Normalisation réaliste d'images médicales pour de la segmentation multi-domaine

par

Pierre-Luc DELISLE

MÉMOIRE PRÉSENTÉ À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
COMME EXIGENCE PARTIELLE À L'OBTENTION DE LA MAÎTRISE
AVEC MÉMOIRE EN CONCENTRATION GÉNIE DES TECHNOLOGIES
DE L'INFORMATION
M. Sc. A.

MONTRÉAL, LE 3 MAI 2021

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

PRÉSENTATION DU JURY

CE MÉMOIRE A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE:

M. Hervé Lombaert, Directeur de mémoire
Département de génie logiciel et des technologies de l'information à l'École de technologie supérieure

M. Christian Desrosiers, co-Directeur
Département de génie logiciel et des technologies de l'information à l'École de technologie supérieure

M. Jose Dolz, Président du jury
Département de génie logiciel et des technologies de l'information à l'École de technologie supérieure

M. Carlos Vázquez, Membre du jury
Département de génie logiciel et des technologies de l'information à l'École de technologie supérieure

IL A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE 12 AVRIL 2021

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

REMERCIEMENTS

Au moment d'écrire ces dernières lignes de mon mémoire de recherche, une certitude m'apparait. Je ne peux passer sous silence le fait que plusieurs personnes ont contribué à cet accomplissement.

Tout d'abord, je voudrais remercier sincèrement mon directeur de recherche, le professeur Hervé Lombaert, pour sa passion à transmettre ses connaissances en recherche, ses conseils judicieux qui ont permis d'alimenter ma réflexion et pour sa confiance à mon égard. Je le remercie pour son souci du détail, ce qui m'a permis de grandir tout au long de mon parcours académique.

Par la suite, j'aimerais remercier chaleureusement mon codirecteur de recherche, le professeur Christian Desrosiers, pour son aide précieuse à la rédaction, sa disponibilité et pour son assistance tout au long de mon cheminement académique. Un merci tout particulier pour son support moral et pour sa supervision éclairée tout au long de la rédaction des articles qui composent ce mémoire.

Je voudrais exprimer ma gratitude à tous mes collègues du laboratoire de recherche pour leur support et les échanges enrichissants. Un merci tout particulier à mon collègue Benoit Anctil-Robitaille pour ses précieux conseils, son aide, ses réflexions et sa positivité contagieuse. Je n'aurais pas pu espérer avoir un meilleur collègue, maintenant devenu un grand ami. Je suis, grâce à lui, un bien meilleur ingénieur logiciel et je retire beaucoup d'expérience de nos travaux communs.

Je voudrais exprimer ma reconnaissance à mes parents pour leur soutien émotionnel et financier qui m'a permis de réaliser les études dont je voulais. Ils m'ont supporté et ont cru en moi tout au long de ce parcours. J'aimerais remercier tout particulièrement, ma conjointe Geneviève Lizotte-Dubé pour son amour, sa patience et pour son soutien à travers toutes ces années d'étude.

Je remercie toute ma famille et mes amis de m'avoir encouragé et accompagné dans mon parcours académique.

À tous les membres du jury, je vous offre mes remerciements les plus sincères.

Normalisation réaliste d'images médicales pour de la segmentation multi-domaine

Pierre-Luc DELISLE

RÉSUMÉ

Les réseaux de neurones profonds sont au cœur de l'actualité scientifique depuis maintenant plusieurs années. Les progrès accomplis dans le domaine de la recherche fondamentale se sont transposés dans des applications dans le domaine médical. La classification automatique des voxels dans une image médicale à l'aide d'un réseau de neurones convolutif permet entre autres de suivre le développement d'un organe ou l'évolution d'une maladie.

L'accessibilité des données d'imagerie médicale à des fins d'entraînement d'algorithmes d'apprentissage automatique est une principale limitation inhérente au domaine. En raison du coût et de l'expertise requis afin d'annoter des ensembles de données, ces derniers ne sont généralement composés que de quelques images. Cela a un impact direct sur la capacité de généralisation des algorithmes d'apprentissage. Un des moyens de remédier à cette disponibilité limitée se trouve dans la normalisation d'ensembles de données issus de plusieurs domaines d'imagerie. Un domaine d'imagerie peut être un site où l'image est acquise, un protocole d'imagerie ou encore le type d'équipement servant à l'acquisition de l'image médicale. Les approches conventionnelles de normalisation couramment utilisées sont basées sur la normalisation en fonction d'un seul ensemble de données. Toutefois, cette stratégie ne peut exploiter l'information structurelle et l'information intrinsèque des échelles d'intensités aux images issues de plusieurs ensembles de données. Par conséquent, ignorer ces informations entraîne une performance suboptimale pour un algorithme de segmentation. La correction des intensités des images par un processus de normalisation intelligent qui aligne les distributions des tissus sur une échelle commune entre les ensembles de données permettrait l'apprentissage d'une tâche à partir de plusieurs ensembles de données différents. Cela aurait pour conséquences d'améliorer la disponibilité des données et la capacité de généralisation des algorithmes d'apprentissage.

Ce travail propose une nouvelle méthode de normalisation conjointe à la segmentation. Cette méthode est apprise à l'aide d'une architecture orientée à la fois par les données et par une tâche. Dans notre méthode, trois réseaux s'entraînent simultanément afin d'apprendre la relation entre une image d'entrée issue d'un domaine d'imagerie quelconque et une sortie normalisée. La fonction de transfert optimale liant les distributions d'intensités des images à travers les différents jeux de données est ainsi apprise. Elle a pour but de produire non seulement une carte de segmentation précise, mais également des images intermédiaires normalisées médicalement réalistes et interprétables. Les images sont dites réalistes puisque les changements apportés à l'image n'affectent ni les structures ni sa capacité à être interprétée par un clinicien. Ces caractéristiques importantes recherchées entre autres afin de rendre compréhensible l'image intermédiaire produite, sont rendues possibles par une architecture antagoniste pourvue d'un générateur d'images normalisées et un discriminateur de domaines. Il s'agit d'une architecture simple et évolutive en fonction du nombre de domaines d'imagerie.

La performance de notre méthode est évaluée à l'aide de trois ensembles de données composés de sujets enfants et adultes, soit iSEG, MRBrainS et ABIDE. Les résultats révèlent le potentiel de notre méthode de normalisation pour une tâche de segmentation avec une amélioration de l'indice Dice allant jusqu'à 57,5% par rapport à nos résultats de base consistant en l'apprentissage et le test d'un algorithme de segmentation sur des ensembles de données issus de domaines différents. Notre méthode permet de s'attaquer directement au problème de disponibilité des données en permettant l'apprentissage simultané sur plusieurs domaines d'imagerie différents.

Mots-clés: IRM 3D, segmentation du cerveau, réseau de neurones antagonistes, normalisation d'intensités

Realistic Image Normalization for Multi-Domain Segmentation

Pierre-Luc DELISLE

ABSTRACT

Deep neural networks have repeatedly made the headlines of scientific news in recent years. The progress made in fundamental research in the last decades have now been transposed into applications in the medical domain. The automatic classification of voxels in medical images using Convolutional Neural Networks (CNNs) allows, among other things, to follow the development of an organ or the evolution of a disease.

Availability of medical imaging data for training learning algorithms have been an inherent limitation of the domain. The high cost and expertise required to annotate images to produce the segmentation ground truth are facts making datasets small, often composed of only a few images. This directly impacts the generalization capability of learning algorithms. A way to alleviate this limited availability is in image normalization accross multiple imaging domains. An imaging domain can be the site where the image is acquired, an imaging protocol or the type of machinery used to acquire the image. Conventional approaches are customarily utilized on a per-dataset basis. This strategy, however, prevents the current normalization algorithms from fully exploiting the intrinsic joint structual information and intensity scales available across multiple datasets. Consequently, ignoring such joint information has a direct impact on the performance of segmentation algorithms. The correction of the image's intensities by an intelligent normalization process that aligns the tissues' distributions on a common scale could allow the learning of a specific task over many different datasets. This could have for consquence to improve the data availability and the generalization capacity of learning algorithms.

This paper proposes to revisit the conventional image normalization approach by instead learning a common normalizing function across multiple datasets. Jointly normalizing multiple datasets is shown to yield consistent normalized images as well as an improved image segmentation. In our method, three networks are trained simultaneously to learn the relation between an image of any domain and a normalized output. The optimal transfer function binding the image intensities across the different datasets is learned. This function has for goal to output a precise segmentation map and realistic normalized intermediate image. The images are said realistic because the changes made by the function doesn't affect neither the structures nor the capacity of the image to be interpreted by a clinician. These important characteristics sought in order to make the intermediate image produced understandable are made possible by using an adversarial architecture. It's a simple, yet scalable architecture which complexity doesn't increase when increasing the number of imaging domains.

We evaluated the performance of our normalizer on both infant and adult brain images from the iSEG, MRBrainS and ABIDE datasets. Results reveal the potential of our normalization approach for segmentation, with Dice improvements of up to 57.5% over our baseline which consists of training and testing a segmentation algorithm using different datasets. Our method

can also enhance data availability by increasing the number of samples available when learning from multiple imaging domains.

Keywords: 3D MRI, brain segmentation, generative adversarial networks, intensity normalization

TABLE DES MATIÈRES

Page

INTRODUCTION 1

CHAPITRE 1 REVUE DE LA LITTÉRATURE 7
1.1 Imagerie par résonance magnétique 7
1.1.1 Imagerie structurelle et principes de la résonance magnétique 9
1.1.2 Normalisation d'images médicales 10
1.1.2.1 Aucune normalisation 12
1.1.2.2 Mise à l'échelle 13
1.1.2.3 Normalisation gaussienne (*standardization*) 13
1.1.2.4 Normalisation quantile 14
1.1.2.5 Concordance d'histogrammes (*histogram matching*) 14
1.1.2.6 Méthodes apprises (*learned methods)* 15
1.2 Apprentissage automatique profond et réseaux de neurones convolutifs 16
1.2.1 Opération de convolution 18
1.2.2 Couches d'activations 20
1.2.2.1 Sigmoïde 20
1.2.2.2 ReLU 21
1.2.3 Échantillonnage maximum 22
1.2.4 Couche pleinement connectée 23
1.2.5 Optimisation par descente de gradient et rétropropagation du gradient 24
1.2.6 Régularisation et normalisation en lot *(Batch Norm)* 24
1.2.7 Initialisation des poids 25
1.2.8 Réseaux de neurones pleinement convolutionnels 26
1.2.9 Architecture U-Net 27
1.2.10 Architecture ResNet 29
1.3 Réseaux génératifs antagonistes 30
1.4 Apprentissage profond appliqué à la segmentation d'images médicales 33
1.4.1 Contraintes d'entraînement et problématiques liées au domaine médical 35
1.4.1.1 Contrainte de mémoire et décomposition en segments (*patches*) 35
1.4.1.2 Problème de débalancement des classes 36
1.4.1.3 Problème de manque de données 37
1.4.2 Adaptation de domaine 38
1.4.3 Harmonisation de données 39

CHAPITRE 2 NORMALISATION RÉALISTE D'IMAGES POUR DE LA SEGMENTATION MULTI-DOMAINES 43
2.1 Motivation 43

2.2 Méthodologie 44
2.2.1 Proposition 44
2.2.2 Entraînement antagoniste 49
2.2.3 Stratégie d'apprentissage 51
2.2.3.1 Taux d'apprentissage 51
2.2.3.2 Augmentation des données 52

CHAPITRE 3 EXPÉRIMENTATIONS 53
3.1 Données 53
3.1.1 Ensembles de données 53
3.1.1.1 iSEG 53
3.1.1.2 MRBrainS 54
3.1.1.3 ABIDE 54
3.2 Protocole d'expérimentations 55
3.2.1 Prétraitement 55
3.2.2 Détails d'implémentation 56
3.2.3 Évaluation 57
3.3 Résultats 58
3.3.1 Performance de base sur les ensembles de données 58
3.3.2 Évaluation sur deux domaines 60
3.3.2.1 Performance de segmentation 60
3.3.2.2 Performance de normalisation 62
3.3.3 Évaluation multi-site 63
3.3.4 Évaluation multimodale 64
3.3.5 Robustesse à la dégradation de l'image 65
3.3.6 Impact de l'hyperparamètre Lambda 66
3.3.7 Impact de l'architecture du discriminateur 68

CONCLUSION ET RECOMMANDATIONS 75
4.1 Résumé des contributions 75
4.2 Résultats 75
4.3 Limitations et améliorations futures 77

ANNEXE I PREUVE DU THÉORÈME 1 79

BIBLIOGRAPHIE 82

LISTE DES TABLEAUX

Page

Tableau 1.1 Temps de relaxation (T_1 et T_2) et densité protonique (SD) typiques des tissus qui composent le cerveau humain 9

Tableau 2.1 Architecture pleinement convolutive pour le réseau générateur et le réseau de segmentation 48

Tableau 2.2 Architecture DCGAN convolutive pour le réseau de classification des domaines (discriminateur) 49

Tableau 2.3 Architecture ResNet-18 convolutive pour le réseau de classification des domaines (discriminateur) 49

Tableau 3.1 Démographie des sujets présents dans l'ensemble de données ABIDE 55

Tableau 3.2 Résultats de base avec un réseau de segmentation 59

Tableau 3.3 Indice Dice en fonction de l'architecture du modèle et des données 62

Tableau 3.4 Distance de Jensen-Shannon des données brutes et normalisées par le générateur en fonction des ensembles de données 64

Tableau 3.5 Indice Dice et la distance de Hausdorff en fonction des types de tissus du cerveau et des modalités d'imagerie utilisées 65

Tableau 3.6 Coefficient de corrélation de Pearson (ρ) 67

Tableau 3.7 Distance moyenne au carré (MSE) après normalisation pour différents ratios signal-bruit 67

Tableau 3.8 Indice Dice moyen pour différentes valeurs de l'hyperparamètre λ 68

LISTE DES FIGURES

Page

Figure 1.1 Schématique d'un système d'acquisition d'images médicales par résonance magnétique 8

Figure 1.2 Visualisation d'une image pondérée en T_1 acquise par résonance magnétique 11

Figure 1.3 Graphe de la fonction Sigmoïde 20

Figure 1.4 Graphe de la fonction ReLU 21

Figure 1.5 Schéma de l'échantillonnage maximum 23

Figure 1.6 Architecture 3D U-Net 29

Figure 1.7 Architecture ResNet 30

Figure 1.8 Schéma de données pairées 33

Figure 2.1 Architecture de normalisation d'images proposée 48

Figure 3.1 Histogrammes d'intensités des différents tissus composant le cerveau en fonction de l'ensemble de données 55

Figure 3.2 Segmentation d'un cerveau sur l'ensemble de données iSEG en fonction de l'ensemble de données d'entraînement 59

Figure 3.3 Résultats de normalisation avec notre méthode 70

Figure 3.4 Visualisation du résultat de segmentation pour deux images de test 71

Figure 3.5 Histogrammes des images de tests avec contrainte de réalisme 71

Figure 3.6 Intensité moyenne d'un voxel d'une tranche axiale pour un sujet 72

Figure 3.7 Résultat d'une prédiction du générateur avec une image de test dégradée 72

Figure 3.8 Matrices de confusions normalisées pour différentes valeurs de l'hyperparamètre λ 73

Figure 3.9 Visualisation de l'évolution de la valeur des fonctions de coûts en fonction de l'époque 74

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

IRM	Imagerie par résonance magnétique
MLP	*Multi-Layer Perceptron* - Réseau de perceptrons multicouche
DNN	*Deep Neural Network* - Réseau de neurones profond
CNN	*Convolutional Neural Network* - Réseau de neurones convolutif
GPU	*Graphics Processing Unit* - Accélérateur graphique
RSB	Rapport signal bruit
CRF	*Conditional Random Field* - Champ aléatoire conditionnel

LISTE DES SYMBOLES ET UNITÉS DE MESURE

dB	Décibel
mm	millimètre
ms	milliseconde

INTRODUCTION

Devant le nombre toujours croissant d'images médicales à analyser, le développement d'algorithmes de segmentation automatique est un besoin de plus en plus criant. Selon l'Inventaire canadien d'imagerie médicale 2017 (Sinclair, Morrison, Young & P., 2018), les 336 imageurs par résonance magnétique (IRM) en fonction au Canada ont produit 1,86 million d'examens pendant l'année fiscale 2016. L'imagerie par résonance magnétique produit des images tridimensionnelles extrêmement riches en information. La plupart des études longitudinales ou transversales qui souhaitent exploiter la richesse de l'information présente dans une image médicale utilisent des séquences IRM. Cette séquence permet notamment d'analyser les différences structurelles ou fonctionnelles des organes du corps humain. Cette modalité nous aide particulièrement à comprendre le fonctionnement du cerveau humain et son évolution avec l'âge. Afin d'exploiter cette richesse et de traiter la quantité phénoménale de données produite par cette modalité d'imagerie médicale toujours plus accessible, le domaine s'est largement tourné au courant des dernières années vers les algorithmes d'apprentissage automatique. De ces algorithmes, les algorithmes d'apprentissage profond sont très populaires. Ces derniers, composés de millions de paramètres à optimiser, ont la capacité d'apprendre des caractéristiques intrinsèques aux données qui sont structurées de façon hiérarchique. Ces algorithmes constituent la clef donnant accès à l'analyse de cette richesse et à la quantité phénoménale de données d'imagerie médicale complexes et multimodales produite chaque année. Depuis la dernière décennie, ces types d'algorithmes ont permis de repousser les limites du traitement d'images médicales automatique pour des tâches complexes et très coûteuses en temps comme la segmentation d'images médicales. Cette opération manuelle est nécessaire dans plusieurs scénarios, par exemple pour suivre le développement du cerveau et des tissus qui le constitue ou encore afin de préparer une intervention chirurgicale. Elle permet d'avoir des mesures volumétriques précises d'une région d'intérêt par le praticien, mais prive également ce dernier de temps sur le terrain pour soigner ses patients.

