraîner avec un plus grand déséquilibre de classe au niveau voxelique,
- l'entraînement avec de plus grands patches est effectué avec moins d'échantillon par lot (voir **Tableau 5.2**) et donc avec moins de diversité au sein d'un lot, pouvant perturber la généralisation d'apprentissage et la convergence du modèle par extension.

Le comportement des courbes de performance avec la résolution *iso min* de la **Figure 5.1** peut s'expliquer par une instabilité provoquée par le rééchantillonnage avec une information spatiale plus étalée, plus difficile à traiter qui rendrait l'apprentissage plus complexe et moins robuste, amenant à un comportement non reproductible avec des apprentissages aux performances instables.

La **Figure 5.1** montre que les modèles les mieux évalués ont un score de Dice acceptable vers 0,65, mais un score F1 inférieur à 0,5, ce qui signifie qu'il existe un ratio important de lésions non détectées et/ou faussement détectées. Comme la sensibilité et la précision sont liées à la binarisation de la prédiction, nous avons décidé d'évaluer l'influence du seuil sur les performances de prédiction, la sensibilité et la précision par lésion. Nous avons évalué le seuil uniquement sur les modèles les plus performants, c'est-à-dire avec les résolutions *natif* et *iso max* avec une taille de patch de $32 \times 32 \times 32$ pour construire les résultats du **Tableau 5.3**.

Dans le **Tableau 5.3**, nous pouvons voir que pour n'importe quel seuil, le modèle entraîné avec la résolution *iso max* a une meilleure sensibilité lésionnelle, ce qui signifie qu'il détecte plus de lésions. Cela sous-entend que le sous-échantillonnage aide à détecter plus de lésions. Au contraire, le modèle travaillant avec des images de résolution native a une meilleure précision lésionnelle, et détecte donc moins de fausses lésions. Cependant, la meilleure précision est atteinte à 0.618, ce qui signifie qu'en moyenne seulement 62% des lésions détectées sont vraies. De la même façon, seulement 48% des lésions ont été détectées en moyenne au mieux avec un seuil à 0,2 et la résolution *iso max*. Le faible taux de détection s'explique par le grand nombre de petites lésions dans l'ensemble de données qui ne sont pas détectées. Le problème de détection des petites lésions est un problème connu dans la segmentation automatique des lésions de la SEP [102],[103]. Les meilleurs compromis de détection ont été trouvés avec un

Seuil	Résolution	Dice	F1	Précision _l	Sensibilité _l
0.2	<i>iso max</i>	0.620	0.417	0.414	0.481
0.2	<i>natif</i>	0.647	0.457	0.474	0.476
0.3	<i>iso max</i>	0.647	0.455	0.486	0.469
0.3	<i>natif</i>	0.665	0.487	0.562	0.458
0.4	<i>iso max</i>	0.662	0.465	0.518	0.457
0.4	<i>natif</i>	0.667	0.488	0.591	0.440
0.5	<i>iso max</i>	0.675	0.482	0.553	0.446
0.5	<i>natif</i>	0.660	0.482	0.618	0.416
0.6	<i>iso max</i>	0.672	0.471	0.551	0.436
0.6	<i>natif</i>	0.641	0.454	0.575	0.397
0.7	<i>iso max</i>	0.662	0.452	0.529	0.418
0.7	<i>natif</i>	0.622	0.440	0.583	0.374

Table 5.3 – Performances en score de Dice, F1, précision et sensibilité avec les différents seuils pour les modèles entraînés avec les résolutions *natif* et *iso max* et une taille de patch de $32 \times 32 \times 32$. Les meilleurs scores sont en **gras**. Précision_l et Sensibilité_l correspondent respectivement à la précision de détection des lésions et la sensibilité à la détection des lésions.

seuil de 0,4 et la résolution *natif* en termes pour obtenir un score F1 de 0,488 et un score de Dice de 0,675 avec un seuil de 0,5 et la résolution *iso max*. Cependant, les résolutions *iso max* et *natif* donnent des performances proches en termes de score Dice et de score F1.

Le choix de la taille du patch et du rééchantillonnage des images pour la segmentation des lésions de SEP est rarement détaillé, mais a un impact important sur les performances du modèle entraîné.

Il apparaît dans notre étude que, travailler avec une taille de patch de $32 \times 32 \times 32$ avec une taille de lot de 2048 est le meilleur compromis par rapport à des tailles de patch plus grandes et des tailles de lot plus petites. Les résultats montrent que le suréchantillonnage de l'acquisition sagittale induit, dans notre cas, une perte de performance. Cela indique que le rééchantillonnage isovolumétrique n'est pas toujours la meilleure façon de travailler avec des images 3D. Cependant, ici, le sous-échantillonnage de l'image a amélioré la sensibilité lésionnelle du modèle au détriment de la précision lésionnelle. Les résolutions *natif* et *iso max* se sont révélées très similaires en termes de score de Dice et de score F1. La meilleure valeur de seuil varie entre]0.4 ; 0.6[selon la résolution et la métrique considérée. Cependant, dans les prédictions réalisées, moins de 50% des lésions sont détectées en moyenne ce qui reste insuffisant et doit être investigué.

5.2 Adaptation de la fonction de coût vers une meilleure détection des lésions

Dans la section précédente, nous avons observé qu’il y a des problèmes de détections des lésions de SEP avec une sensibilité maximale atteinte à 0,481, c’est à dire que moins de la moitié des lésions sont détectées en moyenne dans l’ensemble de test. Les grandes lésions sont généralement détectées, mais les plus petites lésions le sont plus rarement. Le réseau ne détecte pas assez les petites lésions malgré des scores de Dice acceptables supérieurs à 0,6.

La segmentation privilégiée des grandes lésions s’explique en particulier par la fonction de coût employée, le coût de Dice.

Le problème de détection des petites lésions est un problème déjà pointé du doigt par Fartaria *et al.* [102] et Zhang *et al.* [103] et constitue un problème ouvert.

Nous présentons ici le travail que nous avons mené pour améliorer la détection des petites lésions de SEP en pondérant la fonction de coût en fonction de la taille des lésions.

5.2.1 Analyse du problème

Nous pensons que le problème de détection des petites lésions provient des données et de la fonction de coût adoptées pour la segmentation. D’une part, les petites lésions sont moins nombreuses et plus difficiles à détecter, d’autre part, la fonction de coût, le coût de Dice présenté dans l’**Equation 3.1** n’est pas conçue pour la détection des lésions, mais pour la segmentation voxelique des lésions.

Le score de Dice, sur lequel est calqué le coût de Dice s’écrit aussi :

$$Dice = \frac{2TP}{2TP + FP + FN}$$

où TP est le nombre de voxels correctement détectés (vrais positifs), FP le nombre de voxels segmentés n’appartenant pas à une lésion (faux positifs) et FN le nombre de voxels non détectés (faux négatifs).

Le coût de Dice encourage donc la détection d’un maximum de voxels de lésions, tout en décourageant la détection de voxels n’appartenant pas à une lésion et la non-détection des voxels de lésions. Seulement, dans les images des patients atteints de SEP, il y a des grandes, voir très grandes lésions périventriculaires (caractéristiques de la maladie) et des lésions généralement plus petites dans les autres localisations (tronc cérébrale,

le cervelet, la substance blanche et dans l'espace juxtacortical). L'optimisation par descente de gradient avec le coût de Dice pousse à privilégier la détection des grandes lésions potentiellement plus détectables et beaucoup plus représentées en termes de voxel, car très volumiques et donc permettent d'augmenter "facilement" le nombre de TP . De plus, comme les petites lésions sont moins volumiques et donc moins représentées, leur non-détection entraîne une quantité négligeable de FN par rapport au nombre de TP obtenus avec les grandes lésions. De façon perverse, la pénalisation de détection des voxels en dehors des lésions, FP , peut aussi décourager la prise de risque à segmenter les petites lésions, plus subtiles et potentiellement plus facilement confondables avec des structures saines du cerveau.

De façon désirable, nous voulons pouvoir apprendre notre réseau avec un coût sur la segmentation et sur la détection des lésions de SEP pour apprendre à segmenter correctement un maximum de lésions. Le score F1 est une mesure de détection des lésions, mais son calcul nécessite des étapes longues qui ne sont pas parallélisables et surtout, pas ou très difficilement dérivables comme décrit par Kamraoui *et al.* [104]. Nous proposons donc de modifier et pondérer le coût de Dice pour pondérer de façon plus importante les petites lésions et de ce fait améliorer la détection des petites lésions.

5.2.2 Matériels et méthodes

Nous présentons, ici, une méthode pour apprendre la segmentation des lésions de SEP en donnant plus d'importance aux petites lésions qui sont généralement oubliées par les algorithmes de segmentation automatique. Nous présentons un modèle pour la pondération des lésions en fonction de leur taille, puis abordons l'adaptation réalisée sur la fonction de coût employée et enfin présentons l'ensemble de données maison utilisé et les protocoles d'apprentissages et de test.

Pondération des lésions

La pondération des lésions est conçue pour donner plus de poids aux petites lésions qu'aux grandes lésions pour contrecarrer les biais introduits par les données et la fonction de coût présentés dans la **Section 5.2.1**.

Nous avons pondéré nos lésions en fonction de leur taille, seulement, nous avons un spectre de taille de lésion important allant de quelques voxels à plusieurs milliers de voxels voir dizaine de milliers avec une résolution de $1 \times 1 \times 1 \text{ mm}^3$ soit des lésions de quelques mm^3 à quelques cm^3 . En l'état, une pondération en fonction de

la taille de type $1/taille_{lesion}$ n'est pas adaptée, en effet, la fonction inverse possède une décroissance trop importante. Nous avons donc cherché à mettre un point une fonction de pondération respectant les points suivants :

- être décroissante avec la taille des lésions pour pondérer en fonction de la taille,
- avoir une vitesse de décroissance modulable pour pouvoir séparer plus ou moins fortement les pondérations des grandes et petites lésions,
- produire une pondération proche pour les petites lésions (une lésion de 2 ou 5 voxels doit avoir à peu près le même poids) pour ne pas encourager uniquement les toutes petites lésions, mais un spectre de petites lésions.

Nous avons choisi d'adapter la fonction logistique qui modélise d'équation :

$$f(x) = \frac{K}{1 + \alpha e^{-rx}} \quad (5.1)$$

où K , r et α sont des réels positifs pour obtenir une courbe sigmoïde qui est croissante, dont la croissance est modulable et qui a un plateau au début et à la fin permettant de pondérer les petites lésions proches avec des valeurs proches. Nous avons formalisé notre fonction de pondération par le modèle suivant :

$$f(x) = v_{max} - \frac{k}{1 + \alpha \times e^{\frac{-10x}{inter}}} \quad (5.2)$$

où x est la taille de la lésion, v_{max} est la valeur maximale de pondération, k la valeur maximale de la partie sigmoïde de la fonction de sorte que $v_{max} - k$ donne la valeur minimale de pondération, α paramétrise la translation de la fonction dans l'axe x et $inter$ détermine l'intervalle x entre le plateau maximal et minimal de la courbe. Le terme $10x$ a été fixé arbitrairement pour que les points distants de $inter/2$ du point de symétrie de la courbe soient suffisamment proches des asymptotes horizontales. Les paramètres de la fonction ont été fixés pour obtenir la courbe représentée dans la **Figure 5.2**.

Le paramètre α est fixé à $\alpha = \frac{1}{e^{-5}}$ pour fixer le point de symétrie de la courbe à la moitié de l' $inter$, et qui en d'autres termes, fixe la fin de la première asymptote à 0 et le début de la deuxième asymptote à l' $inter$.

Il y a trois paramètres à fixer pour la pondération :

- v_{max} qui fixe la valeur maximale de pondération,
- k qui définit l'intervalle de pondération,
- $inter$ qui détermine l'intervalle entre l'asymptote maximale et minimale.

Dans le but d'améliorer la détection des lésions et en particulier les petites lésions nous avons testé plusieurs valeurs de paramétrage de la fonction de pondération.

La fonction de pondération de l'**Equation 5.2** est appliquée dans la fonction de coût sous

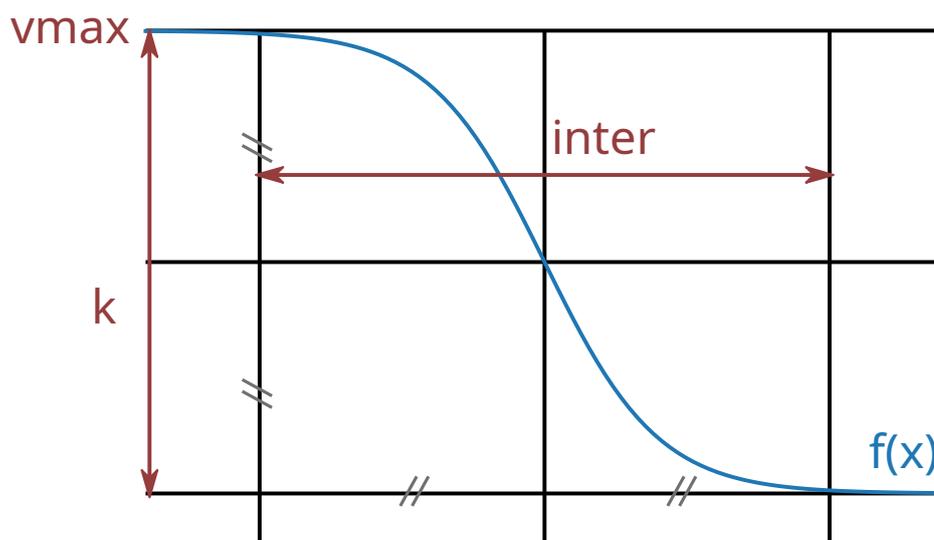


Figure 5.2 – La fonction de pondération utilisée avec sa représentation et son paramétrage k , $vmax$ et $inter$ représenté.

forme de masque de pondération W^{les} avec pour chaque voxel de lésion la pondération calculée pour la taille de lésion à laquelle il appartient.

Fonction de coût

La fonction de coût employée est calquée sur le coût de Dice précédemment mis à profit d'**Equation** 3.1. Pour tenir compte de la pondération, la fonction de coût a été reformulée comme :

$$\text{Coût Dice pondéré}(P, \hat{P}, W^{les}) = -2 \frac{\sum_{b=1}^B \sum_{h=1}^H \sum_{w=1}^W \sum_{d=1}^D \sum_{s=1}^S P_{c,b,h,w,d,s} \hat{P}_{c,b,h,w,d,s} W_{c,b,h,w,d,s}^{les}}{\sum_{b=1}^B \sum_{h=1}^H \sum_{w=1}^W \sum_{d=1}^D \sum_{s=1}^S P_{c,b,h,w,d,s} + \hat{P}_{c,b,h,w,d,s}} \quad (5.3)$$

où B est la taille du lot, H la hauteur, W la largeur D la profondeur de l'image, S le nombre de segmentations expert, $P \in \{0; 1\}$ la segmentation binaire des radiologues et $\hat{P} \in [0; 1]$ la prédiction du réseau.

Données

Nous avons choisi de travailler, ici, sur un ensemble de données nouveau interne au CHU de Poitiers, acquis par le laboratoire I3M. Ce choix s'explique par la découverte

de certaines incohérences dans l'ensemble d'images publique de Lesjak *et al.* [89] et par le fait que cet ensemble ne soit plus accessible en ligne et donc plus public.

L'ensemble de données utilisé a été réalisé avec 35 patients diagnostiqués SEP du CHU de Poitiers. Pour ces patients les images FLAIR ont été acquises sur les scanners IRM Siemens Verio et Skyra du CHU de Poitiers avec un champs magnétique B_0 de 3 Tesla. Les images ont été acquises de façon sagittale avec une résolution de $1 \times 0,5 \times 0,5 \text{ mm}^3$. Les masques de segmentation ont été segmentés coupe à coupe pour chaque examen sur l'image FLAIR acquise sans prétraitement par une interne en neurologie de l'hôpital.

Nous avons conservé 20 patients pour l'entraînement et 15 patients pour le test. Conscients que les profils des examens sont différents et qu'ils mènent à différentes qualités d'exemples d'apprentissage (voir **Section** 4.3), nous avons réparti les patients dans l'ensemble d'apprentissage et de test de façon équilibrée en fonction du profil lésionnel visualisé de chacun.

Nous avons prétraité les images FLAIR en rééchantillonnant les images à une résolution de 1 mm^3 isométrique pour travailler avec une résolution physique isovolumétrique avec un rééchantillonnage proche de la résolution *iso max* de la **Section** 5.1 pour laquelle nous avons eu une bonne segmentation et détection des lésions de SEP tout en ayant une résolution cohérente avec l'acquisition des images. Les images ont été masquées avec le masque du cerveau calculé avec l'outil *BrainSuite extractor* pour ne conserver que l'encéphale. Pour rendre les intensités comparables, nous avons mis en correspondance les histogrammes [48] et centré réduit les valeurs d'intensité des images.

Paramètres d'apprentissage

L'apprentissage s'est effectué avec l'architecture du MPU-next (voir **Section** 3.3) avec les poids initialisés avec les mêmes valeurs de façon uniforme [21]. Pour suivre le travail effectué dans la **Section** 5.1, nous avons réalisé notre entraînement avec une taille de patch de $32 \times 32 \times 32$ avec des lots de taille 2048. Nous avons appris les modèles avec 6144 patchs par examen d'apprentissage par époque pendant 40 époques. Le modèle a été optimisé avec l'optimiseur Adam [79] avec un taux d'apprentissage de 0,0004.

Test

Les prédictions sont générées par patches régulièrement espacés de 8 voxels dans chaque direction spatiale, puis moyennés spatialement pour obtenir l'image de prédiction. La prédiction est ensuite rééchantillonnée et alignée avec l'image de départ avant d'être binarisée au seuil de 0,5 et d'être comparée avec la segmentation radiologique de référence.

5.2.3 Résultats et analyse

Nous avons testé différents paramétrages de la fonction de pondération présentée dans l'**Equation 5.2**. Nous avons effectué un apprentissage sans pondération pour comparaison et avons entraîné en faisant varier $l'inter$ pour séparer plus ou moins rapidement les “grandes” lésions (pondérées avec des valeurs proches de l'asymptote minimale $vmax - k$) des “petites” lésions (pondérées avec des valeurs proches de l'asymptote maximale, $vmax$) et en faisant varier k pour contraster plus ou moins la pondération entre grandes et petites lésions. Nous avons testé plusieurs valeurs k parfois très proches de $vmax$. Pour avoir des valeurs de k proches de $vmax$ sans problème de propagation du gradient ou de précision flottante, nous avons fixé pour chaque entraînement pondéré $vmax = 10$. Les valeurs d' $inter$ et de k testées ainsi que l'allure des sigmoïdes de pondérations sont illustrées dans la **Figure 5.3**.

Nous nous sommes intéressés aux performances moyennes réalisées sur l'ensemble de test et, pour valider notre méthode, nous avons procédé à une visualisation de certaines segmentations prédites avec différentes pondérations et avons évalué leur qualité visuellement avec l'aide de la radiologue qui a segmenté les lésions.