Problématique

Toutefois, les algorithmes d'apprentissage profond sont sensibles aux données qu'on leur présente, particulièrement lorsqu'on travaille avec des données d'imagerie médicale. De plus, ces algorithmes requièrent d'énormes quantités de données étiquetées afin d'être pourvues d'un certain pouvoir statistique. Malheureusement, les jeux de données étiquetés par des professionnels hautement qualifiés dans le but de faire de la segmentation d'images sont très petits. Ceux-ci sont parfois composés seulement d'une dizaine d'images. Cela limite grandement le pouvoir statistique des études qui utilisent ces petits jeux de données. Afin d'accroître ce pouvoir statistique, on serait alors tenté de combiner des jeux de données acquis depuis différents sites sur différents imageurs. Cependant, l'imagerie par résonance magnétique n'a pas de standard d'échelle d'intensités, ce qui augmente la sensibilité des algorithmes d'apprentissage automatiques tout en réduisant grandement leur performance. Les intensités varient non seulement en raison des différences physiques des imageurs utilisés, mais également en raison des protocoles d'acquisition qui varient d'un endroit à un autre, ce qui réduit considérablement le pouvoir de généralisation des algorithmes. Les intensités peuvent également varier en fonction du stade d'une maladie. Par exemple, dans le cas de la sclérose en plaques, il est connu que les intensités peuvent varier non seulement en raison des différences entre les sites d'acquisition, mais également en raison du stade de la maladie (Shah, Xiao, Subbanna, Francis, Arnold, Collins & Arbel, 2011). Des maladies à différents stades peuvent alors affecter les intensités de certains tissus, ce qui, ajouté à l'absence d'une échelle standardisée, augmente la variance et rend la tâche de généralisation encore plus difficile pour un algorithme d'apprentissage automatique.

Approche

C'est ainsi que l'harmonisation et la normalisation des données prend tout son sens. L'harmonisation des données médicales peut se traduire par la suppression des effets liés au site d'acquisition de l'image (Pomponio, Erus, Habes, Doshi, Srinivasan, Mamourian, Bashyam, Nasrallah, Sat-

terthwaite, Fan, Launer, Masters, Maruff, Zhuo, Völzke, Johnson, Fripp, Koutsouleris, Wolf, Gur, Gur, Morris, Albert, Grabe, Resnick, Bryan, Wolk, Shinohara, Shou & Davatzikos, 2020) alors que la normalisation permet de modifier la plage de valeurs des intensités des voxels d'une image. Le défi dans les deux cas est d'adapter des domaines d'imagerie différents afin d'avoir une représentation commune, unifiée. De cette harmonisation découle un plus grand potentiel d'utilisation des données par le biais d'algorithmes d'apprentissage automatique.

Ce travail élabore une méthode qui, non seulement harmonise des données entre différents domaines d'imagerie, mais qui optimise également une représentation intermédiaire dans le but de faire une tâche précise, soit la segmentation automatique d'images médicales 3D. Des réseaux de neurones, au compte de trois, s'entraînent mutuellement afin d'obtenir une image intermédiaire à la fois optimisée et réaliste, pouvant ainsi être consultée et analysée par un clinicien. Cette méthode permet également d'apprendre la tâche de segmentation à partir de plusieurs ensembles de données en harmonisant les intensités des voxels de chaque ensemble de données. Nous affirmons ainsi que cette technique d'entraînement, en raison du potentiel accru du nombre de données pouvant être présenté aux réseaux de neurones, augmente la capacité de généralisation de ces réseaux.

Contributions

La contribution de ce mémoire peut être résumée comme suit :

1. L'élaboration de la première méthode de normalisation et de segmentation d'images médicales 3D qui produit à la fois des images optimisées pour la segmentation tout en demeurant visuellement réalistes. Les approches publiées récemment utilisent une technique d'apprentissage appelée CycleGAN qui emploie une dépendance de cycle limitée à deux domaines. Notre méthode peut avoir un nombre arbitraire de domaines d'imagerie sans ajout de complexité ;

2. Une nouvelle méthode antagoniste pour le traitement d'images médicales 3D qui optimise conjointement trois réseaux de neurones convolutifs. Contrairement aux techniques antagonistes standards qui utilisent des discriminateurs différents afin d'effectuer la classification du domaine et la classification permettant de différencier les images générées des images réelles, notre modèle combine les deux en un seul réseau de classification. Comme contribution théorique, nous montrons qu'optimiser ce modèle correspond à minimiser la différence entre la distribution des images générées et la moyenne des images des ensembles de données. En d'autres mots, notre méthode minimise la divergence de Kullback–Leibler (*KL divergence*) entre la distribution des probabilités des images générées de chaque domaine et la distribution moyenne des images réelles ;
3. Une analyse expérimentale des modèles de normalisation appris sur trois ensembles de données très différents d'imagerie par résonance magnétique. La méthode est également évaluée sur des données multimodales et sur des images dégradées par un effet de champ, un signal basse fréquence qui corrompt souvent les images IRM et décroît la qualité globale de l'image.

La méthode proposée entraîne une augmentation significative de la performance de segmentation par rapport à une méthode de segmentation utilisant un réseau de neurones conventionnel où un réseau de segmentation est entraîné avec un ensemble de données D_s et testé sur un ensemble de données D_t. À notre connaissance, ceci est le premier travail utilisant une méthode de normalisation d'images médicales purement pilotée par les données et une tâche (*task-and-data-driven*) et qui conserve l'interprétabilité de l'image intermédiaire produite tout en exploitant l'information conjointe de plusieurs ensembles de données.

Les chapitres qui suivent aborderont notamment une revue de la littérature afin de mieux cerner la problématique et les avancements dans le domaine de la normalisation et de la segmentation d'images médicales. Une présentation formelle de notre méthode sera ensuite effectuée. Les

expériences et les résultats de celles-ci démontrant la validité de notre méthode suivront. Finalement, une conclusion rappelant les travaux futurs en lien avec notre méthode clôturera ce manuscrit.

CHAPITRE 1

REVUE DE LA LITTÉRATURE

Dans ce premier chapitre, une revue de la littérature entourant le sujet de l'imagerie médicale et de l'apprentissage automatique appliqué aux données d'imagerie médicale est introduite. Les techniques d'apprentissage automatique et les composants de l'apprentissage automatique profond sont couverts. Une discussion des problèmes que posent les données médicales relatives à ce type d'apprentissage, notamment du manque de données et de la normalisation de ce type d'images, est également présentée. Les défis à relever afin de remédier à ces problématiques y sont aussi présentés.

1.1 Imagerie par résonance magnétique

L'imagerie par résonance magnétique (IRM) est une technique d'imagerie non invasive produisant des images 3D haute résolution des tissus qui composent le corps humain. Contrairement à l'imagerie par rayon X, cette technique ne dépend pas d'une source de radiation extérieure et repose plutôt sur la résonance magnétique nucléaire du noyau de l'atome d'hydrogène. Contrairement aux rayons X où l'image résultante est produite par l'absorption du signal par les tissus se situant entre la source émettrice du signal et le capteur, l'imagerie par résonance magnétique repose sur la résonance magnétique nucléaire du proton de l'hydrogène qui compose l'objet à imager. L'imagerie par résonance magnétique se veut une modalité d'imagerie permettant de représenter dans un espace tridimensionnel les propriétés nucléaires de l'atome d'hydrogène. L'atome d'hydrogène est choisi puisqu'il est le plus abondant dans le corps humain, celui-ci étant constitué d'environ 60% d'eau [1]. De plus, cette modalité d'imagerie permet une meilleure représentation des tissus mous, telle que la matière blanche ou grise du cerveau, que la modalité tomodensitométrie par rayons X puisqu'elle permet un meilleur contraste de ce type de tissus.

[1] U.S. Geological Survey. *The Water in You : Water and the Human Body*. Consulté le 26 juin 2020 au https://www.usgs.gov/special-topic/water-science-school/science/water-you-water-and-human-body?qt-science_center_objects=0#qt-science_center_objects

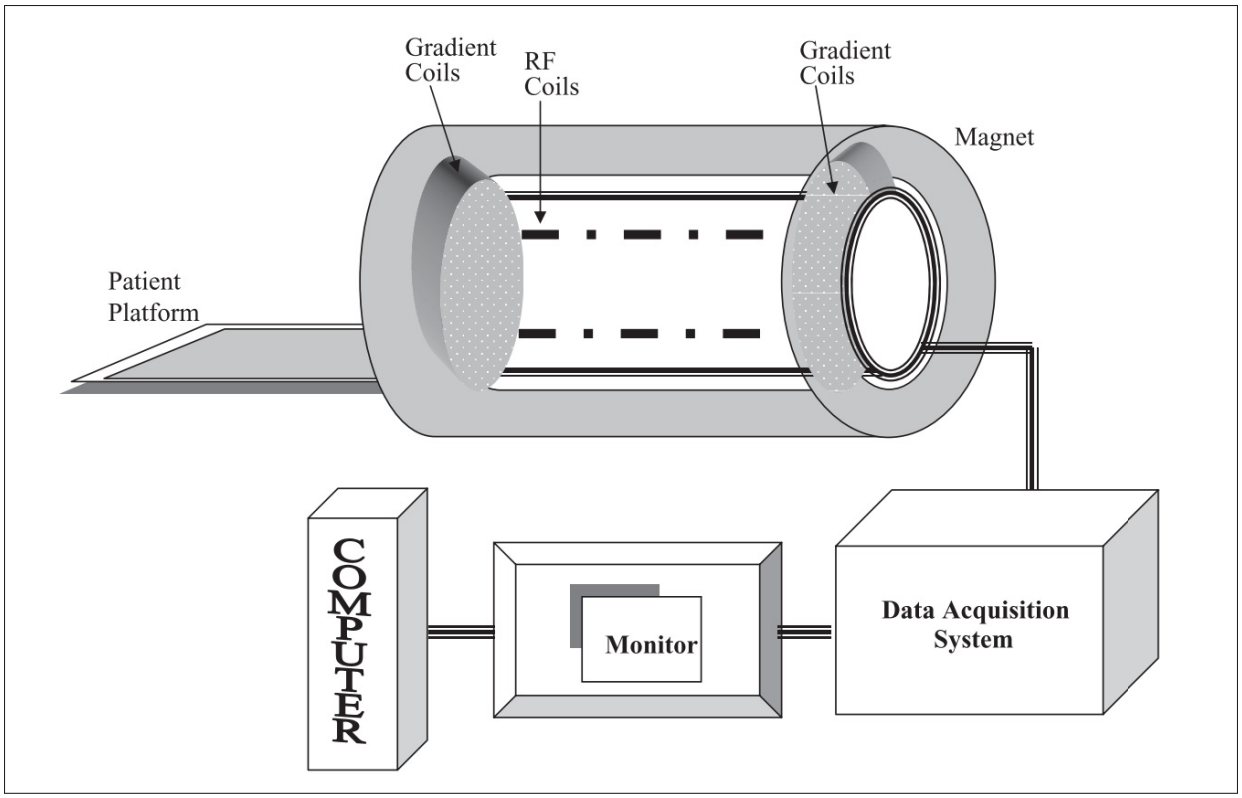

Figure 1.1 Schématique d'un système d'acquisition d'images médicales par résonance magnétique. Tirée de Atam P. Dhawan (2011)

La figure 1.1 montre un schéma de l'équipement nécessaire afin de faire l'acquisition d'une image par résonance magnétique. Le patient est d'abord inséré dans un dispositif cylindrique jusqu'à l'immersion de la portion du corps à imager. Ce cylindre permet, à l'aide d'un puissant aimant refroidi jusqu'à l'état de supraconducteur, de produire un champ électromagnétique homogène de l'ordre de 0,5, 1,5, 3 ou 7 Tesla B_0. Des bobines de gradient (*gradient coils*), généralement au compte de 3, sont disposées de manière orthogonale, permettent de produire un gradient dans le champ électromagnétique B_0. Ce mécanisme permet ainsi de sélectionner une tranche de voxels à exciter, ces derniers étant l'unité tridimensionnelle singulière d'un volume d'une grandeur physique déterminée par la résolution de l'imageur. Le gradient électromagnétique est, dans sa forme la plus simple, un gradient linéaire. Ce gradient permet de moduler légèrement la fréquence de Larmor en une tranche donnée, fréquence en laquelle un proton d'hydrogène peut recevoir une énergie sous forme de radiofréquence et changer son état afin d'exposer la

résonance magnétique nucléaire. C'est lors du retour à l'équilibre initial, soit le processus de relaxation, que le noyau de l'atome libère de l'énergie à la même fréquence qui est ensuite captée par les bobines radiofréquence de l'imageur.

Des paramètres d'imagerie peuvent être appliqués lors de l'acquisition de l'image. Celle-ci peut notamment être pondérée en T_1 ou T_2, ou représenter la densité protonique, donnant lieu à des représentations visuelles différentes de la matière échantillonnée. Ces pondérations sont directement en lien avec le temps de relaxation T_1 et T_2 de la matière qui compose l'objet imagé. Le tableau 1.1 illustre les temps de relaxation des tissus qui composent le cerveau humain lors la segmentation d'images médicales.

Tableau 1.1 Temps de relaxation (T_1 et T_2) et densité protonique (*SD*) typiques des tissus qui composent le cerveau humain. Tiré de Atam P. Dhawan (2011)

Tissu	T_1 **(ms)**	T_2 **(ms)**	**SD(%)**
Matière blanche	300	133	11.0
Matière grise	475	118	10.5
Liquides cérébraux spinaux (CSF)	2000	250	10.8

1.1.1 Imagerie structurelle et principes de la résonance magnétique

Comme mentionné précédemment, trois propriétés peuvent être considérées lors de l'acquisition d'une image par résonance magnétique. Ces trois propriétés sont le temps de relaxation longitudinale ou le temps de relaxation *spin-lattice* (T_1), le temps de relaxation transversale ou le temps de relaxation *spin-spin* (T_2) et la densité protonique ρ de l'atome d'hydrogène. Le signal reconstruit dépend aussi d'autres paramètres d'imagerie tels que le champ magnétique homogène B_0 ainsi que d'autres facteurs physiques tels que la sensibilité des bobines utilisées lors de la conception de l'imageur ainsi que leur type. De ce fait, un imageur par résonance magnétique peut produire des images multicontrastes en fonction des paramètres sélectionnés par l'opérateur lors du processus d'acquisition.