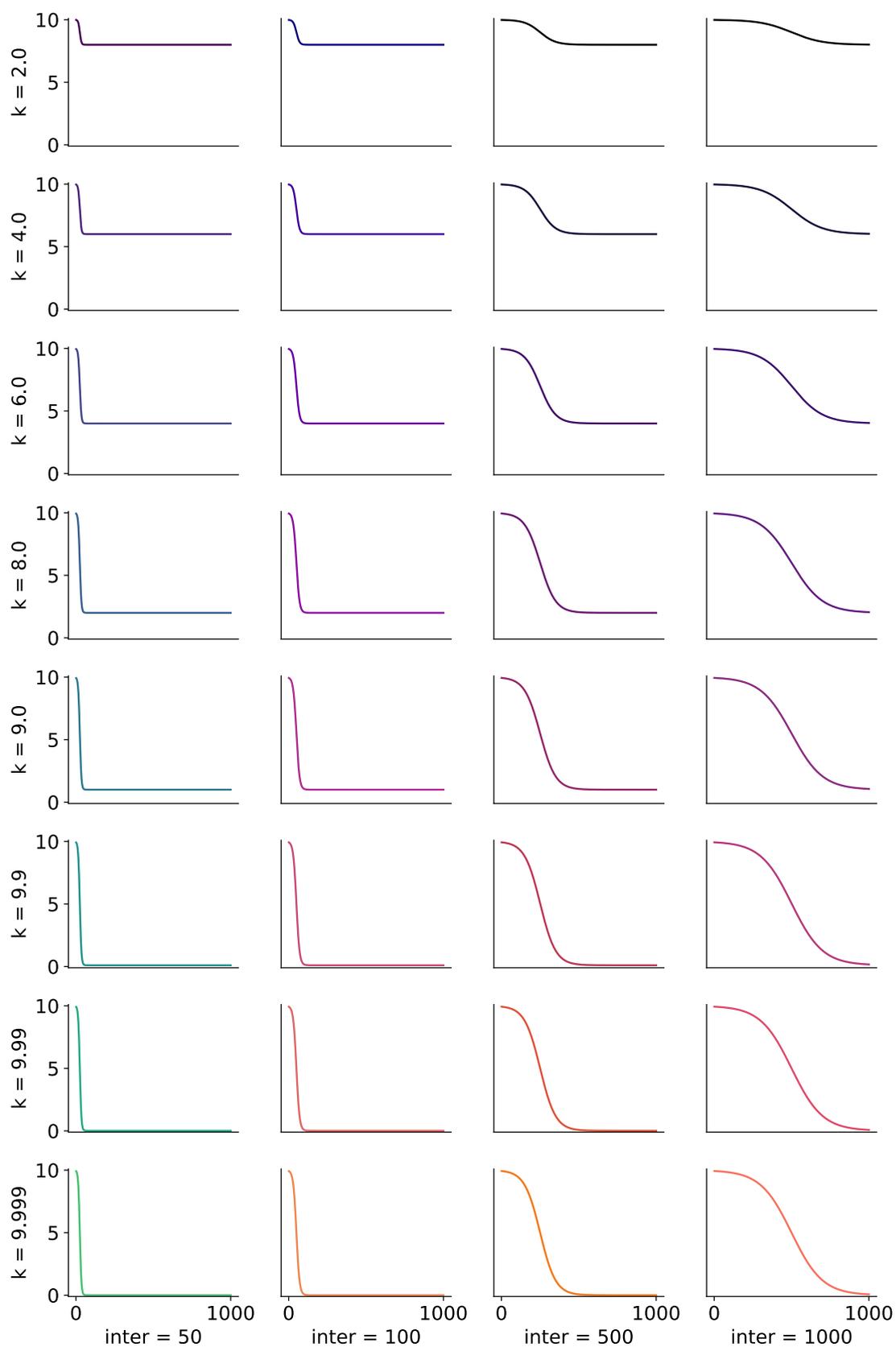


Figure 5.3 – Les différentes valeurs d'*inter* et de k testées avec les profils de pondérations.

Dice	F1	Sensibilité_L	Précision_L
0.5990	0.4883	0.4490	0.5764

Table 5.4 – Performances moyennes de référence sur l'ensemble de test avec le modèle appris sans pondérations. Sensibilité_L est la sensibilité de détection des lésions et Précision_L la précision de détection des lésions.

Performances moyennes

Nous pouvons voir dans la **Figure 5.4** et le **Tableau 5.4** qu'en fonction de la pondération, des résultats différents sont obtenus en termes de score moyen de Dice, F1, précision et sensibilité de détection lésionnelle.

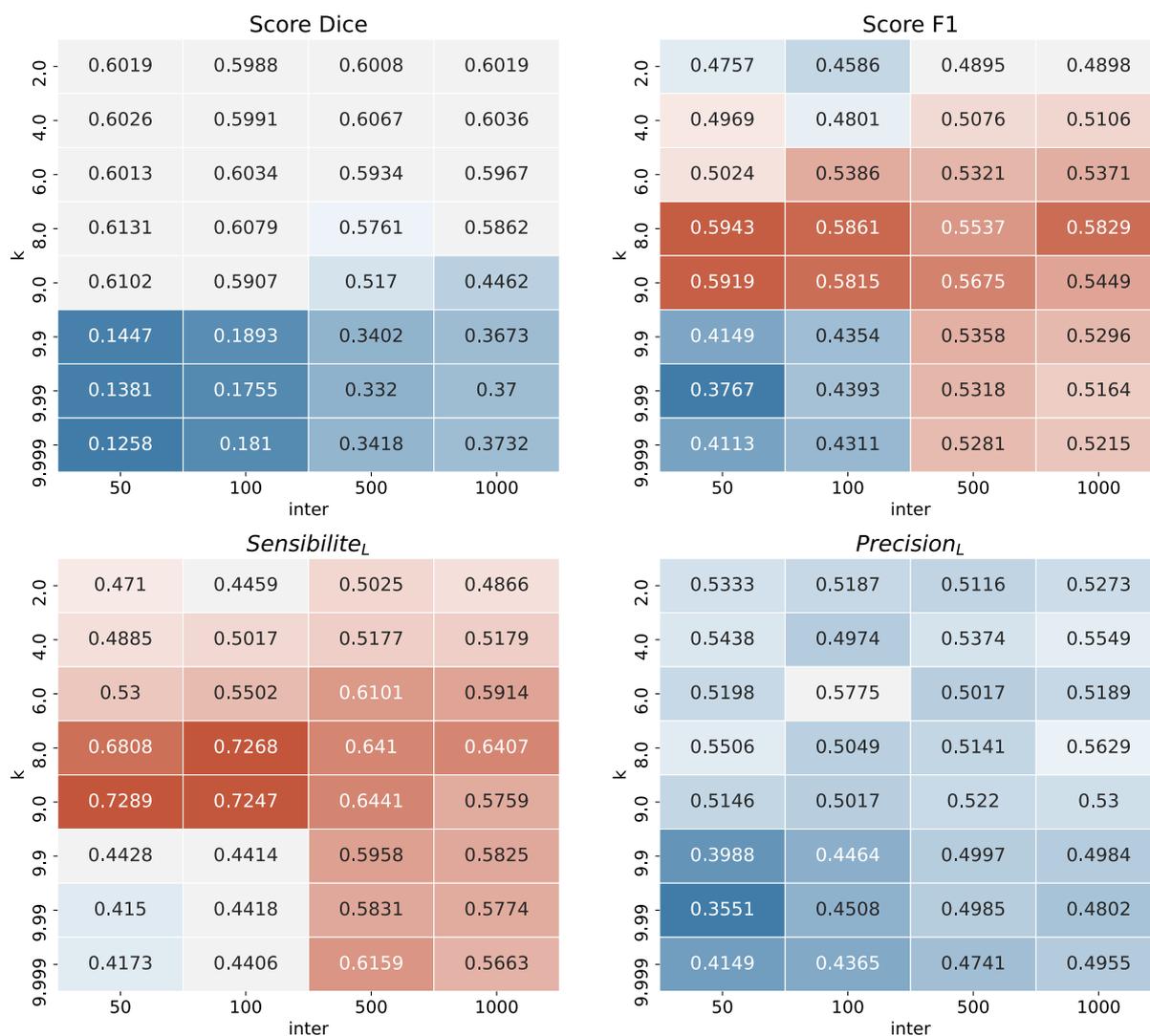


Figure 5.4 – Les scores moyens de score de Dice, F1, sensibilité et précision de détection obtenues en variant les valeurs de k et d' $inter$. Pour chaque métrique, si la couleur est bleue cela signifie que la valeur est inférieure à la performance sans pondération (dans le **Tableau 5.4**) et si la couleur est rouge cela signifie qu'elle est supérieure.

Pour toutes les métriques, nous observons des comportements différents à partir de $k = 9,9$. Pour chaque métrique, quand $k \geq 9,9$ et $inter \leq 100$, nous obtenons les moins bonnes performances, inférieures aux scores obtenus sans pondération du **Tableau 5.4**. Cela indique que la séparation rapide et très contrastée des pondérations des grandes et petites lésions entraîne une baisse générale de la qualité de détection et de segmentation des lésions. Lorsque l' $inter$ augmente avec les mêmes valeurs de k , nous observons une amélioration qui montre qu'en conservant les mêmes valeurs de pondérations, mais en les étalant plus sur l'intervalle de taille, le modèle apprend à détecter plus de lésions et à mieux les segmenter.

Dans la **Figure 5.4**, nous observons qu'en augmentant k jusqu'à 8, et 9, et en diminuant

l'inter jusqu'à 50, 100 il y a une augmentation du score de Dice, F1 et de la sensibilité de détection lésionnelle alors qu'il n'y a pas vraiment de tendance sur la précision de détection. La précision de détection lésionnelle est même inférieure que sans pondération pour presque chaque paramétrage évalué, cette observation reste cohérente avec la pondération qui est conçue pour détecter plus de petites lésions qui sont généralement plus difficiles à détecter et plus facilement confondues avec des structures saines de l'encéphale.