De manière générale, le contraste résultant de l'image reconstruite entre deux tissus de composition différente peut être estimé en calculant la différence respective des intensités du signal. Dans le cas d'une séquence d'impulsion *spin-écho*, l'équation du signal est :

$$S = k\rho\left(1 - e^{-\frac{T_R}{T_1}}\right)\left(e^{-\frac{T_E}{T_2}}\right) \tag{1.1}$$

où S est l'amplitude du signal dans le spectre des fréquences, k est une constante de proportionnalité qui dépend de la sensibilité de l'imageur, T_E est le temps de formation de l'écho et T_R est le temps de répétition de cycle entre l'application de deux séquences pulsées consécutives spin-écho. La manipulation de ces variables avec la pondération T_1 et T_2 permettent de créer des images avec des contrastes différents afin de mettre en évidence certains types de tissus ou pathologies (figure 1.2). Dans cette figure, la ligne du haut est une image issue de l'ensemble de données iSEG, la ligne du milieu est une image issue de l'ensemble de données MRBrainS et la ligne du bas est une image de l'ensemble de données ABIDE. On remarque que les images ont des intensités très différentes pour représenter la même matière du cerveau. De ce fait, il n'existe aucun standard universel de paramètres permettant d'avoir des intensités de voxels standards pour un tissu donné. De plus, d'autres facteurs externes comme la déformation du champ magnétique, les mouvements physiques et physiologiques, le champ de vision (*field of view value*), la résolution du voxel et l'intensité du gradient sont autant de variables qui peuvent différer d'un protocole d'acquisition à un autre. Le matériel utilisé afin de construire l'imageur peut également faire varier les intensités résultantes de la reconstruction de l'image 3D.

1.1.2 Normalisation d'images médicales

Puisque les échelles d'intensités des images médicales IRM sont exprimées dans des plages très arbitraires, la normalisation d'images médicales est une manipulation de prétraitement très importante. Malgré sa dimension fondamentale, spécialement lorsqu'on conjugue ce type d'images avec des algorithmes d'apprentissage automatique, il n'y a pas de consensus sur la manière de normaliser des images médicales. Diverses méthodes telles que la mise à l'échelle,

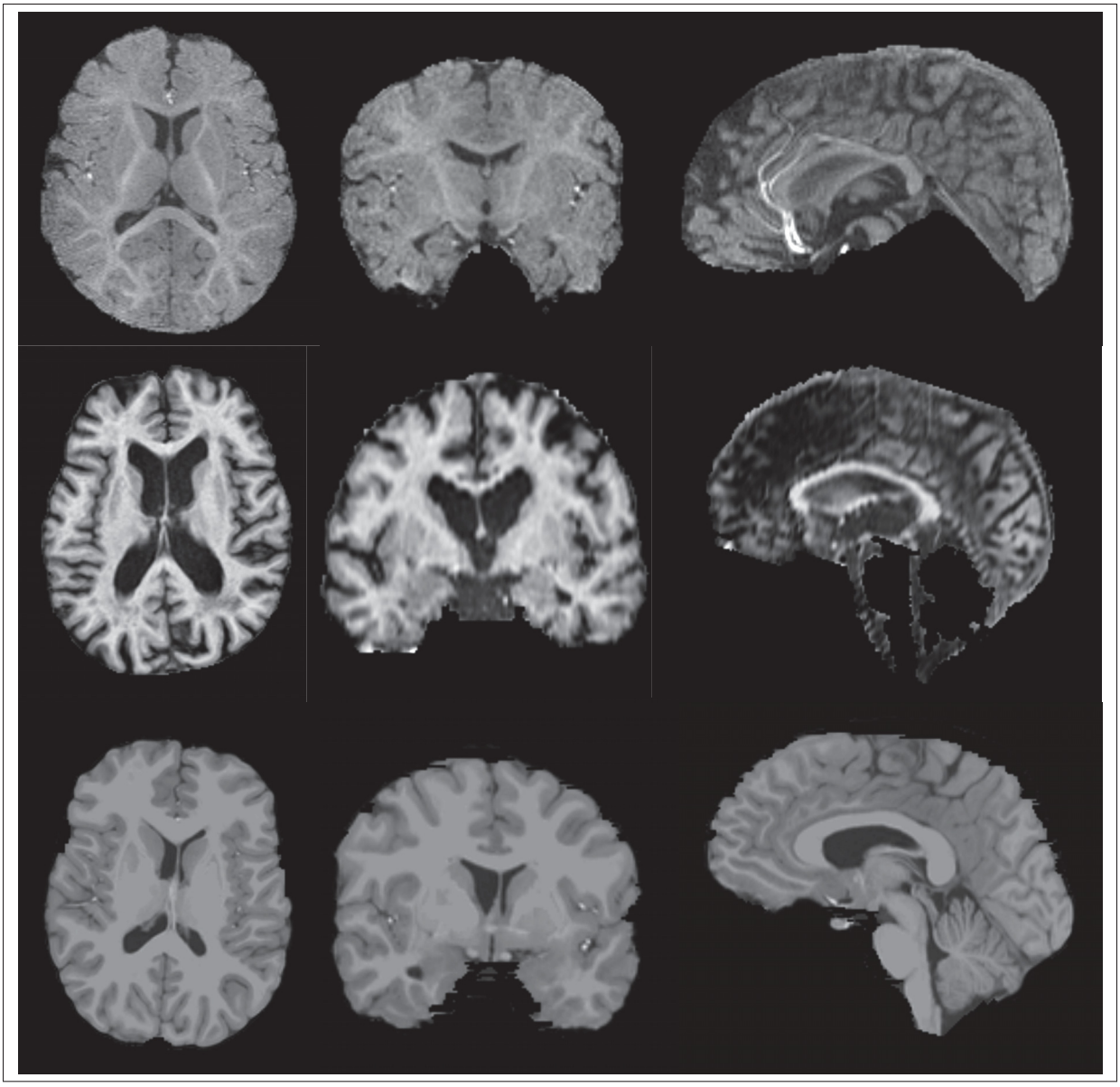

Figure 1.2 Visualisation d'une image pondérée en T_1 acquise par résonance magnétique

la normalisation gaussienne (Birenbaum & Greenspan, 2016; Kamnitsas, Ledig, Newcombe, Simpson, Kane, Menon, Rueckert & Glocker, 2017; Casamitjana, Puch, Aduriz & Vilaplana, 2016; Chen, Dou, Yu, Qin & Heng, 2018a), la normalisation quantile et la concordance d'histogrammes (Onofrey, Casetti-Dinescu, Lauritzen, Sarkar, Venkataraman, Fan, Sonn, Sprenkle, Staib & Papademetris, 2019) sont couramment utilisées dans le domaine médical à titre d'étape de prétraitement. Généralement, lorsqu'on travaille avec un seul jeu de données d'images

naturelles, retirer la moyenne globale du jeu de données suffit aux algorithmes d'apprentissage profond pour atteindre de bonnes performances de classification. Cependant, lorsqu'on est en présence d'images médicales issues de plusieurs sites d'acquisition, cette astuce n'est guère utile. Le changement d'équipement peut avoir des conséquences drastiques sur les niveaux d'intensité de gris des images reconstruites. Une méthode de normalisation parmi celles mentionnées est alors de mise afin de ramener les valeurs d'intensités dans les mêmes plages de valeurs.

Il a été défini dans Shinohara, Sweeney, Goldsmith, Shiee, Mateen, Calabresi, Jarso, Pham, Reich & Crainiceanu (2014) qu'un processus de normalisation devrait produire des unités qui :

1. ont une interprétation commune entre les locations au sein d'un même type de tissus ;
2. sont reproductibles ;
3. préserve le rang des intensités ;
4. ont une distribution similaire au sein d'un même tissu d'intérêt entre des patients différents ;
5. ne sont pas influencés par une anomalie biologique ou par l'hétérogénéité d'une population ;
6. sont minimalement sensibles aux bruits et artéfacts visuels ;
7. ne résulte pas en une perte d'information associée à une pathologie ou un phénomène médical.

[traduction libre], (Shinohara *et al.*, 2014)

1.1.2.1 Aucune normalisation

Les images brutes (*RAW*), constituent l'image de sortie de l'imageur. Elles peuvent avoir subi un prétraitement anatomique, tel que le retrait des os du crâne. Puisqu'il n'y a pas de standard des intensités de niveaux de gris, l'échelle des intensités peut aller de valeurs négatives à une infinité positive. Vu cette échelle non bornée des images dites «*RAW*», il en résulte généralement des performances moindres des algorithmes d'apprentissage automatique puisque ceux-ci ont généralement plus de difficulté à converger lorsqu'on leur présente de telles données.

1.1.2.2 Mise à l'échelle

Cette méthode permet de borner l'échelle des intensités généralement dans une plage de valeurs se situant dans l'intervalle [0, 1]. Elle n'affecte pas la forme de la distribution des intensités, mais permet de ne pas surévaluer certaines valeurs plus élevées que d'autres. La mise à l'échelle permet également une meilleure convergence chez les réseaux de neurones. La mise à l'échelle peut se décrire mathématiquement comme suit :

$$z = \frac{x - min(x)}{max(x) - min(x)} \tag{1.2}$$

où z est l'image résultante et x l'image d'entrée.

1.1.2.3 Normalisation gaussienne (*standardization*)

La normalisation gaussienne, ou *standardization*, prend en considération que les données sont issues d'une distribution gaussienne (normale). Cette méthode soustrait la moyenne et divise les pixels par la variance. Ces deux statistiques peuvent être issues soit de l'image elle-même ou du jeu de données. De cette façon, les intensités de l'image deviennent une distribution normale ayant une moyenne d'intensité 0 avec une variance unitaire. Cependant, n'ayant que deux paramètres, la méthode laisse présager à une performance suboptimale lorsque plusieurs ensembles de données sont utilisés simultanément. De plus, cette méthode est principalement utilisée dans le cadre de la classification d'images naturelles. Les pixels de ces images se situent généralement dans une plage de valeur $[0, 255]$ et ce peu importe la source de l'image. Or, ce n'est pas le cas dans les images médicales qui ont une plage de valeurs indéfinie.

$$z = \frac{x - \mu}{\sigma} \tag{1.3}$$

où z est l'image résultante, x l'image d'entrée, μ est la moyenne des intensités de l'image et σ est l'écart-type des intensités de l'image.

1.1.2.4 Normalisation quantile

La normalisation quantile est une technique de normalisation qui rend deux distributions identiques après transformation. La normalisation quantile transforme les intensités des images de façon à avoir une médiane de 0 et une distance interquartile unitaire sous condition d'avoir un jeu de données contenant suffisamment d'échantillons. La distribution cible est généralement une distribution uniforme ou gaussienne. Cependant, cette méthode assume que la distribution entre les échantillons est semblable. Or, ce n'est pas toujours le cas en imagerie médicale, spécialement lorsque plusieurs domaines d'imageries sont considérés dans un même problème. De plus, la distribution entre certains types de patients peut grandement varier. Par exemple, la distribution de la matière blanche et la matière grise pour un enfant de 6-12 mois n'est pas la même que celle d'un adulte (Wang, Nie, Li, Puybareau, Dolz, Zhang, Wang, Xia, Wu, Chen, Thung, Bui, Shin, Zeng, Zheng, Fonov, Doyle, Xu, Moeskops, Pluim, Desrosiers, Ayed, Sanroma, Benkarim, Casamitjana, Vilaplana, Lin, Li & Shen, 2019). L'utilisation de cette méthode de normalisation peut ainsi mener à des résultats suboptimaux sur une tâche de segmentation.

1.1.2.5 Concordance d'histogrammes (*histogram matching*)

Cette méthode consiste d'abord en la création d'un histogramme de référence (*template histogram*), souvent représenté par l'histogramme moyen des images représentant une population. Durant cette phase d'entraînement, des points clés peuvent également être définis, comme les centiles de la distribution Nyul, Udupa & Xuan Zhang (2000). Les histogrammes des images du jeu de données étudié sont ensuite ajustés en fonction de l'histogramme modèle. La fonction de transformation est une fonction par morceaux ou polynomiale qui ajuste l'histogramme de l'image de chaque sujet de façon à réduire la somme des erreurs absolues entre l'histogramme de référence et l'image ajustée. Toutefois, cette méthode assume que la distribution des tissus est la même entre les sujets, que ces derniers n'ont pas d'anomalies biologiques et que les artéfacts techniques tels que le mouvement ou l'inhomogénéité spatiale (*bias field*) n'existent pas (Shinohara *et al.*, 2014). Cela en fait une méthode de normalisation moins robuste et moins appropriée pour des études comportant des images de plusieurs sujets distincts ayant

potentiellement des pathologies ou un développement différent. Shinohara *et al.* (2014) présente la méthode *White Stripe* qui introduit la notion de classes des tissus dans la méthode traditionnelle de concordance d'histogrammes. En échantillonnant les intensités à des endroits précis du cerveau constitué de matière blanche, l'auteur peu normaliser une image de manière plus robuste tout en préservant les contrastes des tissus. Toutefois, la méthode se veut singulière dans son nombre de domaines et ne tente pas de normaliser à travers plusieurs de ceux-ci.

1.1.2.6 Méthodes apprises (*learned methods)*

Récemment, quelques études ont exploré le potentiel de méthodes d'apprentissage profond afin de normaliser des images médicales (Drozdzal, Chartrand, Vorontsov, Shakeri, Di Jorio, Tang, Romero, Bengio, Pal & Kadoury, 2018), le débruitage d'images (Oguz, Malone, Atay & Tao, 2020), l'adaptation de domaine (Ciga, Chen & Martel, 2019) et l'augmentation de données par transfert de style (Hesse, Kuling, Veta & Martel, 2020). La normalisation dynamique présentée dans Drozdzal *et al.* (2018) permet d'optimiser le processus de normalisation pour une tâche spécifique telle que la segmentation sémantique. Puisqu'aucune contrainte sur le réalisme de l'image n'est appliquée, il en résulte des images aux intensités grandement altérées, les rendant difficiles à interpréter par un clinicien. Dans Ciga *et al.* (2019), bien que la méthode supporte plusieurs domaines, par exemple différents ensembles de données acquis avec des protocoles d'imagerie différents, celle-ci n'a pas été élaborée dans le cadre d'une tâche spécifique telle que la segmentation. Ainsi, il en résulte une performance de segmentation suboptimale. Dans Hesse *et al.* (2020), une fonction de style est extraite d'un réseau de neurones prédicteur de style, puis appliqué à une image. Il en résulte des images aux intensités augmentées de manière intelligente en prétraitement, avant d'être soumises à un réseau de segmentation. Cette méthodologie permet d'entraîner un réseau de segmentation avec une plus grande distribution d'intensités sans toutefois optimiser cette distribution en fonction de la performance de segmentation. En effet, l'aspect *hors-ligne* de la méthode proposée est aussi un désavantage puisqu'elle ne peut optimiser de manière directe la performance de segmentation, celle-ci reposant sur le fait qu'une plus grande variété d'intensités et d'échantillons permettent d'accroître le pouvoir statistique du

réseau de segmentation, ayant pour conséquence d'augmenter sa précision. De plus, les images produites sont explicitement caractérisées comme étant non réalistes, et donc non utilisables dans un environnement clinique. Modanwal, Vellal, Buda & Mazurowski (2020) a utilisé un réseau *cycle-consistant generative adversarial network* afin de générer des images normalisées de mammographies acquises à partir de deux types d'imageurs différents. La méthode proposée, de par sa dépendance de cycle, permet d'entraîner deux pairs de générateurs et discriminateurs avec des images non pariées. Cela permet de contourner la limitation courante des algorithmes de translation d'image qui requiert des données pairées. Cette méthode a aussi été explorée dans Oguz *et al.* (2020) afin de retirer le bruit dans des images. Alors que les résultats montrent des images harmonisées visuellement réalistes, la méthode ne considère pas de tâche spécifique effectuée après le prétraitement. De plus, cette approche d'harmonisation est limitée à deux domaines. Étendre la méthode à plusieurs domaines requiert des modifications importantes.