Les pondérations testées dans la **Figure 5.4** n'améliorent que faiblement le score de Dice jusqu'à 0,6103 pour $k = 8$ et $inter = 50$ contre 0,5990 sans pondération dans le **Tableau 5.4**. La segmentation voxelique est donc légèrement améliorée, cependant, la détection des lésions est drastiquement améliorée avec un score F1 jusqu'à 0,5943 pour les mêmes valeurs de k et d'*inter* comparé à un score F1 de 0,4883 sans pondération. Le gain apporté au score F1 s'explique par l'augmentation drastique de la sensibilité de détection apportée par la pondération allant jusqu'à 0,7289 en moyenne pour $k = 9$ et $inter = 50$. À l'opposé, la précision de détection est légèrement inférieure entre 0,5 et 0,55 pour les meilleurs scores F1 contre 0,5764 sans pondération. Ce comportement est cohérent avec l'objectif de la pondération de détecter plus de lésions et particulièrement les plus petites. En effet, sans pondération, le modèle appris détecte moins de la moitié des lésions avec une précision proche de 58% alors qu'avec le bon jeu de pondération nous pouvons détecter plus de 68% des lésions voir 73% avec une précision moindre de 3% à 8%. Dans ces cas-ci la pondération augmente drastiquement la balance sensibilité/précision avec un score F1 aux alentours de 0,59 contre 0,49 sans pondération.

En moyenne, les meilleures segmentations et détections de lésions sont atteintes pour $k = 8$ avec $inter = 50$ avec une meilleure précision et pour $k = 9$ avec $inter = 50$ pour une meilleure sensibilité. En mettant en correspondance ces résultats avec la **Figure 5.3**, nous avons les meilleures performances de détection en pondérant 5 à 10 fois plus les petites lésions par rapport aux grandes avec une séparation progressive des lésions par taille jusqu'à une taille 50 voxels soit $50mm^3$ avec la résolution employée.

Visualisation

La visualisation a été réalisée sur plusieurs patients avec plusieurs pondérations. Pour plus de clarté, nous avons illustré au mieux les observations avec un exemple dans la **Figure 5.5**.

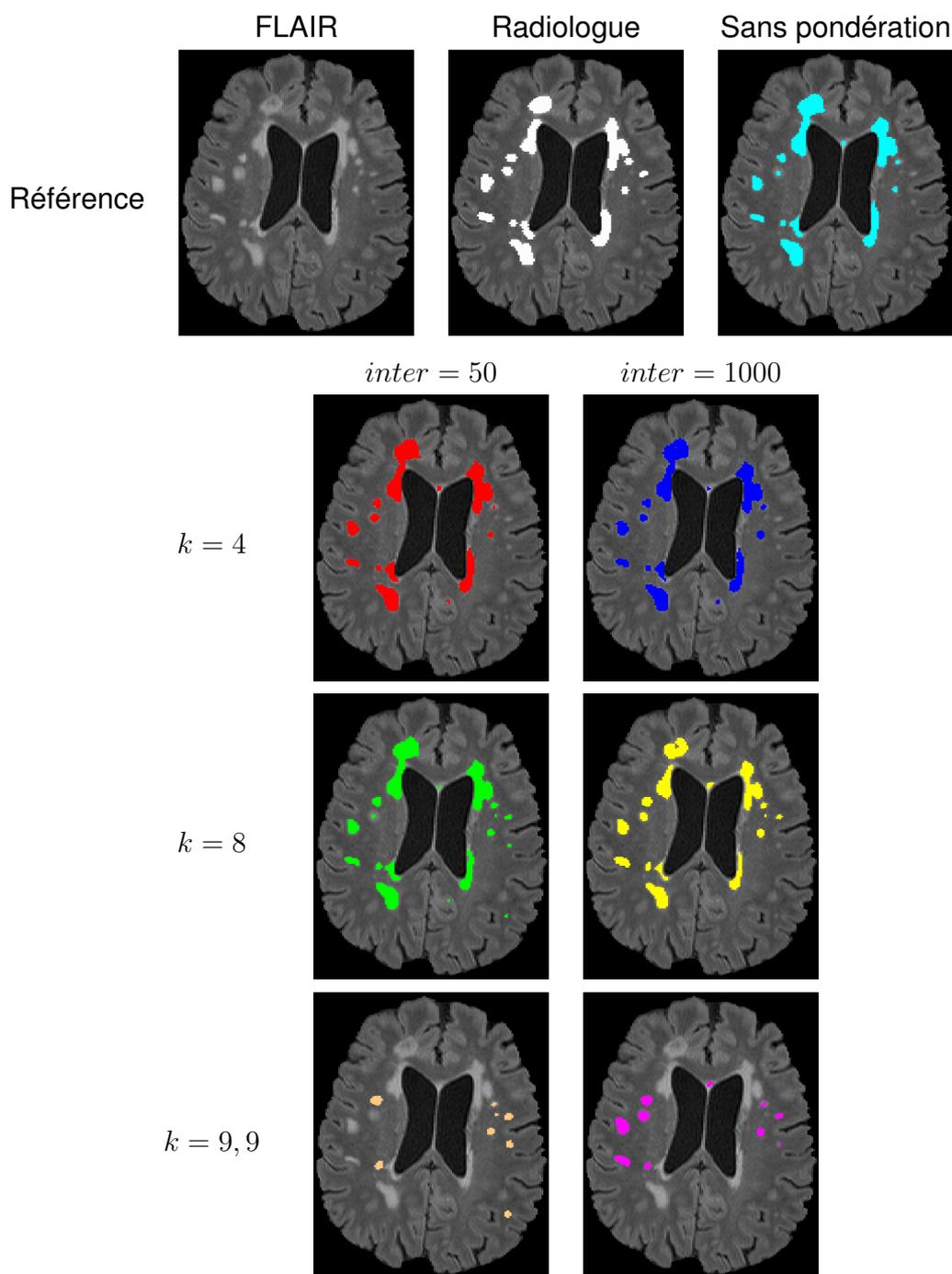


Figure 5.5 – Visualisation des segmentations avec apprentissage du modèle avec différentes pondérations.

Dans la **Figure 5.5**, nous observons, pour cet exemple, dans un premier temps que la segmentation sans pondération et la segmentation radiologie sont très proches, excepté une petite lésion non segmentée par le modèle proche de la substance grise du cortex de la partie droite de l'encéphale. Certaines parties sont sursegmentées et

d'autres sous-segmentées par rapport à la segmentation de référence.

Lorsque $k = 9,9$ nous observons que les plus grandes lésions ne sont pas segmentées du tout, indiquant que les petites lésions sont trop privilégiées pendant l'entraînement. Pour $k = 9,9$, nous observons qu'en augmentant *inter* de 50 à 1000, le modèle a tendance à segmenter de plus grandes lésions ce qui est cohérent avec la pondération dans la **Figure 5.3** qui est plus progressive.

Les segmentations pour $k = 4$ sont très proches des segmentations sans pondération ce qui est cohérent avec les résultats dans la **Figure 5.4** et qui s'expliquent par une différence de pondération faible entre les différentes tailles de lésions.

Pour $k = 8$, de nouvelles petites lésions sont segmentées qui ne sont pas sur la segmentation radiologique. Cependant, après vérification, il s'agit de bords de lésions qui n'ont pas été segmentées, de très petites lésions qui n'ont pas été aperçues lors de la segmentation radiologique et une lésion faussement positive. Avec $k = 8$, nous observons que l'augmentation d'*inter* de 50 à 1000 élimine les plus petites lésions segmentées, ce qui reste cohérent avec la pondération illustrée dans la **Figure 5.3**, en effet, le passage de grande à petite lésion est plus progressif et étalé, ce qui diminue légèrement l'importance des très petites lésions.

À l'observation des segmentations prédites, nous constatons qu'avec des valeurs de k se rapprochant de 10, il y a une tendance à ignorer les grandes lésions et à préférer uniquement les plus petites. Pour les valeurs de k à 8 et 9 et les valeurs d'*inter* à 50 et 100, nous avons observé de très bonnes segmentations avec très peu de faux négatifs. Un certain nombre de petites lésions qui ne sont pas détectées sans pondération le sont avec cette pondération.

En compilant les observations, l'influence du paramètre k permet de filtrer la taille des lésions segmentées avec un k petit, les petites lésions sont peu segmentées alors qu'au contraire quand k est grand, les grandes lésions sont faiblement segmentées. La valeur d'*inter* agit comme un curseur et contrebalance l'effet du paramètre k en augmentant. En effet, en augmentant l'*inter*, les différences de pondérations entre les petites et les grandes lésions s'amenuisent et les petites lésions sont moins détectées alors que les grandes lésions sont mieux segmentées.

D'une façon plus générale, à l'observation des différentes segmentations, nous nous sommes aperçus que les modèles performants ont tendance à segmenter plus largement en bordure de lésions que la segmentation radiologique de référence, baissant ainsi le score de Dice sans influence sur l'analyse radiologique de la segmentation en elle-même. Les images les plus bruitées sont aussi les images sur lesquelles les modèles ont été le moins performants, cependant, les modèles n'ont pas été entraînés

avec l'augmentation des données qui aurait pu pallier ce problème, au moins en partie. Les lésions à la base du crâne ne sont généralement pas segmentées, en effet, cette zone est aussi rarement segmentée dans la vérité terrain puisqu'elle contient, la plupart du temps, des artefacts dus au champ magnétique. Les modèles ont donc, appris à reconnaître cette zone et éviter de segmenter des lésions dans ce volume.

Nous avons observé quelques lésions qui ont été segmentées par le modèle, mais oubliées par notre radiologue. Cette dernière observation fait état d'un fait intéressant : un modèle automatique de segmentation, bien qu'imparfait, peut aider un praticien en détectant des lésions peu saillantes ou trop petites pour paraître évidente, particulièrement avec notre méthode de pondération.

5.3 Conclusion

Nous avons mis au point une analyse de l'impact du rééchantillonnage et de l'envoi des données pour la tâche de segmentation des lésions de SEP. Nous avons constaté que le sous-échantillonnage est préférable au suréchantillonnage pour les apprentissages réalisés. Pour des apprentissages comparables en nombre de voxels, il est préférable de travailler avec de nombreux petits patches de $32 \times 32 \times 32$ voxels par lot plutôt que de plus grands patches avec des lots moins diversifiés. Lors de ce travail, il a été constaté que la détection des lésions n'était pas suffisante malgré un bon score de Dice. La détection des petites lésions de SEP est un problème rencontré fréquemment dans la segmentation automatique des lésions de SEP. Ces lésions sont moins représentées en nombre de pixels et souffrent d'un problème de sous-représentation au sein des voxels de lésions en plus d'être plus difficiles à détecter.