1.2 Apprentissage automatique profond et réseaux de neurones convolutifs

La quantité de données à analyser étant en perpétuelle augmentation, de nouveaux algorithmes prédictifs ont vu le jour. Bien que le perceptron fut inventé en 1958 par Rosenblatt et l'algorithme de rétropropagation (*backpropagation*) en 1963 par Bryson, ce n'est que dans la dernière décennie que nous avons véritablement vu l'explosion de l'utilisation des réseaux de neurones (Litjens, Kooi, Bejnordi, Setio, Ciompi, Ghafoorian, van der Laak, van Ginneken & Sánchez, 2017).

Le perceptron, soit l'unité primaire du réseau de neurones, se voit alloué des poids qui seront multipliés par les valeurs en entrée issues des perceptrons de la couche précédente. Un biais est ajouté, suivi d'une fonction d'activation non linéaire. Ainsi, la sortie z d'un perceptron peut être exprimée comme suit :

$$z = w \cdot x + b \tag{1.4}$$

où w est le vecteur de poids, x l'entrée, b le biais et $\cdot$ est le produit scalaire.

La valeur optimale des poids est calculée de façon itérative par descente de gradients stochastique (*SDG*) selon l'erreur mesurée par une fonction de coût donnée. L'algorithme de rétropropagation permet l'ajustement systématique des poids à chaque itération en appliquant à ceux-ci la norme du gradient multiplié par un coefficient, soit le taux d'apprentissage (*learning rate*).

Le *Multi-Layer Perceptron* (MLP), ou réseau de perceptrons multicouches, est le premier modèle de réseaux de neurones permettant l'apprentissage de fonctions de mappage complexes, non linéaires, et de haute dimension. Le MLP est essentiellement une composition successive de transformations affines et de fonctions d'activations. Dans sa plus simple expression, un réseau de neurones peut s'exprimer ainsi :

$$f : \mathbb{R}^n \rightarrow \mathbb{R}^m \tag{1.5}$$

où n est le nombre de perceptrons en entrée, m est le nombre de perceptrons en sortie et f est la fonction non linéaire apprise.

Dans le cas d'un MLP, celui-ci est une structure de perceptrons composée d'au minimum trois couches, soit une couche d'entrées, une couche cachée et une couche de sorties. Lorsqu'il possède plus de deux couches cachées, le MLP prend généralement le nom de *Deep Neural Network* (*DNN*), ou réseau de neurones profond.

L'universalité des réseaux de neurones fut démontrée par le théorème de Cybenko (1989). Ce théorème stipule qu'on peut approximer toute fonction compacte dans $\mathbb{R}$ avec un réseau à trois couches tout en ayant N perceptrons dans la couche cachée. Cependant, afin d'avoir une erreur de classification sous un certain seuil ϵ, le nombre de perceptrons dans la couche cachée augmente de façon exponentielle. Ainsi, alors que la largeur des réseaux de neurones est associée à *l'additivité* des fonctions, l'idée d'avoir un réseau de neurones profond est plutôt associée à la *composition* de fonctions. Cette composition augmentera de façon exponentielle la non-linéarité de la fonction résultante du réseau tout en gardant le nombre d'hyperparamètres linéaire. Le contrôle du nombre de ces hyperparamètres (poids) est important puisqu'on restreint l'espace

nécessaire en mémoire tout en conservant une complexité de calcul moindre. La profondeur du réseau permet de composer des fonctions plus complexes et d'engendrer des espaces plus complexes souhaités à des tâches telles que la classification. Les bienfaits d'avoir des réseaux plus profonds a d'ailleurs été démontré par le théorème de Telgarsky (2016).

Cependant, ces types de réseaux travaillent avec pour entrées des caractéristiques des données. Ces caractéristiques, extraites manuellement à l'aide d'algorithmes d'extraction, devraient idéalement être invariantes à l'échelle, la rotation et la translation. Or, ce processus d'extraction des caractéristiques est très souvent un processus fastidieux et limite le nombre de dimensions du problème à quelques-unes. Aussi, les réseaux de neurones profonds, ayant plus de poids, sont plus sujets au surapprentissage et requièrent davantage de données d'entraînement. De plus, les MLP ne peuvent localiser. Une image 2D doit, au préalable, être transformée en vecteur, perdant ainsi sa structure spatiale. LeCun, Bottou, Bengio & Haffner (1998) permet d'extraire des caractéristiques à même les données en proposant le réseau de neurones convolutif (CNN). Le CNN est l'architecture ayant été la plus utilisée et la plus couronnée de succès dans le domaine de l'imagerie dans le courant de la dernière décennie. Alors que le MLP se base sur un arrangement de perceptrons disposés en couches, le CNN repose sur les opérations de convolution, sous échantillonnage par valeur maximale (*max-pooling*) et parfois une couche pleinement connectée (*fully connected*) servant à la classification.

1.2.1 Opération de convolution

L'opération de convolution permet essentiellement d'extraire des caractéristiques intrinsèques aux images. Il faut reconnaître qu'une image est une structure locale 2D dont les pixels d'un voisinage sont fortement corrélés. Ces caractéristiques prennent la forme de filtres, soit une matrice carrée, généralement de taille $3x3$, $5x5$ ou $7x7$ dont les coefficients et leur biais sont appris tout au long du processus d'entraînement. Ces filtres sont regroupés en plans (aussi appelés canaux), et chaque plan représente une carte de caractéristiques *feature map*. Les CNN peuvent partager des paramètres. Conformément à cette caractéristique, un CNN permet d'avoir beaucoup moins de paramètres à entrainer, réduisant considérablement l'empreinte en mémoire

et le nombre de données requises pour l'entraînement tout en prenant en considération le voisinage des pixels d'une image.

Les filtres appris par le biais de la convolution sont invariants à la translation et aux déformations locales, contournant ainsi une des plus importantes problématiques du MLP lorsque celui-ci traite des images. Un CNN est alors le produit d'une extraction de caractéristiques locales. Ces caractéristiques peuvent être utilisées partout sur l'image. Contrairement au modèle MLP dit *feed-forward*, toutes ces cartes de caractéristiques ne sont pas nécessairement interconnectées à chaque couche. Les filtres des couches suivantes sont connectés à un filtre en particulier. Plus la profondeur du réseau est grande, plus l'agencement des sorties de ces filtres, les cartes de caractéristiques, produit des formes complexes. Chaque filtre ne voit qu'une portion de l'image, soit le champ de vision local (*local receptive field*). Ce champ de vision local est la portion de la donnée d'entrée d'une couche de convolution qui est exposée à l'opération de convolution. Ainsi, les CNN exploitent une caractéristique propre à ce type de réseau de neurones, soit la connectivité locale. De ce fait, chaque neurone d'une couche apprend visuellement des caractéristiques invariables dans l'espace d'une région de l'image d'entrée, chose impossible à faire avec un réseau de type MLP. Les CNN présument alors implicitement que deux pixels voisins ont une plus forte corrélation que deux pixels éloignés, une caractéristique propre aux signaux 1D et 2D comme une image.

Dans le contexte d'une image médicale IRM structurelle, par exemple une image T1, cette image représente un volume 3D. Ainsi, l'opération de convolution 2D est convertie en 3D. De ce fait, les filtres sont plutôt de taille $3x3x3$, $5x5x5$ ou $7x7x7$. Cela augmente nécessairement le nombre de paramètres du réseau, mais conserve l'aspect tridimensionnel crucial à ce type de données.

1.2.2 Couches d'activations

1.2.2.1 Sigmoïde

Tout comme dans les MLP, chaque neurone se voit attribuer une fonction d'activation. Puisque la convolution est une opération mathématique linéaire, cette fonction permet de rendre le modèle appris non linéaire afin que celui-ci épouse au mieux les données de l'ensemble d'entraînement. Il existe néanmoins plusieurs variantes de fonctions d'activation. La fonction sigmoïde est l'une des plus utilisées :

$$y = \frac{1}{1 + e^{-x}} \tag{1.6}$$

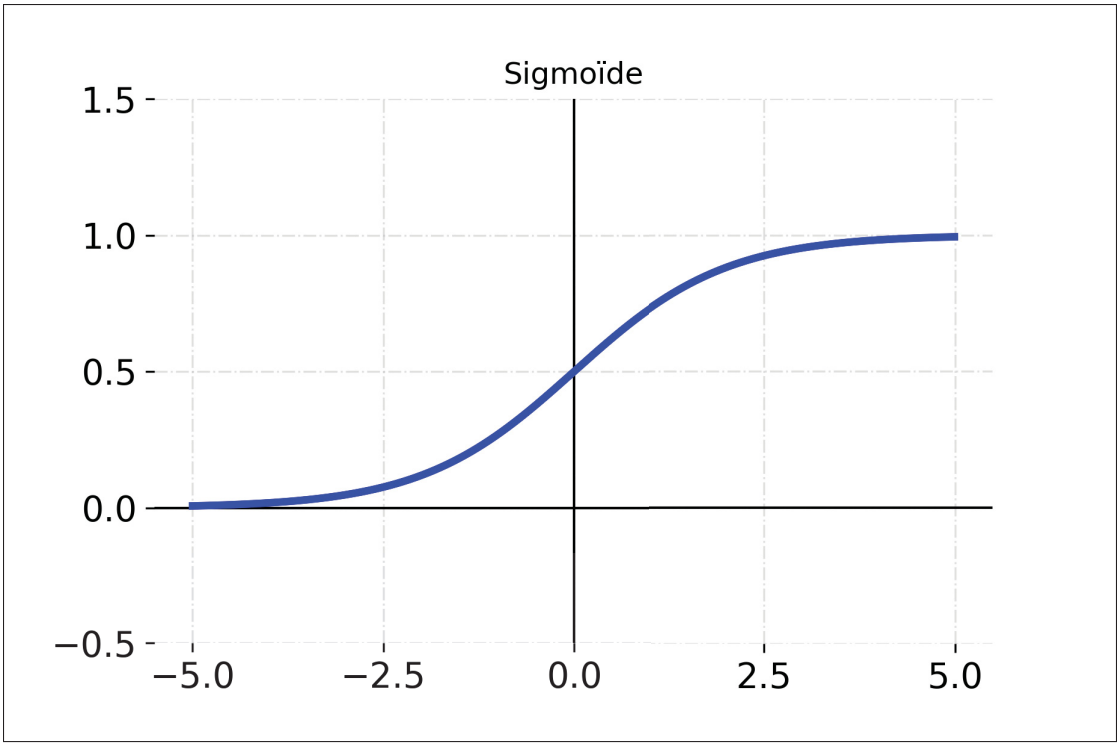

Figure 1.3 Graphe de la fonction Sigmoïde

Bornée dans l'intervalle [0, 1], lisse et dérivable sur tout son domaine, la fonction sigmoïde est généralement utilisée dans le but de prédire une probabilité dans le cas d'un réseau de

classification. Toutefois, en raison de sa forme allongée aux extrémités, la fonction sigmoïde a tendance à saturer (Glorot, Bordes & Bengio, 2010). Les réseaux profonds qui implémentent cette fonction d'activation ont tendance à souffrir du problème de la disparition des gradients (*vanishing gradients*). Lorsqu'un réseau fait face à ce problème, le processus d'apprentissage de la fonction cible est très lent, voire inexistant.

1.2.2.2 ReLU

La fonction *Rectified Linear Unit*, ReLU, vient pallier ce problème de disparition des gradients :

$$y = max(0, x) \tag{1.7}$$

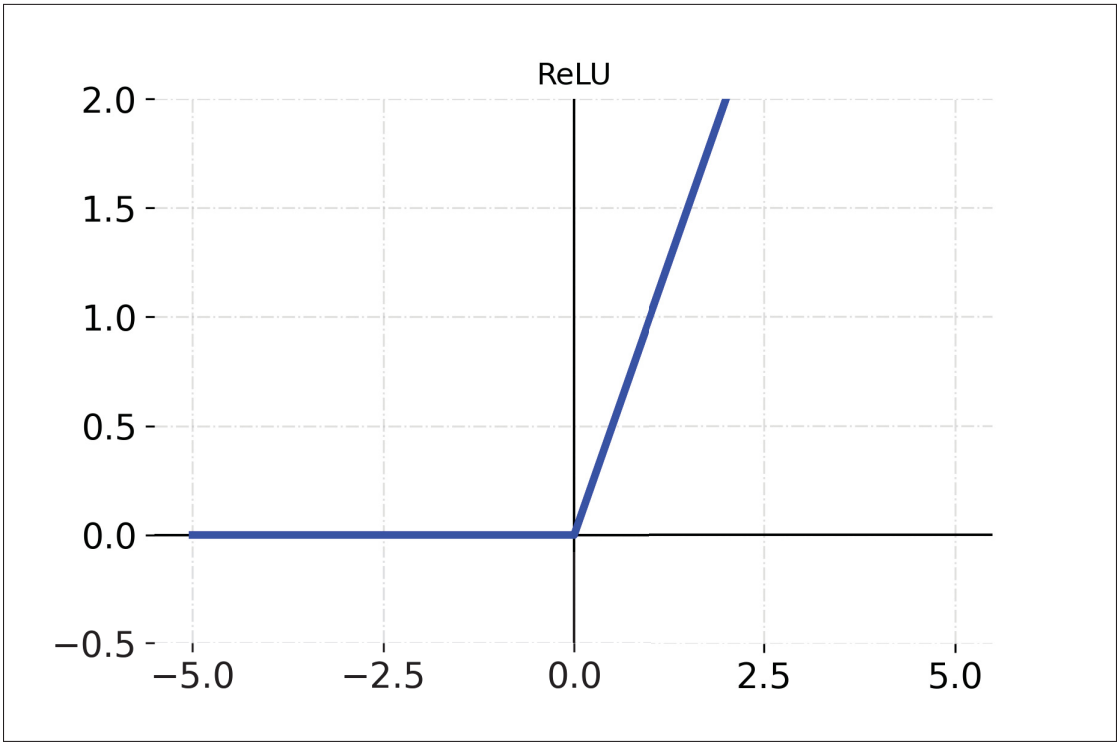

Figure 1.4 Graphe de la fonction ReLU

Toute fonction peut être approximée par une composition de fonctions ReLU. De plus, cette fonction exploite la sparsité des réseaux de neurones (Glorot *et al.*, 2010), les rapprochant

beaucoup plus du fonctionnement des neurones biologiques connectés dans le cerveau humain. En effet, alors que la fonction sigmoïde a un régime permanent autour de $y = \frac{1}{2}$, la fonction ReLU permet d'avoir des valeurs nulles à différents endroits dans le réseau, le rendant *creux* (de l'anglais *sparse*) et ainsi beaucoup plus plausible d'un point de vue biologique. La sparsité et l'aspect distribué des neurones d'un cerveau humain furent étudiés dans (Attwell & Laughlin, 2001) en étudiant notamment la consommation d'énergie du cerveau en analysant les visualisations d'IRM fonctionnelles. Les avantages de la sparsité des réseaux de neurones fut étudiée dans Glorot *et al.* (2010). Avec le phénomène de la sparsité, il est plus probable de trouver une solution permettant une séparation linéaire des données, d'avoir un isolement des données (*information disentangling*) et d'avoir une structure distribuée plus représentative de la neuroscience au sein du réseau de neurones artificiel. Il est plus probable qu'un neurone au sein d'un réseau de neurones creux (*sparse*) soit plus significatif. Cependant, puisque la fonction ne retourne aucune valeur négative, le réseau peut avoir du mal à mapper les valeurs négatives, ce qui, dans certains cas, pourrait nuire à la performance du réseau, menant au problème du *Dying ReLU*. Puisque l'activation pour cette portion du domaine de la fonction est de 0, des neurones peuvent voir un gradient toujours égal à 0 et ainsi «mourir». Ce problème peut mener à la stagnation du réseau dans un endroit non optimal de l'espace des solutions.

Afin de pallier ce problème, d'autres variantes de la fonction ReLU sont également populaires, notamment la fonction Leaky-Relu. Cette variante de ReLU permet d'avoir une légère pente lorsque $x < 0$ et ainsi d'avoir un domaine sur l'intervalle $[-\infty, \infty]$. Parametric ReLU (PReLU) permet de paramétrer un coefficient α définissant la pente de la droite lorsque $x < 0$. Ce coefficient peut être entraîné et optimisé tout au long du processus d'entraînement.

1.2.3 Échantillonnage maximum

Dans un réseau de neurones convolutif, l'étape de sous-échantillonnage non linéaire permet de réduire la taille des cartes de caractéristiques (*feature maps*) extraites. Plusieurs méthodes de sous-échantillonnage existent. Parmi les plus populaires, le sous-échantillonnage par valeur maximale (*Max Pooling*) est généralement utilisé. Cette méthode permet de conserver la réponse

maximale d'un signal en sélectionnant la valeur la plus haute dans un noyau tout en réduisant sa taille. Cela permet ainsi une meilleure propagation de l'information au fur et à mesure qu'on avance dans la profondeur du réseau.

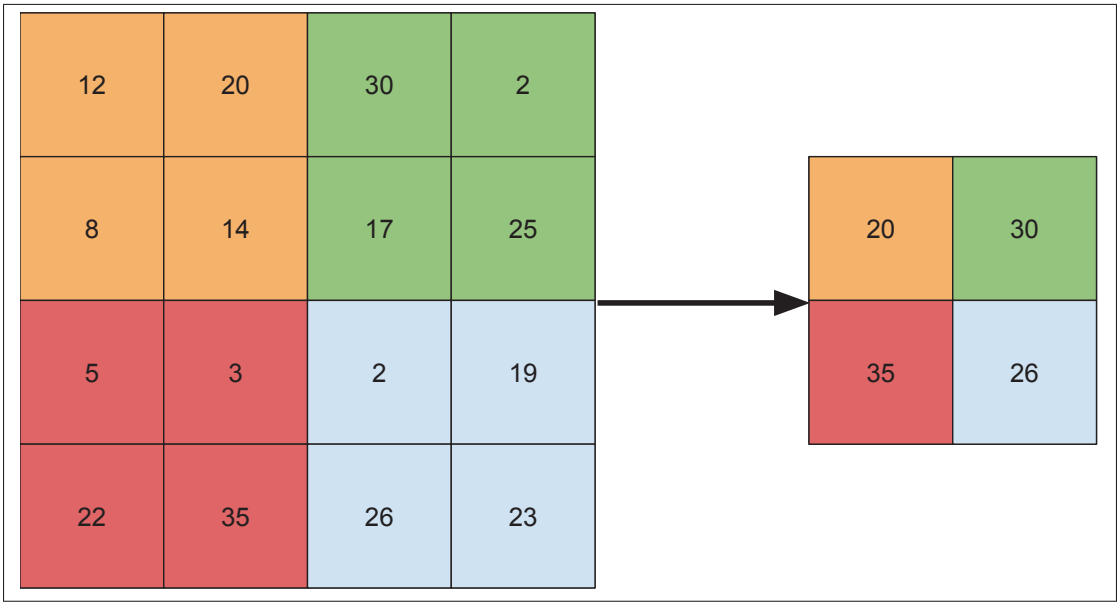

Figure 1.5 Schéma de l'échantillonnage maximum 2D avec noyau de dimension 2×2

Une autre méthode d'échantillonnage, soit le sous-échantillonnage moyen (*average pooling*), fait une moyenne des cartes de caractéristiques.

Dans les deux cas, la carte de caractéristique est partitionnée généralement en noyaux de tailles $2x2x2$ qui ne se chevauchent pas. La valeur maximale ou la valeur moyenne de ces noyaux est sélectionnée et retenue comme sortie.

1.2.4 Couche pleinement connectée

Une couche pleinement connectée est généralement ajoutée à la fin du réseau convolutif afin d'effectuer la classification finale. Elle est connectée à tous les neurones de la couche précédente tout comme dans un modèle MLP. Elle permet de réduire la dimensionnalité aux nombres de classes présentes dans le problème.

1.2.5 Optimisation par descente de gradient et rétropropagation du gradient

Puisque la mémoire est limitée, l'entraînement d'un réseau de neurones se fait par itération en utilisant la technique de descente de gradients. Chaque itération se voit présenter un sous-ensemble de l'ensemble de données d'entraînement. En plus d'exploiter directement l'efficacité du parallélisme des architectures matérielles de calculs modernes, chacun des sous-ensembles (*mini-batch*) est une approximation du gradient de l'ensemble de données d'entrainement. La descente de gradients, qui optimise les hyperparamètres Θ d'un réseau de neurones en se basant sur une fonction de coût ℓ, peut s'exprimer comme suit (Ioffe & Szegedy, 2015) :

$$\Theta = \arg\min_{\Theta} \frac{1}{N} \sum_{i=1}^{N} \ell\left(X_i, \Theta\right) \tag{1.8}$$

où Θ représente les hyperparamètres du réseau de neurones, ℓ représente la fonction de coût, $X_{1..N}$ est l'ensemble de données d'entraînement. Dans le cas d'un entraînement par sous-ensembles, on remplacera N par m où m la taille (nombre de données) du sous-ensemble désiré. Le gradient de la fonction de coût devient (Ioffe & Szegedy, 2015) :

$$\frac{1}{m} \frac{\partial \ell\left(X_i, \Theta\right)}{\partial \Theta} \tag{1.9}$$

Lors d'une itération, le gradient est appliqué à chaque neurone de la sortie du réseau jusqu'à l'entrée. Ce processus est répété jusqu'à la convergence de la fonction de coût.

1.2.6 Régularisation et normalisation en lot *(Batch Norm)*

La normalisation de lots de données (des sous-ensembles de l'ensemble de données) réduit la difficulté qu'est l'entraînement d'un réseau de neurones. Le processus d'entraînement est complexe en raison de la rétropropagation du gradient. Par conséquent, les neurones d'une certaine couche du réseau sont affectés par les paramètres des couches qui les précèdent, pouvant ainsi accentuer certains comportements et faire diverger le réseau de la solution recherchée.

Puisque nous utilisons le principe de sous-ensembles (*mini-batch*) précédemment décrit, le phénomène du *covariate shift* apparaît. Ce phénomène se décrit comme étant un changement dans la distribution dans le domaine d'une fonction et est principalement issu du manque de ressources computationnelles (Shimodaira, 2000). L'*internal covariate shift* est quant à lui le changement de la distribution des activations du réseau de neurones en raison du changement de ses hyperparamètres tout au long de l'entraînement. La normalisation en lot (*batch normalization*) (Ioffe & Szegedy, 2015) est un mécanisme permettant d'éliminer ce changement dans la distribution des neurones constituant le réseau. Cette opération permet notamment d'accélérer grandement l'entraînement du réseau en permettant d'avoir un taux d'apprentissage plus élevé et réduit la dépendance du réseau à la qualité de son initialisation. De plus, elle régularise le réseau en évitant de produire des valeurs déterministiques pour un échantillon donné. De plus, cette technique stabilise l'entraînement en stabilisant les distributions des données en entrée dans le réseau. Finalement, cette technique ajoute un bruit inhérent aux valeurs ponctuelles de l'espérance et de la variance d'un sous-ensemble d'entraînement. La normalisation en lot s'exprime ainsi :

$$\hat{x}^{(k)} = \frac{x^{(k)} - \mathbb{E}\left[x^{k}\right]}{\sqrt{Var\left[x^{(k)}\right]}} \tag{1.10}$$

où la variance et l'espérance sont calculées sur l'ensemble d'entraînement au fur et à mesure que le processus d'entraînement s'effectue. Cette normalisation est appliquée sur toutes les dimensions des données en entrée.

1.2.7 Initialisation des poids

Malgré l'efficacité du principe de la descente de gradient précédemment discuté, un réseau de neurones est très sensible aux valeurs avec lesquelles il est initialisé. Une mauvaise initialisation des poids dans le réseau peut mener à divers problèmes, notamment la disparition et l'explosion du gradient (*gradient vanishing and exploding*). L'initialisation serait également étroitement liée

aux fonctions d'activation présentes dans le réseau. He, Zhang, Ren & Sun (2015) a développé une méthode d'initialisation optimisée pour les réseaux ayant pour fonction d'activation ReLU et ses variantes, soit *Kaiming Initialization* :

$$\frac{1}{2}(1 + a^2)n_l Var[w_l] = 1, \forall l \tag{1.11}$$

où l est le nombre de neurones dans la couche précédente (*fan-in*), w est le poids à initialiser, n est le nombre de connexions à un poids (*fan-out*) et où a est est la pente de la portion négative du domaine de la fonction ReLU utilisée (0 dans le cas de ReLU). Cela mène à l'initialisation du poids avec une distribution gaussienne où la variance est $\sqrt{\frac{2}{n_l}}$.

He *et al.* (2015) ont démontré qu'en utilisant cette méthode, la convergence du réseau s'effectue beaucoup plus rapidement. La méthode est issue de la modélisation de la non-linéarité et la non-symétrie de la fonction ReLU contrairement à l'initialisation *Xavier* (Glorot & Bengio, 2010) qui tenait pour acquis que les fonctions d'activations étaient linéaires et symétriques. Cette hypothèse est invalide dans le cas d'un réseau pourvu des fonctions ReLU et PReLU. L'initialisation Kaiming permet également d'avoir des réseaux plus profonds sans tomber dans les problèmes de disparition ou explosion du gradient.

1.2.8 Réseaux de neurones pleinement convolutionnels

Alors que les réseaux de neurones convolutifs sont typiquement conçus pour effectuer de la classification, les réseaux de neurones pleinement convolutifs (*Fully Convolutional Neural Networks - FCN*) (Shelhamer, Long & Darrell, 2017) se veulent dédiés notamment à la segmentation sémantique. Ces types de réseaux éliminent la couche pleinement connectée qui se retrouve généralement à la fin d'un réseau de classification. Dans le cas d'un FCN, la classification est quant à elle effectuée plutôt par une convolution $1x1x1$. Cette opération permet ainsi d'avoir des images de taille variable en entrée du réseau de segmentation et de produire une carte de probabilités (*heatmap*), c'est-à-dire le résultat de la composition des filtres profonds.

Puisqu'un CNN est typiquement un extracteur de caractéristiques, la résolution diminue au fur et à mesure qu'on avance dans les couches du réseau. Afin de retrouver la taille initiale de l'entrée, une ou plusieurs couches de convolutions transposées (*deconvolution*) ou des couches de suréchantillonnage (*upsampling*) sont utilisées. Ce type de réseau est tout à fait approprié à un problème *dense* comme la segmentation sémantique puisque la sortie du réseau est de même taille que la donnée en entrée. Le nombre de dimensions à la sortie du réseau correspond au nombre de classes du problème, chacune de ces dimensions étant un vecteur de probabilités pour chaque voxel classifié. Ainsi, la valeur maximale pour chaque voxel le long de ces dimensions représente la classe à laquelle le voxel appartient. Afin de ne pas biaiser l'apprentissage du réseau vers une classe en particulier (généralement celle ayant le plus grand nombre de voxels), la fonction de coût utilisée est communément pondérée en fonction de la fréquence d'occurrence de chaque classe présente dans l'ensemble d'entraînement. Drozdzal, Vorontsov, Chartrand, Kadoury & Pal (2016) propose un réseau de neurones pleinement convolutif profond basé sur les connexions résiduelles.

1.2.9 Architecture U-Net

L'architecture U-Net (Ronneberger, Fischer & Brox, 2015) est un réseau de neurones pleinement convolutif (FCN) principalement utilisé à des fins de segmentation sémantique où chaque pixel de l'image de sortie doit se voir attribuer une étiquette correspondant à la classe à laquelle il appartient. La première partie de U-Net est un encodeur. Dans cette portion du réseau, l'information est encodée au moyen de deux filtres de convolutions successifs suivis d'une opération de sous-échantillonnage. Le nombre de filtres est doublé après chaque sous-échantillonnage L'encodeur est composé d'au moins 3 de ces blocs. La deuxième portion du réseau, soit le décodeur, permet l'expansion des cartes de caractéristiques. Il s'agit d'opérations successives de convolutions suivies d'une opération de suréchantillonnage (interpolation). Le suréchantillonnage (*upsampling*) peut également être remplacé par une déconvolution qui divise par deux le nombre de filtres dans la couche suivante tout en doublant la résolution de l'image. Cela permet au réseau de propager de l'information contextuelle aux couches à plus haute

résolution. La déconvolution propose certains avantages. Notamment, elle est une opération qui peut être optimisée puisqu'elle dispose de paramètres qui sont entraînables. Elle dispose de la même connectivité qu'une couche de convolution. À chaque niveau du U-Net où la résolution entre l'encodeur et le décodeur est identique, l'information spatiale haute résolution est transmise de l'encodeur au décodeur. Cela permet au réseau de localiser tout en préservant l'intégrité structurelle des données. Les convolutions successives, quant à elles, permettent au réseau de produire une sortie plus précise à l'aide de l'information transmise de l'encodeur. Cette information est transmise sous forme de concaténation des cartes de caractéristiques apprises par l'encodeur. La classification finale est effectuée par une convolution avec un noyau unitaire, permettant ainsi d'avoir une sortie avec le nombre de canaux désirés représentant le nombre de classes du problème. Le réseau possède au total 23 couches de convolution. N'ayant pas de couches pleinement connectées, le réseau peut prendre des images de différentes résolutions en entrée. En raison du faible échantillon et du nombre élevé d'hyperparamètres à optimiser, l'auteur propose d'utiliser conjointement le réseau avec une technique d'augmentation de données qui est généralement une transformation élastique, une rotation ou translation.

3D U-Net (Çiçek, Abdulkadir, Lienkamp, Brox & Ronneberger, 2016) reprend le principe de U-Net, mais l'applique à la segmentation de données volumétriques. Ainsi, avec 3D U-Net, l'aspect spatial tridimensionnel des données médicales est conservé tout au long du réseau de neurones. Par conséquent, cela augmente le nombre d'hyperparamètres et l'espace mémoire requise pour effectuer les calculs. Néanmoins, 3D U-Net demeure un réseau FCN très populaire et efficace pour une tâche de génération et de segmentation d'images.

Milletari, Navab & Ahmadi (2016) a proposé une architecture 3D U-Net proposant une architecture résiduelle. Aussi, l'opération de sous-échantillonnage est remplacée par une convolution avec un pas de 2, ce qui rend possible l'optimisation du sous-échantillonnage en fonction de la tâche effectué par le réseau.