Nous avons proposé de contrebalancer la faible représentation des voxels des petites lésions en mettant en place un système de pondération configurable dans la fonction de coût. En pondérant jusqu'à 5 à 10 fois plus les petites lésions de volume inférieur à $50mm^3$, la performance de segmentation est légèrement améliorée, mais le taux de détection est largement augmenté. L'amélioration de la détection s'explique principalement par l'augmentation de sensibilité de détection des lésions qui passe de 45% sans pondération jusqu'à 73% avec une baisse de 3% à 8% de précision de détection en moyenne uniquement avec notre méthode.

En visualisant les résultats de segmentation avec un avis radiologique, nous avons pu valider les tendances décrites par les métriques et observer le comportement

de segmentation de façon plus détaillée. Nous avons pu mettre en évidence certaines petites lésions segmentées par nos modèles pondérés qui n'avaient pas été segmentées dans la vérité terrain, montrant la complémentarité et l'aide apportée par ce genre d'approche avec l'analyse radiologique.



Conclusion

Sommaire

La SEP est une maladie neurodégénérative et est la première cause de handicap non traumatique chez les jeunes. La maladie provoque des lésions dans le système nerveux central qui sont visibles sur images IRM et particulièrement en FLAIR. L'examen IRM a une influence directe sur le diagnostic et la prise en charge du patient pour la prise du traitement qui ralentit l'avancée de la maladie. La segmentation automatique des lésions de sclérose en plaques sur les images IRM est champ de recherche actif qui a pour but d'aider le radiologue dans une tâche répétitive et quotidienne afin de fournir des analyses plus reproductives d'une mesure visuelle en amenant des métriques sur le profil lésionnel du patient.

Les méthodes les plus performantes pour la segmentation des lésions de SEP emploient des réseaux de neurones convolutifs qui sont des structures de données qui nécessitent généralement beaucoup de données et capacité de calcul pour leur entraînement. Ces besoins sont autant de freins au développement et à l'utilisation plus massive de ces outils dans l'environnement clinique. Le travail de thèse a consisté à concevoir des méthodes de segmentations performantes tout en limitant le besoin en données et en capacité de calcul.

Les principales contributions de ce travail ont été :

- le développement d'architectures de RNC légères et adaptées à la tâche de segmentation,
- l'étude et la réduction du nombre d'exams nécessaires avec un accent sur leurs qualités pour apprendre à segmenter les lésions,
- l'amélioration des performances de segmentation et de détection notamment des petites lésions.

Le travail sur l'architecture, traité dans le **Chapitre 3**, décompose et adapte l'architecture de référence pour la segmentation d'images médicales, le U-net, pour créer le MPU-net. Cette première architecture est performante et légère avec environ 33 000 paramètres d'apprentissage à optimiser ce qui constitue une architecture extrêmement légère pour un réseau de neurones convolutif. La faible paramétrisation de cette architecture permet d'économiser des ressources de calcul pendant son entraînement et l'inférence en évitant des besoins matériels importants et réduisant le besoin en exemple d'apprentissage. Le MPU-net, bien que performant, a une conception très simple qui a été améliorée en ajoutant des couches de régularisation et en concaténant des segmentations prédites à différents niveaux dans le modèle pour créer le MPU-next. Le MPU-next effectue de meilleures segmentations et est plus reproductible que le MPU-net pour un surcoût négligeable d'environ 4000 paramètres d'apprentissage et d'une couche convolutive. Par la suite, le MPU-next a été adapté de façon siamoise

pour le challenge MSSEG2 de segmentation des nouvelles lésions de SEP.

Dans le **Chapitre 4**, nous avons souhaité évaluer et réduire l'apport en données d'apprentissage. Dans un premier temps, partant de l'hypothèse que le nombre d'images annotées à disposition n'est pas suffisant, nous avons proposé une méthode d'autosupervisée de segmentation des lésions de SEP. L'autosupervision consiste à préentraîner le réseau de neurones sur une tâche prétexte, non supervisée, avec plus d'images d'apprentissage pour apprendre des caractéristiques réutilisables pour l'apprentissage supervisé sur la tâche d'intérêt réduisant ainsi le besoin en exemple d'apprentissage annoté. Cependant, la méthode d'autosupervision développée n'apporte qu'une très légère amélioration non significative de segmentation. Ainsi, après cette tentative de pallier au manque d'exemples d'apprentissage, nous avons décidé d'investiguer que l'apport en images annotées supervisées nécessaire avec le MPU-next. Il s'est avéré que seulement 10 examens d'apprentissages étaient suffisants, cependant, en dessous de cette limite, les différents apprentissages montrent des problèmes de reproductibilité. Ces problèmes de reproductibilité s'expliquent notamment par des profils lésionnels très différents d'un patient à un autre menant à différentes qualités d'exemples d'apprentissages. Pour réduire le nombre d'annotations nécessaires, nous avons conçu une méthode d'apprentissage semi-supervisée, tirant parti à la fois des données annotées et non annotées. Avec la méthode semi-supervisée mise en point et le bon choix d'examens d'apprentissage, il est possible d'entraîner un modèle plus performant en termes de segmentation et détection des lésions avec seulement un examen segmenté et trois examens segmentés comparé à un entraînement complètement supervisé sur 16 examens.

Les recherches présentées dans le dernier chapitre, le **Chapitre 5**, se concentrent sur l'amélioration de segmentation et de détection des lésions de SEP. Nous montrons que la façon d'échantillonner, de découper en patch et d'envoyer les images IRM durant l'entraînement de notre architecture ont un impact important sur les performances du modèle appris. Le suréchantillonnage est à privilégier au sous-échantillonnage lors de l'apprentissage. Il est aussi préférable d'entraîner les modèles avec un grand nombre de petits patches plutôt qu'avec un petit nombre de grands patches. Nous avons constaté un problème de détection des lésions, qui constitue un problème déjà soulevé dans la littérature sur le sujet. En effet, les petites lésions sont généralement oubliées par les modèles d'apprentissage profond. Nous avons alors conçu une méthode qui permet d'augmenter l'importance des petites lésions lors de l'apprentissage de la segmentation pour augmenter le taux de détection moyen des lésions de plus de 25% avec une faible baisse de la précision pour gagner significativement en détection des lésions, et notamment des petites.

Perspectives

La détection des nouvelles lésions de SEP est importante pour le diagnostic et le suivi des patients atteints par la maladie. En effet, l'apparition de nouvelles lésions est liée à la progression de la maladie et permet d'adapter en conséquence la prise en charge du patient. Cependant, la mise en évidence de ces nouveaux foyers inflammatoires est plus difficile, répétitive et chronophage que la détection simple des lésions puisqu'elle nécessite la comparaison coupe à coupe d'images acquises à différents temps. Nous avons proposé un premier modèle de segmentation des nouvelles lésions qui peut être amélioré de plusieurs façons : en ajoutant l'augmentation de données, en changeant le coût et en limitant l'impact de la faible représentation des nouvelles lésions au niveau voxelique des images par exemple.

L'ajout d'autres acquisitions IRM a le potentiel d'améliorer encore les segmentations réalisées par le modèle, notamment les séquences DIR qui permettent une meilleure détection des lésions juxtacorticales. De plus, l'arrivée de l'IRM MAGNETOM Terra de Siemens Healthineers avec son champ magnétique de 7 Tesla, donne accès à l'acquisition d'images avec une résolution inframillimétrique de grande qualité qui permettent la visualisation des plus petites lésions et accéder aux plaques corticales rares et très petites jusqu'alors invisibles en IRM 3 Tesla. La segmentation de lésions sur images acquises à 7 Tesla offre un certain nombre de challenges liés notamment à l'élimination d'artefacts spécifiques, mais a le potentiel de quantifier encore plus finement le profil lésionnel du patient. L'analyse d'image avec une résolution physique très fine est plus complexe et pose alors des problématiques intéressantes de segmentation multiéchelles avec une gestion fine du déséquilibre de représentation des différentes tailles de lésions.

L'intégration dans l'environnement clinique de nos méthodes de segmentation est aussi un sujet important qui permet de faire le lien entre la recherche et la pratique clinique qui est chère au laboratoire I3M. L'équipe d'I3M a déjà commencé à travailler à l'intégration du MPU-next pour générer automatiquement les cartes de segmentation des lésions pour les patients atteints de SEP lors de l'examen et ainsi amener directement les bénéfices de nos recherches aux praticiens. Les cartes de segmentations générées peuvent être directement traitées pour obtenir des métriques et caractéristiques de haut niveau sur le profil lésionnel du patient vers un ajout supplémentaire d'information pour une médecine de plus en plus personnalisée.

L'ajout de l'expert dans la boucle est, selon nous, une étape nécessaire à l'adoption des techniques de segmentation automatique en environnement clinique. L'approche automatique permet d'apporter des métriques et d'apporter un résultat plus reproductible sur une tâche visuelle, cependant, la machine, comme l'expert n'est pas infaillible. La réelle plus-value est apportée par la conjugaison des deux analyses et doit passer par une phase d'échange entre les deux. L'algorithme automatique apprend des annotations réalisées par le radiologue, mais produit des segmentations fixes une fois qu'il a été entraîné. Le développement d'une technique interactive qui permet au radiologue de corriger les segmentations prédites permettrait de ramener le radiologue dans la boucle de segmentation automatique et ainsi renforcer l'emploi de tels outils. De plus, une telle méthode permettrait aussi, à terme, de créer plus facilement des ensembles de données annotées en simplifiant la tâche d'annotation et ainsi en permettant de réentraîner à moindre coût des modèles performants vers des dynamiques d'amélioration continue en clinique pour faire bénéficier à la fois le modèle et le radiologue.