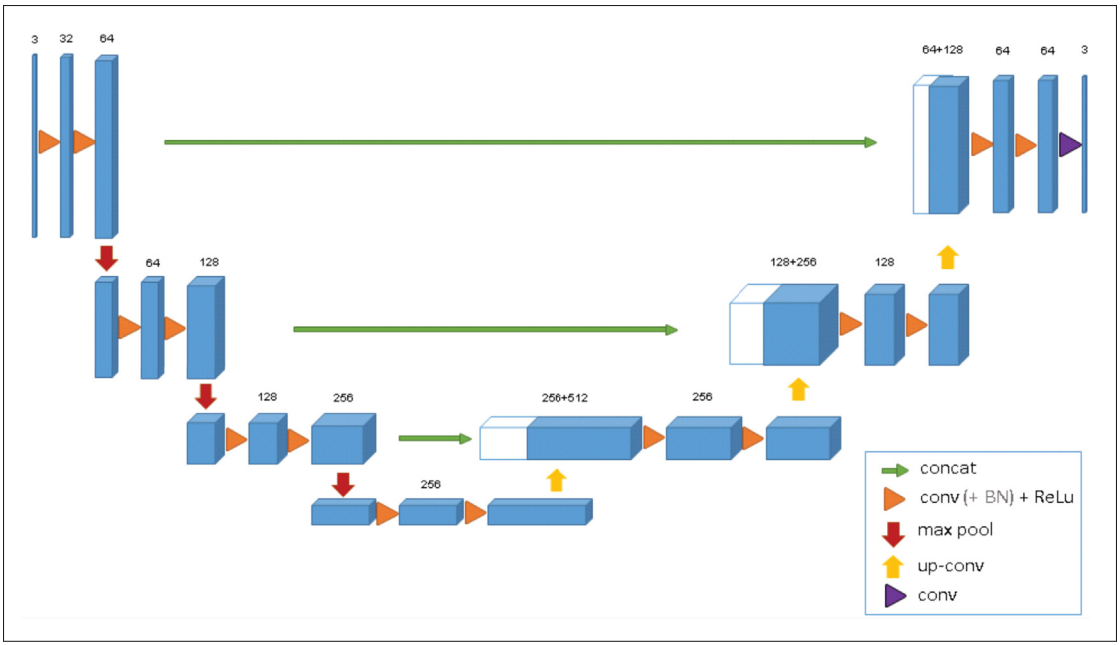

Figure 1.6 Architecture 3D U-Net. Tirée de Çiçek *et al.* (2016)

1.2.10 Architecture ResNet

ResNet (He, Zhang, Ren & Sun, 2016) est une architecture de réseau de neurones de classification élaboré afin d'avoir un meilleur taux de classifications sur le populaire ensemble de données ImageNet. Un problème majeur des réseaux de neurones est leur dégradation de performance au fur et à mesure qu'on ajoute de la profondeur au réseau. Malgré les succès des réseaux de neurones profonds, un réseau plus profond ne veut pas nécessairement dire qu'il sera plus performant. Son taux de classification aura plutôt tendance à saturer. Ainsi, ce problème explicite le fait que des couches profondes apprennent nécessairement une fonction identité puisqu'elles n'apportent aucun pouvoir discriminant au réseau. Ce problème de dégradation des performances est considéré dans ResNet et l'auteur propose une architecture en blocs résiduels. Ce type de bloc, constitué d'opérations de convolution, permet aux couches du réseau d'apprendre une fonction résiduelle :

$$\mathcal{F}(x) := \mathcal{H}(x) - x\mathcal{H}(x) := \mathcal{F}(x) + x \tag{1.12}$$

où $\mathcal{F}(x)$ représente la fonction de mappage de l'entrée apprise et x la fonction identité, communément appelée *skip connection*. Cette connexion se traduit par l'addition de la donnée d'entrée à la sortie d'un bloc résiduel, ce dernier étant constitué d'une opération de convolution suivie d'une fonction d'activation.

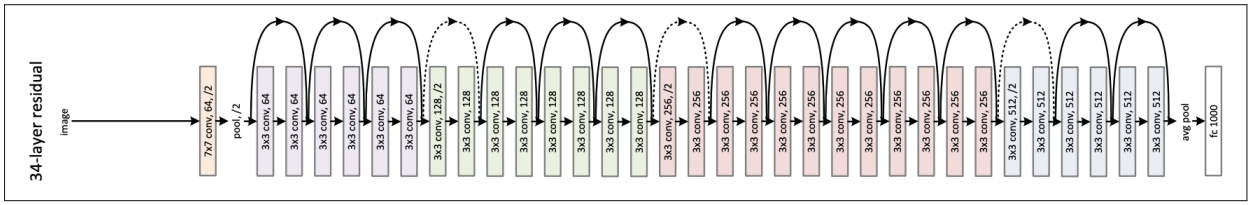

Figure 1.7 Architecture ResNet. Tirée de He *et al.* (2015)

L'hypothèse derrière cette connexion est qu'un réseau qui apprend dans son optimalité la fonction identité $\mathcal{F}(x) = x$, cette fonction $\mathcal{F}$ est forcément plus proche de la solution réelle qu'une fonction de mappage $\mathcal{F}(x) = 0$. Ainsi, l'information apprise par les couches précédentes est transmise aux couches suivantes, laissant le réseau se focaliser sur des perturbations différentes des données en entrées. Ainsi, ResNet est un excellent extracteur de caractéristiques, ce qui en fait un réseau de classification hors pair.

1.3 Réseaux génératifs antagonistes

Goodfellow, Pouget-Abadie, Mirza, Xu, Warde-Farley, Ozair, Courville & Bengio (2014) a proposé les réseaux génératifs antagonistes. Il s'agit d'une architecture qui regroupe un *générateur* et un *discriminateur* d'images. Le générateur échantillonne des images directement à partir d'un vecteur de bruit de la distribution souhaitée (généralement gaussienne ou uniforme). Le discriminateur, quant à lui, apprend une fonction de classification permettant de classifier l'échantillon qu'on lui présente. Le discriminateur détermine si l'image est générée ou issue du vrai ensemble de données. Sa fonction de coût est ensuite propagée jusqu'au générateur, qui corrige ses poids de manière à avoir une image plus réaliste. Au début de l'entraînement, l'image générée par le générateur est de mauvaise qualité et peu réaliste. Au fur et à mesure que l'entraînement se poursuit, le générateur ajuste ses poids et l'image devient de plus en plus

réaliste, de telle sorte que le discriminateur n'arrive plus à distinguer de quelle distribution provient l'échantillon. Ainsi, les deux réseaux arrivent à convergence. Le système peut être représenté par l'équation suivante :

$$\min_{G}\max_{D} V(D,G) = \mathbb{E}_{x\sim\mathcal{P}_{data(x)}}\left[logD(x)\right] + \mathbb{E}_{z\sim\mathcal{P}_{z}(z)}\left[log\Big(1 - D\big(G(x)\big)\Big)\right] \tag{1.13}$$

où G est une fonction différentiable représentée par un réseau de neurones convolutif jouant le rôle de générateur d'images, D est également une fonction différentiable aussi représentée par un réseau de neurones convolutif servant à classifier le domaine d'origine d'une image en entrée, $\mathcal{P}_z(z)$ représente la distribution du vecteur de bruit servant à la génération d'une image par G, $\mathcal{P}_{data(x)}$ est la distribution d'images réelles, $G(x)$ est une image générée, $D(x)$ est la probabilité qu'une entrée x soit une image réelle.

Les GANs ont notamment été utilisés dans le domaine médical afin d'explorer les structures intrinsèques aux données (Yi, Walia & Babyn, 2019). Des applications comme la reconstruction d'images médicale sont passées de méthodes analytiques à itératives puis basées sur des techniques d'apprentissage automatique super-résolution (Sánchez & Vilaplana, 2018; Chen *et al.*, 2018a). La segmentation (Yang, Xu, Zhou, Georgescu, Chen, Grbic, Metaxas & Comaniciu, 2017; Kamnitsas, Bai, Ferrante, McDonagh, Sinclair, Pawlowski, Rajchl, Lee, Kainz, Rueckert & Glocker, 2018), la classification (Hu, Tang, Eric, Chang, Fan, Lai & Xu, 2018; Madani, Moradi, Karargyris & Syeda-Mahmood, 2018), la synthèse d'images (Zhang, Yang & Zheng, 2018) et le recalage d'images (Fan, Cao, Xue, Yap & Shen, 2018) ont été explorés avec les GANs. Plusieurs variantes ont vu le jour. DCGAN (Radford, Metz & Chintala, 2016), est une variante dans laquelle le générateur est un réseau pleinement convolutif. Il en résulte des images plus réalistes qu'avec un équivalent MLP. Cela a permis de générer des images tout en conservant l'aspect spatial intrinsèque à ce type de données. Une seconde variante, LSGAN (Mao, Li, Xie, Lau, Wang & Paul Smolley, 2017a), utilise une fonction de coût des moindres carrés (*least squares*) sur le discriminateur afin de stabiliser l'apprentissage. Cette variante permet non seulement de classifier, mais également d'insérer une notion de distance par rapport à la frontière de décision apprise par le modèle.

Dans LSGAN, même si un échantillon est bien classifié, il sera tout de même pénalisé s'il est proche de la frontière de décision. Cette méthode permet notamment de contrer les problèmes courants de disparition du gradient et de saturation du discriminateur. Une troisième variante, CycleGAN (Zhu, Park, Isola & Efros, 2017), permet d'inclure une dépendance de cycle. Une paire de générateurs G_1 G_2 est jumelée à une paire de discriminateurs D_1 D_2. G_1 génère une image I_1 et G_2 génère une image I_2. Le coût de cycle agit de sorte que $G_2(G_1(I_1)) \approx I_1$ et $G_1(G_2(I_2)) \approx I_2$ où I_1 et I_2 sont issus d'ensembles de données différents. Ainsi, les modèles apprennent deux fonctions de mappage G et F, soit $G : X \rightarrow Y$ et $F : Y \rightarrow X$ où X sont des images réelles et Y des images générées. Le coût de cycle assure que l'image générée Y par un générateur ressemble à l'image réelle X issue du vrai ensemble de données du second domaine. Cette architecture permet de travailler avec des images x_i n'ayant aucune correspondance avec la vérité terrain y_i (non pairées) (figure 1.8). Dans cette figure, des données pairées (gauche) consistent en des échantillons $x_i, y_i{}_{i=1}^{N}$ où une correspondance entre x_i et y_i existe, alors que des données non pairées (droite) consistent en un ensemble de données source $x_i{}_{i=1}^{N}(x_i \in X)$ et cible $y_j{}_{j=1}^{N}(y_j \in Y)$ où aucune information n'est commune et ne permet d'identifier x_i à y_j [traduction libre] (Zhu *et al.*, 2017). Cette technique permet notamment d'entraîner un modèle avec plus de données afin d'effectuer une tâche précise comme une translation d'image à image ou la génération d'images haute résolution (*super-resolution*).

Les GANs sont également utilisés dans des applications d'apprentissage semi-supervisées où seulement une partie des données d'entraînement est étiquetée. Ce type de problème tente notamment de tirer profit de l'abondance de données non étiquetées d'un domaine afin d'augmenter la précision d'une tâche en particulier. Dans Dai, Yang, Yang, Cohen & Salakhutdinov (2017), une architecture de réseaux de neurones est présentée et permet de classifier des entrées non seulement entre des domaines existants, mais également dans un domaine généré. Ainsi, le générateur devient un *générateur complémentaire*. La fonction de coût devient alors :

$$\max_D \mathbb{E}_{x,y\sim\mathcal{L}} \log \mathcal{P}_D(y|x, y \leq K) + \mathbb{E}_{x\sim p} \log \mathcal{P}_D(y \leq K|x) + \mathbb{E}_{x\sim\mathcal{P}_G} \log \mathcal{P}_D(K+1|x) \quad (1.14)$$

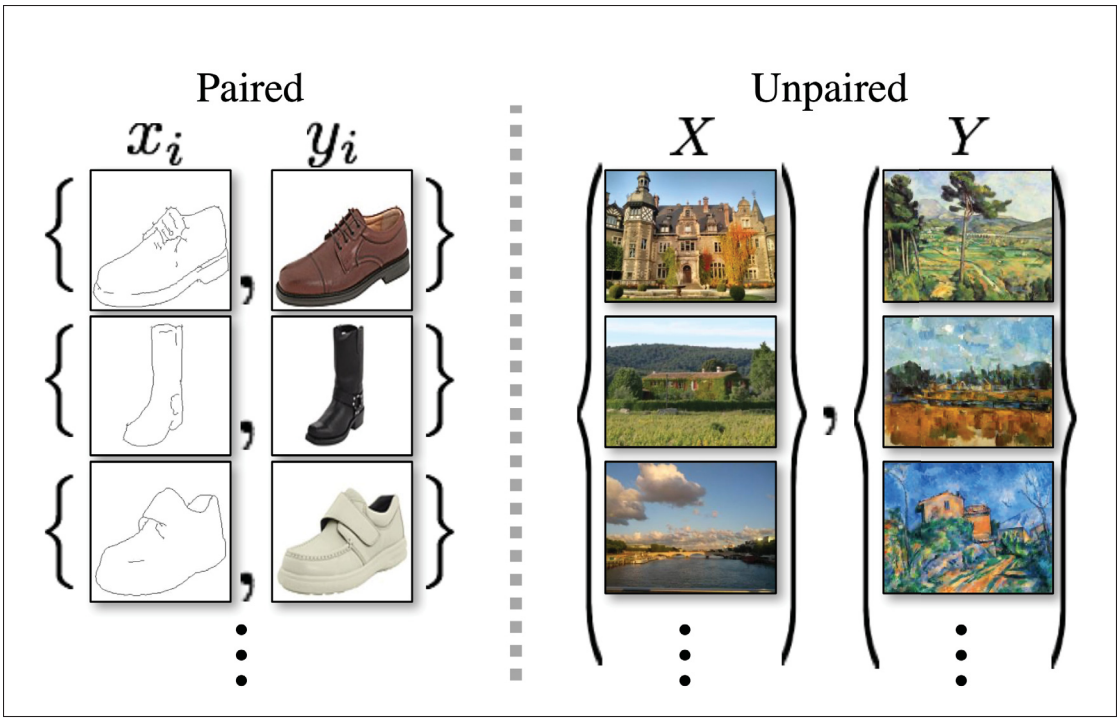

Figure 1.8 Schématique de données pairées. Tirée de Zhu *et al.* (2017)

où K représente le nombre de classes vraies et la classe $K + 1$ représente la classe dont les échantillons générés par le générateur appartiennent. Cette formulation permet au générateur de générer des échantillons complémentaires, ce qui encourage le discriminateur à tracer les frontières de décision dans une région à faible densité. Les images générées, qui sont intentionnellement de moindre qualité que les images réelles, sont réinjectées dans le réseau de classification afin d'améliorer la performance de celui-ci. De plus, la fonction de coût contraint le générateur à ne pas être *parfait*, ce qui améliore sa capacité de généralisation.

1.4 Apprentissage profond appliqué à la segmentation d'images médicales

L'apport des réseaux convolutifs à l'imagerie médicale est non-négligeable. Au courant de la dernière décennie, nombre de chercheurs ont appliqué différentes configurations et architectures de réseaux de neurones à des images médicales dans le but d'accroître la productivité des spécialistes de la santé dans des tâches complexes telle que la segmentation volumique. Cette

tâche est essentiellement une tâche de classification, où chaque voxel est classifié (associé) à une classe selon son intensité. Kamnitsas *et al.* (2017) a pavé la voie à la segmentation de lésions et de tumeurs cérébrales en proposant *DeepMedic*, un CNN 3D à double échelle suivit d'un champ aléatoire conditionnel (CRF) pleinement connecté. Dans cette configuration, les données en entrées suivent parallèlement deux canaux à échelles différentes, permettant d'extraire des caractéristiques à différentes échelles. Ainsi, le réseau permet de mieux incorporer dans la fonction apprise des caractéristiques locales et contextuelles cruciales à la localisation et la segmentation de lésion cérébrales. Le réseau élaboré est profond, mais utilise des noyaux de convolution plus petits, limitant ainsi le nombre d'hyperparamètres à optimiser et réduisant le surapprentissage. Le CRF permet de lisser le résultat de la segmentation en éliminant les régions aberrantes et isolées. Dans Kamnitsas *et al.* (2018), l'auteur propose une configuration ensembliste de plusieurs architectures différentes afin de réduire la variance au sein de la fonction de transfert apprise, augmentant ainsi la qualité de la prédiction et le pouvoir de généralisation du modèle. Dans Dolz, Gopinath, Yuan, Lombaert, Desrosiers & Ben Ayed (2019), la segmentation de tissus du cerveau auprès d'images IRM d'enfants a été réalisée à l'aide d'*HyperDense-Net*. En raison de son architecture densément connectée et l'utilisation conjointe de deux modalités d'imagerie IRM, le réseau produit une segmentation de qualité de la matière blanche, de la matière grise et des liquides cérébraux spinaux de ces images à faible contraste.