7

Bibliographie

Sommaire

Liste des travaux	149
Références	150

Liste des Travaux

Journaux internationaux à comité de lecture

- A. Fenneteau, P. Bourdon, D. Helbert, C. Fernandez-Maloigne, C. Habas, and R. Guillevin, "Investigating efficient cnn architecture for multiple sclerosis lesion segmentation," *Journal of Medical Imaging*, vol. 8, no. 1, p. 014504, 2021.

Conférences internationales à comité de lecture

- A. Fenneteau, P. Bourdon, D. Helbert, C. Fernandez-Maloigne, C. Habas, and R. Guillevin, "Cnn for multiple sclerosis lesion segmentation: how many patients for a fully supervised method?" In *International Conference on Advances in Biomedical Engineering*, 2021.
- —, "Siamese convolutional neural network for new multiple sclerosis lesion segmentation," in *MSSEG-2 challenge proceedings: Multiple sclerosis new lesions segmentation challenge using a data management and processing infrastructure. MICCAI 2021*, 2021.
- —, "Learning a CNN on multiple sclerosis lesion segmentation with self-supervision," in *3D Measurement and Data Processing, IS&T Electronic Imaging 2020 Symposium*, Best paper award 3D Measurement and Data Processing, San Francisco, United States, Jan. 2020.

Conférences nationales à comité de lecture

- A. Fenneteau, P. Bourdon, D. Helbert, C. Fernandez-Maloigne, and C. de Poitiers, "Apprentissage d'un cnn sur la segmentation des lésions de sclérose en plaques avec auto-supervision," in *RFIAP*, 2020.

Références

- [1] R. Dobson and G. Giovannoni, “Multiple sclerosis a review,” *European journal of neurology*, vol. 26, no. 1, pp. 27–40, 2019.
- [2] M. T. Wallin, W. J. Culpepper, E. Nichols, Z. A. Bhutta, T. T. Gebrehiwot, S. I. Hay, I. A. Khalil, K. J. Krohn, X. Liang, M. Naghavi, *et al.*, “Global, regional, and national burden of multiple sclerosis 1990–2016: a systematic analysis for the global burden of disease study 2016,” *The Lancet Neurology*, vol. 18, no. 3, pp. 269–285, 2019.
- [3] C. Baecher-Allan, B. J. Kaskow, and H. L. Weiner, “Multiple sclerosis: mechanisms and immunotherapy,” *Neuron*, vol. 97, no. 4, pp. 742–768, 2018.
- [4] F. D. Lublin, S. C. Reingold, J. A. Cohen, G. R. Cutter, P. S. Sørensen, A. J. Thompson, J. S. Wolinsky, L. J. Balcer, B. Banwell, F. Barkhof, *et al.*, “Defining the clinical course of multiple sclerosis: the 2013 revisions,” *Neurology*, vol. 83, no. 3, pp. 278–286, 2014.
- [5] A. J. Thompson, B. L. Banwell, F. Barkhof, W. M. Carroll, T. Coetzee, G. Comi, J. Correale, F. Fazekas, M. Filippi, M. S. Freedman, *et al.*, “Diagnosis of multiple sclerosis: 2017 revisions of the mcdonald criteria,” *The Lancet Neurology*, vol. 17, no. 2, pp. 162–173, 2018.
- [6] A. Shoeibi, M. Khodatars, M. Jafari, P. Moridian, M. Rezaei, R. Alizadehsani, F. Khozeimeh, J. M. Gorriz, J. Heras, M. Panahiazar, *et al.*, “Applications of deep learning techniques for automated multiple sclerosis detection using magnetic resonance imaging: a review,” *arXiv preprint arXiv:2105.04881*, 2021.
- [7] W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, 1943.
- [8] D. O. Hebb, *The organization of behavior: a neuropsychological theory*. J. Wiley; Chapman & Hall, 1949.

- [9] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain.," *Psychological review*, vol. 65, no. 6, p. 386, 1958.
- [10] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *nature*, vol. 323, no. 6088, pp. 533–536, 1986.
- [11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: a large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, IEEE, 2009, pp. 248–255.
- [12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.
- [13] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of control, signals and systems*, vol. 2, no. 4, pp. 303–314, 1989.
- [14] M. Bianchini and F. Scarselli, "On the complexity of neural network classifiers: a comparison between shallow and deep architectures," *IEEE transactions on neural networks and learning systems*, vol. 25, no. 8, pp. 1553–1565, 2014.
- [15] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.
- [16] M. Sun, Z. Song, X. Jiang, J. Pan, and Y. Pang, "Learning pooling for convolutional neural network," *Neurocomputing*, vol. 224, pp. 96–104, 2017.
- [17] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.
- [18] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [20] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.
- [21] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.
- [22] E. Kats, J. Goldberger, and H. Greenspan, "Soft labeling by distilling anatomical knowledge for improved ms lesion segmentation," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, IEEE, 2019, pp. 1563–1566.

- [23] M. Cabezas, J. Corral, A. Oliver, Y. Diez, M. Tintore, C. Auger, X. Montalban, X. Llado, D. Pareto, and A. Rovira, “Improved automatic detection of new t2 lesions in multiple sclerosis using deformation fields,” *American Journal of Neuroradiology*, vol. 37, no. 10, pp. 1816–1823, 2016.
- [24] A. Danelakis, T. Theoharis, and D. A. Verganelakis, “Survey of automated multiple sclerosis lesion segmentation techniques on magnetic resonance imaging,” *Computerized Medical Imaging and Graphics*, vol. 70, pp. 83–100, 2018.
- [25] M. Styner, J. Lee, B. Chin, M. S Chin, O. Commowick, H.-H. Tran, V. Jewells, and S. Warfield, “3d segmentation in the clinic: a grand challenge ii: ms lesion segmentation,” *MIDAS Journal*, Nov. 2007.
- [26] A. Carass, S. Roy, A. Jog, J. L. Cuzzocreo, E. Magrath, A. Gherman, J. Button, J. Nguyen, F. Prados, C. H. Sudre, *et al.*, “Longitudinal multiple sclerosis lesion segmentation: resource and challenge,” *NeuroImage*, vol. 148, pp. 77–102, 2017.
- [27] O. Commowick, A. Istace, M. Kain, B. Laurent, F. Leray, M. Simon, S. C. Pop, P. Girard, R. Ameli, J.-C. Ferré, *et al.*, “Objective evaluation of multiple sclerosis lesion segmentation using a data management and processing infrastructure,” *Scientific reports*, vol. 8, no. 1, p. 13 650, 2018.
- [28] A. Fenneteau, P. Bourdon, D. Helbert, C. Fernandez-Maloigne, C. Habas, and R. Guillevin, “Investigating efficient cnn architecture for multiple sclerosis lesion segmentation,” *Journal of Medical Imaging*, vol. 8, no. 1, p. 014 504, 2021.
- [29] —, “Cnn for multiple sclerosis lesion segmentation: how many patients for a fully supervised method?” In *International Conference on Advances in Biomedical Engineering*, 2021.
- [30] —, “Siamese convolutional neural network for new multiple sclerosis lesion segmentation,” in *MSSEG-2 challenge proceedings: Multiple sclerosis new lesions segmentation challenge using a data management and processing infrastructure. MICCAI 2021*, 2021.
- [31] D. Goldberg-Zimring, A. Achiron, S. Miron, M. Faibel, and H. Azhari, “Automated detection and characterization of multiple sclerosis lesions in brain mr images,” *Magnetic resonance imaging*, vol. 16, no. 3, pp. 311–318, 1998.
- [32] O. Commowick, F. Cervenansky, and R. Ameli, *Msseg challenge proceedings: multiple sclerosis lesions segmentation challenge using a data management and processing infrastructure*, Athenes, Greece, 2016. [Online]. Available: <https://www.hal.inserm.fr/inserm-01397806>.
- [33] R. McKinley, R. Wepfer, T. Gundersen, F. Wagner, A. Chan, R. Wiest, and M. Reyes, “Nabla-net: a deep dag-like convolutional architecture for biomedical

- image segmentation,” in *International Workshop on Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*, Springer, 2016, pp. 119–128.
- [34] E. Roura, M. Cabezas, S. Valverde, S. González-Villa, J. Salvi, A. Oliver, and X. Lladó, “Unsupervised multiple sclerosis lesion detection and segmentation using rules and level sets,” *MSSEG Challenge Proceedings: Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure*, p. 51, 2016.
- [35] S. Valverde, M. Cabezas, E. Roura, S. González-Villa, J. Salvi, A. Oliver, and X. Lladó, “Multiple sclerosis lesion detection and segmentation using a convolutional neural network of 3d patches,” *MSSEG Challenge Proceedings: Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure*, p. 75, 2016.
- [36] F. Vera-Olmos, H. Melero, and N. Malpica, “Random forest for multiple sclerosis lesion segmentation,” *Proceedings of the 1st MICCAI Challenge on Multiple Sclerosis Lesions Segmentation Challenge Using a Data Management and Processing Infrastructure-MICCAI-MSSEG*, pp. 81–86, 2016.
- [37] O. Ronneberger, P. Fischer, and T. Brox, “U-net: convolutional networks for biomedical image segmentation,” in *MICCAI*, ser. LNCS, (available on arXiv:1505.04597 [cs.CV]), vol. 9351, Springer, 2015, pp. 234–241. [Online]. Available: <http://lmb.informatik.uni-freiburg.de/Publications/2015/RFB15a>.
- [38] H. Zhang and I. Oguz, “Multiple sclerosis lesion segmentation—a survey of supervised cnn-based methods,” *arXiv preprint arXiv:2012.08317*, 2020.
- [39] H. Zhang, A. M. Valcarcel, R. Bakshi, R. Chu, F. Bagnato, R. T. Shinohara, K. Hett, and I. Oguz, “Multiple sclerosis lesion segmentation with tiramisu and 2.5 d stacked slices,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2019, pp. 338–346.
- [40] F. Isensee, P. F. Jaeger, S. A. Kohl, J. Petersen, and K. H. Maier-Hein, “Nnu-net: a self-configuring method for deep learning-based biomedical image segmentation,” *Nature Methods*, pp. 1–9, 2020.
- [41] G. Kang, B. Hou, Y. Ma, F. Labeau, Z. Su, *et al.*, “Acu-net: a 3d attention context u-net for multiple sclerosis lesion segmentation,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2020, pp. 1384–1388.
- [42] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, “Understanding deep learning requires rethinking generalization,” *arXiv preprint arXiv:1611.03530*, 2016.