Au niveau de la normalisation d'images médicales, Drozdzal *et al.* (2018) propose un pipeline de réseaux de neurones consécutifs afin d'améliorer la qualité de segmentation d'images médicales. Les données en entrée se voient passer au travers d'un premier réseau se conformant à l'architecture U-Net, puis un second réseau FCN résiduel conception de l'auteur (Drozdzal *et al.*, 2016). Il est ainsi démontré que le premier réseau FCN permet de normaliser une image médicale alors que le deuxième apprend la fonction de classification des voxels. L'aspect consécutif de ces deux réseaux de neurones permet l'optimisation conjointe de la représentation intermédiaire et de la classification des voxels. Cette représentation intermédiaire a un histogramme transformé et d'allure gaussienne, qui est optimisé afin d'effectuer la segmentation sémantique. Cependant, en raison de cette transformation draconienne de l'histogramme, la représentation intermédiaire

est difficilement interprétable par un clinicien, visuellement difficile à consulter et peu réaliste, ce qui est pourtant d'une importance cruciale dans un environnement clinique. Toutefois, la normalisation des images facilite la tâche de segmentation, obtenant ainsi un score plus élevé sur le coefficient de similarité moyen (*Dice*), principale métrique utilisée pour la segmentation d'images médicales.

1.4.1 Contraintes d'entraînement et problématiques liées au domaine médical

En raison de sa complexité de calcul et son nombre de dimensions, l'apprentissage automatique appliqué à l'imagerie médicale amène son lot de contraintes. Les sections suivantes décrivent les principales rencontrées lors de la réalisation de ce travail.

1.4.1.1 Contrainte de mémoire et décomposition en segments (*patches*)

Afin d'accélérer l'apprentissage des réseaux de neurones, les chercheurs utilisent des accélérateurs graphiques dont l'architecture permet l'exécution parallèle d'une instruction avec plusieurs données (SIMD). Toutefois, la mémoire est très limitée sur ces dispositifs. La déclaration en mémoire de modèles de réseaux de neurones profonds convolutifs tels que U-Net ou ResNet requiert la copie en mémoire de millions de variables à virgules flottantes 32 bits (Float32). Ainsi, une image volumétrique médicale complète peut rarement tenir en mémoire à titre d'entrée de ces réseaux de neurones, le nombre de cartes de caractéristiques requis pour supporter une telle résolution étant trop grand. Pour entrainer un modèle, on doit alors décomposer l'image médicale 3D en plus petits segments (*patch*) qui se chevauchent. Ainsi, on passera d'une image de dimension $256x256x192$ à des milliers de segments de taille $32x32x32$. Un pas de 4 dans chaque axe permet d'avoir des segments se chevauchant, augmentant drastiquement le nombre de données et permettant ainsi d'avoir des résultats de segmentation plus précis. Ces segments sont par la suite mis en lots (*mini-batch*) afin d'effectuer l'entraînement selon une descente de gradient stochastique tout en respectant la mémoire disponible embarquée sur les accélérateurs graphiques.

1.4.1.2 Problème de débalancement des classes

Le déséquilibre des classes en segmentation sémantique d'images médicales se traduit par le fait que la distribution des classes (étiquettes) des échantillons au sein d'un ensemble de données n'est pas identique d'une classe à l'autre. Ainsi, on peut se retrouver avec une classe sous-représentée au sein de la fonction de classification apprise. Les causes du déséquilibre des classes peuvent être variées. Elles vont du biais d'échantillonnage à l'erreur lors de la prise de mesures. La cause peut aussi faire partie du domaine du problème. Dans le cas d'images médicales, le déséquilibre des classes est directement lié à l'hétérogénéité de la matière échantillonnée (corps humain). Dans le cas du cerveau par exemple, les voxels associés aux liquides cérébraux spinaux sont beaucoup moins nombreux que ceux associés à la matière grise. Ainsi, un classificateur peut être biaisé vers la classe majoritaire. Le problème est encore plus flagrant lorsque la tâche de segmentation est axée sur les lésions ou tumeurs cérébrales. Ces petites régions occupent, par rapport au reste du cerveau, un volume très faible. Il en résulte des performances de segmentation moindre.

Afin de contourner ce problème, trois stratégies ressortent de la littérature. La première est le sous-échantillonnage des classes majoritaires. La deuxième est le suréchantillonnage des classes minoritaires (Pereira, Pinto, Alves & Silva, 2016) où les données qui sont constituées des classes cibles sont augmentées à l'aide de transformations affines. Dans Kamnitsas *et al.* (2017), le problème de déséquilibre des classes est notamment attaqué en échantillonnant 50% des voxels appartenant à une classe du premier plan (*foreground*), l'autre moitié étant des voxels appartenant à l'arrière-plan. La troisième stratégie est d'intervenir directement sur le processus d'apprentissage par l'utilisation d'une fonction de coût spécifique au problème. À cet effet, on peut notamment pondérer la fonction de coût en fonction de la distribution de chaque classe. On pondère généralement la fonction de coût de manière inversement proportionnel à la fréquence de la classe.

1.4.1.3 Problème de manque de données

Le manque de données est un problème inévitable dans le domaine de l'apprentissage automatique appliqué à l'imagerie médicale. Les méthodes d'apprentissage donnant les meilleurs résultats de segmentation sont dites *supervisées* Kamnitsas *et al.* (2017); Çiçek *et al.* (2016); Shelhamer *et al.* (2017); Pereira *et al.* (2016). Les algorithmes entraînés de cette manière reposent intégralement sur les images annotées manuellement par des professionnels. Ce processus est coûteux et monopolise beaucoup de temps de la part de l'expert. De plus, cette segmentation manuelle repose principalement sur l'expérience de l'expert et son habileté à segmenter les différents tissus. Ainsi, cela pose un problème de variance entre différents experts. Afin de réduire cette variance, les ensembles de données sont souvent revalidés par plusieurs experts, augmentant ainsi considérablement les coûts. Ainsi, un phénomène de rareté de ces données de qualité s'est créé dans le domaine médical.

Afin de contourner cette limitation inhérente au domaine, différentes techniques ont été explorées, notamment l'apprentissage par transfert (Raghu, Zhang, Kleinberg & Bengio, 2019). Un autre moyen est de créer artificiellement des données à partir de celles qui sont existantes dans l'ensemble de données d'origine en leur faisant subir une transformation géométrique comme une rotation, un effet miroir (Havaei, Davy, Warde-Farley, Biard, Courville, Bengio, Pal, Jodoin & Larochelle, 2017), une translation ou une déformation élastique (Çiçek *et al.*, 2016). Toutefois, en fonction de la tâche, cela n'augmente pas le pouvoir statistique du modèle puisque celui-ci apprend toujours sur une population relativement restreinte de sujets.

Une autre stratégie est d'utiliser une approche semi-supervisée (Feyjie, Azad, Pedersoli, Kauffman, Ayed & Dolz, 2020). Dans cette technique, l'ensemble de données contient deux types de données, soit les données annotées et les données non annotées. Dans le dernier cas, elles sont utilisées afin de raffiner le modèle entraîné. La tâche peut non seulement être de prédire les étiquettes sur des données acquises dans le futur (*inductive semi-supervised learning*) ou de prédire les étiquettes sur des données actuellement disponibles, mais non étiquetées (*transductive semi-supervised learning*) (Zhu & Goldberg, 2009).

1.4.2 Adaptation de domaine

L'adaptation de domaine repose sur l'apprentissage par transfert (*transfer learning*), plus spécifiquement le *transductive transfer learning*.

Un domaine $\mathcal{D}$ consiste en un espace de caractéristiques $\mathcal{X}$ ainsi qu'une distribution de probabilités $P(X)$ où $X = x_1, ..., x_n \in \mathcal{X}$. Une tâche $\mathcal{T}$ consiste en un espace de caractéristique $\mathcal{Y}$ et une fonction prédictive $f(\cdot)$ (Wang & Deng, 2018). Si on considère un problème à deux domaines, nous aurons ainsi :

- $D^s = X^s, P(X)^s$
- $D^t = X^t, P(X)^t$

où $P(X)^s \neq P(X)^t$, c'est-à-dire que D^s accuse une translation (*shift*) par rapport à D^t.

On parle de *domain shift* lorsque D^s et D^t ont des distributions de données différentes. Les techniques d'adaptation de domaine tentent d'attaquer ce problème qu'est le *domain shift* en trouvant une fonction de mappage entre la distribution source et la distribution cible. Un autre moyen est également de trouver un mappage Z entre les domaines D^s et D^t, où les trois domaines sont alignés. Ce mappage devient alors *domain-agnostic* et sa principale fonction est de maximiser la similarité des distributions entre les images qui composent le domaine source et le domaine cible.

Une autre particularité de ce problème réside dans le fait que des caractéristiques des deux domaines se retrouvent dans les deux distributions. Par exemple, la matière blanche se retrouve autant dans l'ensemble de données *iSEG* que *MRBrainS*. Toutefois, la distribution des intensités de ces deux ensembles de données pour la matière blanche n'est pas la même (figure 3.1). Cette différence au sein des intensités n'a aucune valeur discriminante et ne devrait pas affecter la tâche de segmentation, dans ce cas-ci la segmentation de la matière blanche du cerveau. Le mappage au domaine Z doit donc être invariant à cette différence au sein des distributions. La distribution des caractéristiques extraites dans un processus d'entraînement ayant deux domaines ou plus

devrait être indiscernable. Ce concept a d'abord été élaboré dans Ganin & Lempitsky (2015) qui propose une architecture permettant de rendre indiscernable par un classificateur de domaines des caractéristiques précédemment extraites par un réseau extracteur de caractéristiques (*feature extractor*). Afin d'obtenir des caractéristiques qui sont invariantes aux domaines, les paramètres Θ_f de la fonction de classification du classificateur de domaines sont optimisés de sorte à maximiser la fonction de coût. Ainsi, les distributions des deux domaines seront par définition les plus similaires possible. Le modèle permet également la classification des étiquettes des images par le biais d'un troisième réseau, soit le *label predictor* qui minimise le coût de classification.

1.4.3 Harmonisation de données

Le principe d'harmonisation des données est défini comme la suppression explicite des effets liés aux sites dans des données multisites (traduction libre, Pomponio *et al.* (2020)). Il regroupe tout effort permettant de combiner des ensembles de données afin de permettre à un utilisateur d'avoir une vision globale, homogène et comparable des données issues de plusieurs sites ou études. Cette approche fut notamment utilisée afin d'accroître la taille des ensembles de données d'études statistiques. Dans Logue, van Rooij, Dennis, Davis, Hayes, Stevens, Densmore, Haswell, Ipser, Koch, Korgaonkar, Lebois, Peverill, Baker, Boedhoe, Frijling, Gruber, Harpaz-Rotem, Jahanshad, Koopowitz, Levy, Nawijn, O'Connor, Olff, Salat, Sheridan, Spielberg, van Zuiden, Winternitz, Wolff, Wolf, Wang, Wrocklage, Abdallah, Bryant, Geuze, Jovanovic, Kaufman, King, Krystal, Lagopoulos, Bennett, Lanius, Liberzon, McGlinchey, McLaughlin, Milberg, Miller, Ressler, Veltman, Stein, Thomaes, Thompson & Morey (2017), une étude statistique se basant sur 16 cohortes provenant de 5 pays différents, a montré un lien entre des données volumétriques de l'anatomie du cerveau et un trouble de stress post-traumatique. Ce lien fut démontré en grande partie par l'augmentation du pouvoir statistique rendu possible par l'ensemble de données plus grand. Ceci est notamment rendu possible par l'exécution d'un protocole d'analyse harmonisé sur tous les sites. Shinohara, Oh, Nair, Calabresi, Davatzikos, Doshi, Henry, Kim, Linn, Papinutto, Pelletier, Pham, Reich, Rooney, Roy, Stern, Tummala, Yousuf, Zhu, Sicotte & Bakshi (2017) montre que l'acquisition d'images médicales depuis plusieurs sites distincts résulte en une

erreur systématique des analyses volumétriques, et ce même si une attention particulière est portée à l'harmonisation des protocoles d'acquisition ou au manufacturier de l'imageur. Il en résulte un biais sévère dans les analyses volumétriques de lésions, de même que le volume de la matière blanche et matière grise. De plus, la multiplication des sites dans une même étude peut introduire un biais non linéaire relié à l'âge des patients, ce qui affecte les régions d'intérêts dans le cerveau. Afin de prévenir les disparités entre les sites, le *UK7T Network* a été fondé au Royaume-Uni afin de tenter l'harmonisation manuelle des images structurelles et fonctionnelles du cerveau par des protocoles d'acquisition, des séquences d'imagerie et des processus de calibration. Cette harmonisation est appliquée manuellement sur tous les appareils 7 Tesla déployés au Royaume-Uni. En utilisant la procédure standardisée, l'image résultante de l'acquisition d'un imageur s'harmonise au mieux avec les autres images issues des imageurs 7 Tesla déployés sur les autres sites (Clarke, Mougin, Driver, Rua, Morgan, Asghar, Clare, Francis, Wise, Rodgers, Carpenter, Muir & Bowtell, 2020).

L'utilisation d'un ensemble de données issu de plusieurs cohortes étalées sur plusieurs sites peut introduire des différences non linéaires dans une région d'intérêt (ROI) du cerveau (Pomponio *et al.*, 2020). L'harmonisation des données sert alors à soustraire de cet ensemble de données ces différences non linéaires. Cette technique a notamment été utilisée sur l'ensemble de données LIFESPAN, notamment afin de retirer les différences de l'échelle et des mesures entre les sites qui constituent l'ensemble de données. Toutefois, cette harmonisation intervient tard dans le pipeline d'exécution et est une opération faite hors-ligne, soit en prétraitement. Cela ajoute donc une étape supplémentaire dans le pipeline d'exécution. Des approches d'harmonisation de données basées sur l'apprentissage profond ont aussi été élaborées dans des travaux récents. Dewey, Zhao, Reinhold, Carass, Fitzgerald, Sotirchos, Saidha, Oh, Pham, Calabresi, van Zijl & Prince (2019) a proposé une architecture basée sur un réseau pleinement convolutif 3D afin d'harmoniser le contraste des images issues de deux protocoles d'imagerie différents. Cette méthode a mis en évidence une quantification des volumes d'intérêts plus précise. Cependant, pour arriver à cette conclusion, la méthode a dû être entrainée avec des images pairées d'un même sujet acquis par les deux protocoles d'imagerie, ce qui est dans la pratique très difficile à avoir et peu productif

d'un point de vue clinique. Dans Modanwal *et al.* (2020), une architecture adoptant CycleGAN est utilisée avec d'harmoniser des images structurelles de seins issus de deux types d'imageurs. La méthode utilise des données non pairées avec deux paires de générateurs et discriminateurs afin de contourner les limitations communes des algorithmes de translations d'images qui requièrent des données pairées. En utilisant cette architecture, les auteurs sont en mesure de faire des prédictions bidirectionnelles entre les domaines harmonisés. Une attention particulière a été apportée à la taille des segments d'images utilisés afin de préserver les structures fines caractéristiques des images médicales de seins. Alors que les résultats prouvent visuellement une harmonisation réaliste des images, la tâche explicite réside en l'harmonisation elle-même, et non pas la segmentation. La méthode ne tente pas non plus d'augmenter la précision de cette tâche. De plus, la tâche est limitée à deux domaines où des imageurs GE Healthcare et Siemens sont utilisés.

CHAPITRE 2

NORMALISATION RÉALISTE D'IMAGES POUR DE LA SEGMENTATION MULTI-DOMAINES

Dans ce chapitre, nous proposons une nouvelle méthode permettant l'apprentissage d'une fonction de normalisation d'images médicales sur plusieurs ensembles de données à la fois tout en conservant l'interprétabilité et le réalisme de l'image intermédiaire. Une partie de ce travail a mené à une publication acceptée dans le cadre de la conférence ISBI 2020 (Delisle, Anctil-Robitaille, Desrosiers & Lombaert, 2020).

2.1 Motivation

L'accessibilité des données d'imagerie médicale afin d'entraîner des réseaux de classification de manière supervisée demeure un problème majeur. Le coût nécessaire afin de produire une carte de segmentation par un professionnel étant exorbitant, les ensembles de données sont par conséquent relativement petits, souvent composés seulement d'une dizaine d'images. La capacité de généralisation se voit ainsi fortement impactée, limitant l'utilité du modèle entraîné.