- [43] Y. Feng, H. Pan, C. H. Meyer, and X. Feng, “A self-adaptive network for multiple sclerosis lesion segmentation from multi-contrast mri with various imaging protocols,” *CoRR*, vol. abs/1811.07491, 2018.
- [44] N. J. Tustison, B. B. Avants, P. A. Cook, Y. Zheng, A. Egan, P. A. Yushkevich, and J. C. Gee, “N4itk: improved n3 bias correction,” *IEEE Transactions on Medical Imaging*, vol. 29, no. 6, pp. 1310–1320, Jun. 2010.
- [45] V. S. Fonov, A. C. Evans, R. C. McKinstry, C. Almlı, and D. Collins, “Unbiased nonlinear average age-appropriate brain templates from birth to adulthood,” *NeuroImage*, no. 47, S102, 2009.
- [46] N. Shiee, P.-L. Bazin, J. L. Cuzzocreo, C. Ye, B. Kishore, A. Carass, P. A. Calabresi, D. S. Reich, J. L. Prince, and D. L. Pham, “Reconstruction of the human cerebral cortex robust to white matter lesions: method and validation,” *Human brain mapping*, vol. 35, no. 7, pp. 3385–3401, 2014.
- [47] A. Carass, J. Cuzzocreo, M. B. Wheeler, P.-L. Bazin, S. M. Resnick, and J. L. Prince, “Simple paradigm for extra-cerebral tissue removal: algorithm and analysis,” *NeuroImage*, vol. 56, no. 4, pp. 1982–1992, 2011.
- [48] L. G. Nyúl, J. K. Udupa, and X. Zhang, “New variants of a method of mri scale standardization,” *IEEE transactions on medical imaging*, vol. 19, no. 2, pp. 143–150, 2000.
- [49] S. Roy, J. A. Butman, D. S. Reich, P. A. Calabresi, and D. L. Pham, “Multiple sclerosis lesion segmentation from brain mri via fully convolutional neural networks,” *arXiv preprint arXiv:1803.09172*, 2018.
- [50] S. Valverde, M. Salem, M. Cabezas, D. Pareto, J. C. Vilanova, L. Ramió-Torrentà, A. Rovira, J. Salvi, A. Oliver, and X. Lladó, “One-shot domain adaptation in multiple sclerosis lesion segmentation using convolutional neural networks,” *NeuroImage. Clinical*, p. 101 638, 2018.
- [51] R. McKinley, R. Wepfer, F. Aschwanden, L. Grunder, R. Muri, C. Rummel, R. Verma, C. Weisstanner, M. Reyes, A. Salmen, *et al.*, “Simultaneous lesion and neuroanatomy segmentation in multiple sclerosis using deep neural networks,” *arXiv preprint arXiv:1901.07419*, 2019.
- [52] T. Brosch, L. Y. W. Tang, Y. Yoo, D. K. B. Li, A. Traboulsee, and R. Tam, “Deep 3d convolutional encoder networks with shortcuts for multiscale feature integration applied to multiple sclerosis lesion segmentation,” *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1229–1239, May 2016.
- [53] M. D. Zeiler and R. Fergus, “Stochastic pooling for regularization of deep convolutional neural networks,” *arXiv preprint arXiv:1301.3557*, 2013.

- [54] J. Nagi, F. Ducatelle, G. A. Di Caro, D. Cireşan, U. Meier, A. Giusti, F. Nagi, J. Schmidhuber, and L. M. Gambardella, “Max-pooling convolutional neural networks for vision-based hand gesture recognition,” in *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, IEEE, 2011, pp. 342–347.
- [55] F. Isensee, P. Kickingereder, W. Wick, M. Bendszus, and K. H. Maier-Hein, “Brain tumor segmentation and radiomics survival prediction: contribution to the brats 2017 challenge,” in *International MICCAI Brainlesion Workshop*, Springer, 2017, pp. 287–297.
- [56] M. Drozdal, E. Vorontsov, G. Chartrand, S. Kadoury, and C. Pal, “The importance of skip connections in biomedical image segmentation,” in *Deep Learning and Data Labeling for Medical Applications*, Springer, 2016, pp. 179–187.
- [57] S. Zagoruyko and N. Komodakis, “Wide residual networks,” *arXiv preprint arXiv:1605.07146*, 2016.
- [58] B. Graham, “Fractional max-pooling,” *arXiv preprint arXiv:1412.6071*, 2014.
- [59] Y. Bengio, Y. LeCun, *et al.*, “Scaling learning algorithms towards ai,” *Large-scale kernel machines*, vol. 34, no. 5, pp. 1–41, 2007.
- [60] C. Zhang, Y. Song, S. Liu, S. Lill, C. Wang, Z. Tang, Y. You, Y. Gao, A. Klis-toner, M. Barnett, *et al.*, “Ms-gan: gan-based semantic segmentation of multiple sclerosis lesions in brain magnetic resonance imaging,” in *2018 Digital Image Computing: Techniques and Applications (DICTA)*, IEEE, 2018, pp. 1–8.
- [61] S. Aslani, M. Dayan, L. Storelli, M. Filippi, V. Murino, M. A. Rocca, and D. Sona, “Multi-branch convolutional neural network for multiple sclerosis lesion segmentation,” *NeuroImage*, vol. 196, pp. 1–15, 2019.
- [62] B. Hou, G. Kang, X. Xu, and C. Hu, “Cross attention densely connected networks for multiple sclerosis lesion segmentation,” in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, IEEE, 2019, pp. 2356–2361.
- [63] S. R. Hashemi, S. S. M. Salehi, D. Erdogmus, S. P. Prabhu, S. K. Warfield, and A. Gholipour, “Asymmetric loss functions and deep densely-connected networks for highly-imbalanced medical image segmentation: application to multiple sclerosis lesion detection,” *IEEE Access*, vol. 7, pp. 1721–1735, 2019.
- [64] S. Andermatt, S. Pezold, and P. C. Cattin, “Automated segmentation of multiple sclerosis lesions using multi-dimensional gated recurrent units,” in *International MICCAI Brainlesion Workshop*, Springer, 2017, pp. 31–42.

- [65] S. Valverde, M. Cabezas, E. Roura, S. González-Villà, D. Pareto, J. C. Vilanova, L. Ramió-Torrentà, À. Rovira, A. Oliver, and X. Lladó, “Improving automated multiple sclerosis lesion segmentation with a cascaded 3d convolutional neural network approach,” *NeuroImage*, vol. 155, pp. 159–168, 2017.
- [66] O. Maier and H. Handels, “Ms lesion segmentation in mri with random forests,” *Proc. 2015 Longitudinal Multiple Sclerosis Lesion Segmentation Challenge*, pp. 1–2, 2015.
- [67] A. Birenbaum and H. Greenspan, “Longitudinal multiple sclerosis lesion segmentation using multi-view convolutional neural networks,” in *Deep Learning and Data Labeling for Medical Applications*, Springer, 2016, pp. 58–67.
- [68] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, “Unet++: redesigning skip connections to exploit multiscale features in image segmentation,” *IEEE transactions on medical imaging*, vol. 39, no. 6, pp. 1856–1867, 2019.
- [69] G. Song and W. Chai, “Collaborative learning for deep neural networks,” *Advances in neural information processing systems*, vol. 31, 2018.
- [70] M. Tintore, À. Rovira, J. Rio, S. Otero-Romero, G. Arrambide, C. Tur, M. Comabella, C. Nos, M. J. Arévalo, L. Negrotto, *et al.*, “Defining high, medium and low impact prognostic factors for developing multiple sclerosis,” *Brain*, vol. 138, no. 7, pp. 1863–1874, 2015.
- [71] R. McKinley, R. Wepfer, L. Grunder, F. Aschwanden, T. Fischer, C. Friedli, R. Muri, C. Rummel, R. Verma, C. Weisstanner, *et al.*, “Automatic detection of lesion load change in multiple sclerosis using convolutional neural networks with segmentation confidence,” *NeuroImage: Clinical*, vol. 25, p. 102 104, 2020.
- [72] O. Commowick, F. Cervenansky, F. Cotton, and M. Dojat, “Msseg-2 challenge proceedings: multiple sclerosis new lesions segmentation challenge using a data management and processing infrastructure,” in *MICCAI 2021-24th International Conference on Medical Image Computing and Computer Assisted Intervention*, 2021, pp. 1–118.
- [73] B. Moraal, D. S. Meier, P. A. Poppe, J. J. Geurts, H. Vrenken, W. M. Jonker, D. L. Knol, R. A. van Schijndel, P. J. Pouwels, C. Pohl, *et al.*, “Subtraction mr images in a multiple sclerosis multicenter clinical trial setting,” *Radiology*, vol. 250, no. 2, pp. 506–514, 2009.
- [74] X. Lladó, A. Oliver, M. Cabezas, J. Freixenet, J. C. Vilanova, A. Quiles, L. Valls, L. Ramió-Torrentà, and À. Rovira, “Segmentation of multiple sclerosis lesions in brain mri: a review of automated approaches,” *Information Sciences*, vol. 186, no. 1, pp. 164–185, 2012.

- [75] M. Salem, M. Cabezas, S. Valverde, D. Pareto, A. Oliver, J. Salvi, À. Rovira, and X. Lladó, “A supervised framework with intensity subtraction and deformation field features for the detection of new t2-w lesions in multiple sclerosis,” *NeuroImage: Clinical*, vol. 17, pp. 607–615, 2018.
- [76] P. Schmidt, V. Pongratz, P. Küster, D. Meier, J. Wuerfel, C. Lukas, B. Bellenberg, F. Zipp, S. Groppa, P. G. Sämann, *et al.*, “Automated segmentation of changes in flair-hyperintense white matter lesions in multiple sclerosis on serial magnetic resonance imaging,” *NeuroImage: Clinical*, vol. 23, p. 101 849, 2019.
- [77] P. Schmidt, C. Gaser, M. Arsic, D. Buck, A. Förchler, A. Berthele, M. Hoshi, R. Ilg, V. J. Schmid, C. Zimmer, *et al.*, “An automated tool for detection of flair-hyperintense white-matter lesions in multiple sclerosis,” *Neuroimage*, vol. 59, no. 4, pp. 3774–3783, 2012.
- [78] N. Gessert, M. Bengs, J. Krüger, R. Opfer, A.-C. Ostwaldt, P. Manogaran, S. Schippling, and A. Schlaefer, “4d deep learning for multiple sclerosis lesion activity segmentation,” *arXiv preprint arXiv:2004.09216*, 2020.
- [79] D. P. Kingma and J. Ba, “Adam: a method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [80] O. Commowick, A. Masson, B. Combes, S. Camarasu-Pop, F. Cervenansky, M. Kain, S. Lion, R. Casey, M. Dojat, and F. Cotton, *Miccai 2021 msseg-2 challenge quantitative results*, Oct. 2021. [Online]. Available: <https://doi.org/10.5281/zenodo.5775523>.
- [81] H. Zhang, H. Li, and I. Oguz, “Segmentation of new ms lesions with tiramisu and 2.5 d stacked slices,” *MSSEG-2 challenge proceedings: Multiple sclerosis new lesions segmentation challenge using a data management and processing infrastructure*, p. 61, 2021.
- [82] T. Dalbis, T. Fritz, J. Grilo, S. Hitziger, and W. X. Ling, “Triplanar u-net with orientation aggregation for new lesions segmentation,” *MSSEG-2 challenge proceedings: Multiple sclerosis new lesions segmentation challenge using a data management and processing infrastructure*, p. 57, 2021.
- [83] R. A. Kamraoui, V.-T. Ta, J. V. Manjon, and P. Coupé, “Image quality data augmentation for new ms lesion segmentation,” *MSSEG-2 challenge proceedings: Multiple sclerosis new lesions segmentation challenge using a data management and processing infrastructure*, p. 37, 2021.
- [84] A. Masson, B. Le Bon, A. Kerbrat, G. Edan, F. Galassi, and B. Combes, “A nnunet implementation of new lesions segmentation from serial flair images of ms patients,” *MSSEG-2 challenge proceedings: Multiple sclerosis new lesions*

- segmentation challenge using a data management and processing infrastructure*, p. 5, 2021.
- [85] A. Fenneteau, P. Bourdon, D. Helbert, C. Fernandez-Maloin, C. Habas, and R. Guillemin, "Learning a CNN on multiple sclerosis lesion segmentation with self-supervision," in *3D Measurement and Data Processing, IS&T Electronic Imaging 2020 Symposium*, Best paper award 3D Measurement and Data Processing, San Francisco, United States, Jan. 2020. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-02378210>.
- [86] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1422–1430.
- [87] S. Denner, A. Khakzar, M. Sajid, M. Saleh, Z. Spiclin, S. T. Kim, and N. Navab, "Spatio-temporal learning from longitudinal data for multiple sclerosis lesion segmentation," *arXiv preprint arXiv:2004.03675*, 2020.
- [88] M.-S. To, I. G. Sarno, C. Chong, M. Jenkinson, and G. Carneiro, "Self-supervised lesion change detection and localisation in longitudinal multiple sclerosis brain imaging," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2021, pp. 670–680.
- [89] Ž. Lesjak, A. Galimzianova, A. Koren, M. Lukin, F. Pernuš, B. Likar, and Ž. Špiclin, "A novel public mr image dataset of multiple sclerosis patients with lesion segmentations based on multi-rater consensus," *Neuroinformatics*, vol. 16, no. 1, pp. 51–63, Jan. 2018. [Online]. Available: <https://doi.org/10.1007/s12021-017-9348-7>.
- [90] D. S. Marcus, A. F. Fotenos, J. G. Csernansky, J. C. Morris, and R. L. Buckner, "Open access series of imaging studies: longitudinal mri data in nondemented and demented older adults," *Journal of cognitive neuroscience*, vol. 22, no. 12, pp. 2677–2684, 2010.
- [91] S. Bakas, H. Akbari, A. Sotiras, M. Bilello, M. Rozycki, J. S. Kirby, J. B. Freymann, K. Farahani, and C. Davatzikos, "Advancing the cancer genome atlas glioma mri collections with expert segmentation labels and radiomic features," *Scientific data*, vol. 4, p. 170 117, 2017.
- [92] M. Jenkinson, P. Bannister, M. Brady, and S. Smith, "Improved optimization for the robust and accurate linear registration and motion correction of brain images," *NeuroImage*, vol. 17, no. 2, pp. 825–841, 2002. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1053811902911328>.

- [93] S. M. Smith, "Fast robust automated brain extraction," *Human brain mapping*, vol. 17, no. 3, pp. 143–155, 2002.
- [94] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" In *Advances in neural information processing systems*, 2014, pp. 3320–3328.
- [95] J. E. Iglesias, C. Liu, P. M. Thompson, and Z. Tu, "Robust brain extraction across datasets and comparison with publicly available methods," *IEEE Transactions on Medical Imaging*, vol. 30, no. 9, pp. 1617–1634, Sep. 2011.
- [96] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine Learning*, vol. 109, no. 2, pp. 373–440, 2020.
- [97] L. Chen, P. Bentley, K. Mori, K. Misawa, M. Fujiwara, and D. Rueckert, "Self-supervised learning for medical image analysis using image context restoration," *Medical image analysis*, vol. 58, p. 101 539, 2019.
- [98] C. Baur, S. Albarqouni, and N. Navab, "Semi-supervised deep learning for fully convolutional networks," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2017, pp. 311–319.
- [99] W. Bai, O. Oktay, M. Sinclair, H. Suzuki, M. Rajchl, G. Tarroni, B. Glocker, A. King, P. M. Matthews, and D. Rueckert, "Semi-supervised learning for network-based cardiac mr image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2017, pp. 253–260.
- [100] D. Nie, Y. Gao, L. Wang, and D. Shen, "Asdnet: attention based semi-supervised deep networks for medical image segmentation," in *International conference on medical image computing and computer-assisted intervention*, Springer, 2018, pp. 370–378.
- [101] T. Nair, D. Precup, D. L. Arnold, and T. Arbel, "Exploring uncertainty measures in deep networks for multiple sclerosis lesion detection and segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2018, pp. 655–663.
- [102] M. J. Fartaria, A. Todea, T. Kober, K. O'brien, G. Krueger, R. Meuli, C. Granziera, A. Roche, and M. B. Cuadra, "Partial volume-aware assessment of multiple sclerosis lesions," *NeuroImage: Clinical*, vol. 18, pp. 245–253, 2018.
- [103] H. Zhang, J. Zhang, C. Li, E. M. Sweeney, P. Spincemaille, T. D. Nguyen, S. A. Gauthier, and Y. Wang, "All-net: anatomical information lesion-wise loss function integrated into neural network for multiple sclerosis lesion segmentation," *NeuroImage: Clinical*, p. 102 854, 2021.

- [104] R. A. Kamraoui, V.-T. Ta, J. V. Manjon, and P. Coupé, “New ms lesion segmentation with lesion-wise metrics learning,” *MSSEG-2 challenge proceedings: Multiple sclerosis new lesions segmentation challenge using a data management and processing infrastructure*, p. 29, 2021.

Segmentation automatique des lésions de sclérose en plaques en adéquation avec les contraintes cliniques.

Résumé : L'accessibilité d'un outil performant de segmentation des lésions de sclérose en plaques permet de fournir aux radiologues des métriques fiables et reproductibles vers une meilleure prise en charge des malades atteints. Pour rendre plus accessible ce genre d'outil en clinique, nous avons proposé des architectures de réseaux de neurones convolutifs légères et performantes capables d'apprendre sur des stations de travail abordables avec un nombre réduit d'exemples d'apprentissage, en un temps réduit, tout en limitant le risque de surapprentissage. Nous avons mis en place des techniques dans le but de réduire au minimum l'apport en données d'entraînement avec l'autoapprentissage et l'apprentissage semi-supervisé, tout en tenant compte de la qualité des données pour nous apercevoir qu'il suffisait, finalement, de très peu d'exams annotés. Nous présentons aussi, une méthode pour augmenter le nombre de petites lésions détectées qui sont plus difficiles à segmenter ainsi que plus susceptibles d'être omises par le radiologue. Le travail de thèse s'inscrit dans une démarche de recherche pour tirer au mieux parti de la segmentation automatique des lésions pour le radiologue vers une meilleure adoption de tels outils en routine clinique.

Mots clés : CNN, Segmentation, IRM, Sclérose en plaques, Semi-supervisé, Auto-supervisé, Détection

Automatic multiple sclerosis lesion segmentation with clinical constraints.

Abstract: The availability of a powerful tool for the segmentation of multiple sclerosis lesions makes it possible to provide radiologists with reliable and reproducible metrics for better care of affected patients. To make this kind of tool more accessible in the clinic, we proposed lightweight and powerful convolutional neural network architectures capable of learning on affordable workstations with a reduced number of training examples, in a reduced time, while limiting the risk of overfitting. We have implemented techniques to minimize the training data with self-supervision and semi-supervised learning, while taking into account the quality of the data source, to realize that very few annotated exams are needed. We also propose, in this manuscript, a method to increase the number of small lesions detected which are more difficult to segment and more likely to be missed by the radiologist. The thesis work is part of a research approach to get the best out of automatic lesion segmentation for the radiologist towards a better adoption of such tools in clinical routine.

Keywords: CNN, Segmentation, MRI, Multiple sclerosis, Semi-supervised, Self-supervised, Detection

Univ. de Poitiers, CNRS, XLIM, UMR 7252, Poitiers, France