La normalisation des images est une pierre angulaire dans l'analyse d'images médicales. Les approches conventionnelles de normalisation sont couramment utilisées sur un seul ensemble de données à la fois. Cette stratégie a par contre son lot d'inconvénients, y compris l'incapacité de tirer profit de l'information conjointe disponible au sein de plusieurs ensembles de données. Conséquemment, ignorer cette information conjointe a un impact direct sur la performance de segmentation des algorithmes d'apprentissage. Ainsi, ce travail propose de revisiter l'approche conventionnelle de normalisation d'images en proposant l'apprentissage d'une fonction de normalisation sur plusieurs ensembles de données. Dès lors, l'information conjointe des ensembles de données pertinentes à une tâche en particulier est conservée et mise à profit tout en optimisant cette tâche.

2.2 Méthodologie

2.2.1 Proposition

La méthode proposée repose sur une architecture de réseaux de neurones antagoniste composée de trois réseaux. Le premier réseau de neurones est un réseau pleinement convolutif permettant la génération d'images normalisées. Ce générateur G consiste en une version légèrement modifiée de l'architecture 3D U-Net présentée à la section 1.2.9. Le changement réside principalement dans la section décodeur du réseau où les opérations de déconvolution utilisées dans le modèle original ont été remplacées par une opération de suréchantionnage avec un facteur de 2 avec une interpolation basée sur les voisins les plus proches. Cette méthode d'interpolation est choisie principalement à cause de sa rapidité et sa faible empreinte mémoire. Cette modification est apportée à chaque niveau du décodeur du modèle où s'effectue également la concaténation des cartes de caractéristiques. Ce changement permet de réduire le nombre de paramètres du modèle et donne lieu à une utilisation moindre de la mémoire GPU. L'empreinte mémoire réduite permet la cohabitation des trois modèles de l'architecture proposée sur un GPU ayant 8 giga-octets de mémoire. L'encodeur est le même que dans Çiçek *et al.* (2016) et consiste en l'alternance de couches de convolutions et de *max-pooling*. Le réseau utilise également des connexions entre l'encodeur et le décodeur, permettant de récupérer des cartes de caractéristiques de plus haute résolution. Ayant l'avantage d'être un réseau pleinement convolutif, ce réseau permet de générer directement une image en sortie, conservant l'aspect spatial de l'image au travers des couches du réseau. La dernière convolution du réseau, soit une convolution ayant un noyau de $1 \times 1 \times 1$, a en sorti un seul canal suivit d'une fonction d'activation sigmoïde. Cette particularité permet d'avoir une image en tons de gris comme sortie. Le modèle accepte un segment d'image $x \in \mathbb{R}^{|\Omega|}$ où Ω est un ensemble de segments de taille $1 \times 32 \times 32 \times 32$ avec des intensités dans une plage indéfinie. Dans notre cas, les entrées ont préalablement subi une normalisation min-max. Ainsi, les voxels des images en entrée ont une valeur comprise dans la plage $[0, 1]$. Le modèle transforme cet ensemble de segments en une image normalisée inter-domaine $\widehat{x} = G(x)$.

Le deuxième réseau S partage la même architecture 3D U-Net que le premier et permet la segmentation d'une image médicale. Afin d'effectuer la classification finale des voxels, la dernière convolution du réseau a un noyau de $1 \times 1 \times 1$ et a pour sortie quatre canaux représentant les quatre classes du problème (arrière-plan, matière blanche, matière grise, liquides cérébraux spinaux). Cette dernière couche effectue la classification finale et produit la carte de probabilités pour chaque classe $S(\hat{x})$. Une fonction de coûts basée sur la relaxation continue de l'indice *Dice* (Milletari *et al.*, 2016; Carass, Roy, Gherman, Reinhold, Jesson, Arbel, Maier, Handels, Ghafoorian, Platel, Birenbaum, Greenspan, Pham, Crainiceanu, Calabresi, Prince, Roncal, Shinohara & Oguz, 2020) est utilisé afin d'optimiser le réseau :

$$\mathcal{L}_{seg}(\mathbf{s}, \mathbf{y}) \;=\; 1 \;-\; \frac{\epsilon \;+\; 2 \sum_c \; \omega_c \sum_{v \in \Omega} s_{v,c} \cdot y_{v,c}}{\epsilon \;+\; \sum_c \; \omega_c \sum_{v \in \Omega} (s_{v,c} + y_{v,c})}, \tag{2.1}$$

où $s_{v,c} \in [0, 1]$ est la sortie *softmax* de S pour le voxel v et la classe c, ω_c est le poids attribué à une classe c en fonction de son importance dans l'ensemble de données, $y_{v,c}$ est l'étiquette correspondant à la vérité terrain et ϵ est une constante afin d'éviter la division par zéro. Puisque les classes de la segmentation ne sont pas balancées, ω_c vient corriger le débalancement en venant attribuer un poids plus important aux classes où le nombre de voxels est moins important.

Le réseau générateur et le réseau de segmentation ne sont pas liés spécifiquement à l'architecture 3D U-Net, celle-ci étant décrite au tableau 2.1. Dans ce dernier, l'activation finale est l'opération Sigmoïde dans le cas du générateur et softmax à 4 dimensions (classes) en sortie pour le réseau de segmentation. Cependant, la méthode proposée pourrait s'adapter à n'importe quel réseau de neurones de type pleinement convolutif (FCN).

Le troisième réseau D est un réseau de neurones convolutif (CNN) de classification. Le nom attribué à ce réseau pour le reste de ce travail est *discriminateur*. Tout comme les précédents composants de la proposition, l'architecture de ce réseau peut-être variable. Ainsi, une version volumétrique de ResNet (He *et al.*, 2016) telle que décrite à la section 1.2.10 a été explorée, de même que l'architecture du discriminateur présente dans DCGAN (Radford *et al.*, 2016).

Dans le premier cas, la version ResNet-18 fut utilisée. Il s'agit de la plus petite version du réseau résiduel, permettant de limiter le surapprentissage et l'empreinte mémoire. Dans le deuxième cas, il s'agit d'un réseau de convolution peu profond comportant 4 couches de convolution suivies d'une couche pleinement connectée. Le nombre de filtres à chaque couche est réduit, ce qui limite le nombre d'hyperparamètres à optimiser. Afin d'effectuer la classification des domaines d'imagerie, la provenance de chaque segment d'image 3D doit être définie. Ainsi, chaque segment est étiqueté par une étiquette de domaine $z_i \in 1, ..., K$, où K est le nombre total d'ensembles de données qui composent le problème. Cela détermine l'ensemble de données d'origine du segment. Le discriminateur apprend quant à lui une frontière de décision d'un problème à $(K+1)$ classes, où une classe par domaine d'imagerie est prédite et une $(K+1)$-ème classe pour les images appartenant au domaine généré par le générateur. Cette formulation du discriminateur permet à celui-ci d'apprendre à distinguer les images des différents domaines d'imagerie (par exemple, les sites d'acquisition ou les protocoles d'imagerie) *de même* que les images irréalistes classifiées dans la classe $K+1$. Ainsi, le rôle du discriminateur, dont l'architecture est présentée aux Tableaux 2.2 et 2.3, est de transmettre l'information nécessaire au générateur afin que celui-ci produise une image réaliste et invariante au domaine. Dans ces tableaux, l'activation finale est l'opération Sigmoïde avec 3 ou 4 classes en sortie en fonction du nombre d'ensembles de données avec lequel on entraîne le réseau. L'entropie croisée a été choisie afin d'entraîner le réseau de classification :

$$\mathcal{L}_{dis}\big(D(\mathbf{x}),\, z\big) \;=\; -\log\, D_z(\mathbf{x}), \tag{2.2}$$

où $D_z(\mathbf{x})$ est la probabilité pour la classe z. On peut noter que puisque $\sum_{z=1}^{K+1} D_z(\mathbf{x}) = 1$, le coût pour la classe $(K+1)$ correspondant aux segments d'images générés peut s'écrire sous la forme :

$$\mathcal{L}_{dis}\big(D(\mathbf{x}),\, K+1\big) \;=\; -\log\left(1 - \sum_{z=1}^{K} D_z(\mathbf{x})\right).$$

Bien qu'il est possible d'assurer le réalisme et l'invariance des domaines des images normalisées à l'aide de deux discriminateurs distincts, l'utilisation d'un seul réseau jouant ce rôle apporte un lot d'avantages importants. Premièrement, cela évite un problème d'instabilité qui découle de l'entraînement de discriminateurs avec des fonctions de coûts qui se rivalisent. Au lieu de traiter le domaine de l'image et le réalisme comme des propriétés indépendantes de l'image, le modèle prédit ces propriétés de manière conjointe au sein d'un seul réseau. De plus, le fait de n'avoir qu'un seul réseau permet une consommation moindre des ressources computationnelles et rend la méthode beaucoup plus simple. Finalement, notre modèle antagoniste, sans l'addition du coût de la segmentation et appuyé sur de légères hypothèses, mène à une solution optimale où les images normalisées sont générées à partir de la distribution moyenne des images réelles. Ceci peut être démontré par le théorème 1 :

Théorème 1. *Soit* $p_r(\mathbf{x} \mid z)$ *et* $p_g(\mathbf{x} \mid z)$ *les probabilités que* $\mathbf{x}$ *est une image réelle ou générée, respectivement, d'un ensemble de données* z*. Le problème d'optimisation de l'algorithme minimax de l'Eq. (2.3) sans le terme représenté par le coût de la segmentation correspond à la minimisation de la divergence entre* $p_g(\mathbf{x} \mid z) \forall z$ *et la distribution moyenne des images réelles* $\overline{p}_r(\mathbf{x}) = \frac{1}{k} \sum_{z=1}^{K} p_r(\mathbf{x} \mid z)$.

(*Voir* ANNEXE I, p. 75)

Chaque domaine d'imagerie est identifié par un vecteur répondant à l'encodage 1 parmi N (*one-hot encoding*), permettant ainsi au réseau d'effectuer la classification du domaine.

La figure 2.1 illustre notre proposition d'architecture. Un premier réseau FCN, le *générateur G*, a pour entrée un segment d'image non-normalisé et génère un segment normalisé. Ce dernier est ensuite l'entrée du réseau de segmentation (*segmenter S*) qui effectue la classification de chaque voxel. Le troisième réseau, le *discriminateur D*, applique la contrainte de réalisme au segment d'image normalisé en tentant de classifier son domaine d'imagerie. L'algorithme apprend ainsi une fonction de normalisation basée sur les différences entre les ensembles de données qui lui sont présentés.

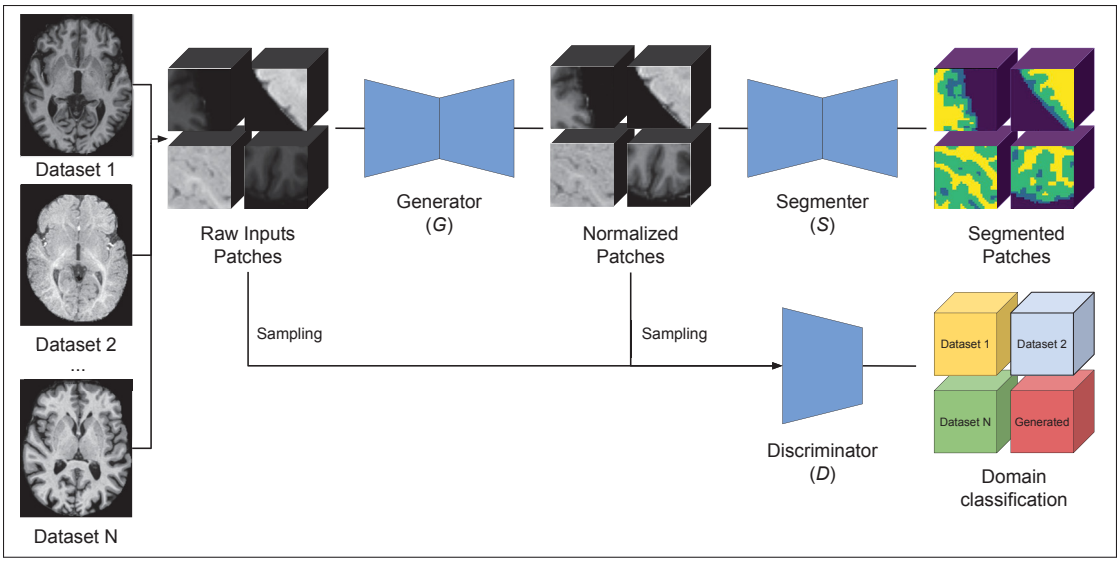

Figure 2.1 Architecture de normalisation d'images proposée

Tableau 2.1 Architecture pleinement convolutive pour le réseau générateur et le réseau de segmentation.

Couche	Résolution de sortie	Largeur de sortie
Conv.1 + BatchNorm + Leaky ReLU	32 ×32 ×32	32
Conv.2 + BatchNorm + Leaky ReLU	32 ×32 ×32	64
MaxPool	16 ×16 ×16	
Conv.3 + BatchNorm + Leaky ReLU	16 ×16 ×16	64
Conv.4 + BatchNorm + Leaky ReLU	16 ×16 ×16	128
MaxPool	8 ×8 ×8	
Conv.5 + BatchNorm + Leaky ReLU	8 ×8 ×8	128
Conv.6 + BatchNorm + Leaky ReLU	8 ×8 ×8	256
MaxPool	4 ×4 ×4	
Conv.7 + BatchNorm + Leaky ReLU	4 ×4 ×4	256
Suréchantillonnage + Concaténation	8 ×8 ×8	256 + 512
Conv.8 + BatchNorm + Leaky ReLU	8 ×8 ×8	256
Suréchantillonnage + Concaténation	16 ×16 ×16	128 + 256
Conv.9 + BatchNorm + Leaky ReLU	16 ×16 ×16	128
Suréchantillonnage + Concaténation	32 ×32 ×32	64 + 128
Conv.10 + BatchNorm + Leaky ReLU	32 ×32 ×32	64
Conv.11 + BatchNorm + Leaky ReLU	32 ×32 ×32	64
Conv.12 + Activation finale	32 ×32 ×32	1 (4)

Tableau 2.2 Architecture DCGAN convolutive pour le réseau de classification des domaines (discriminateur)

Couche (composants)	**Résolution de sortie**	**Largeur de sortie**
Conv.1 + Leaky ReLU + Dropout	32 ×32 ×32	16
Conv.2 + Leaky ReLU + Dropout + BatchNorm	32 ×32 ×32	32
Conv.3 + Leaky ReLU + Dropout + BatchNorm	32 ×32 ×32	64
Conv.4 + Leaky ReLU + Dropout + BatchNorm	32 ×32 ×32	128
Linéaire + Sigmoïde	-	3 (4)

Tableau 2.3 Architecture ResNet-18 convolutive pour le réseau de classification des domaines (discriminateur)

Couche (nom)	**Type de bloc**	**Résolution de sortie**	**Largeur de sortie**	**Nombre de répétitions**
Down 1	Conv 7x7x7	16 ×16 ×16	64	1
Down 2	Simple	8 ×8 ×8	128	2
Down 3	Simple	4 ×4 ×4	256	2
Down 4	Simple	2 ×2 ×2	512	2
Pooling	Adaptative Average Pooling	1 ×1 ×1	512	1
Classification	Linéaire + Softmax	-	3 (4)	1

2.2.2 Entraînement antagoniste

L'architecture précédemment explicitée est entraînée de manière antagoniste. Elle optimise la fonction de coût suivante :

$$\begin{aligned}\min_{G,S}\ \max_{D}\ \mathcal{L}(G,S,D) \;=\; & \mathbb{E}_{\mathbf{x},\mathbf{y}}\left[\mathcal{L}_{seg}\big(S(G(\mathbf{x})),\mathbf{y}\big)\right] \\ & -\lambda\,\mathbb{E}_{\mathbf{x},z}\Big[\mathcal{L}_{dis}\big(D(\mathbf{x}),z\big)\;+\;\mathcal{L}_{dis}\big(D\big(G(\mathbf{x})\big),K{+}1\big)\Big]\end{aligned} \tag{2.3}$$

où $\mathcal{L}_{seg}$ et $\mathcal{L}_{dis}$ sont les fonctions de coûts du réseau de segmentation S dans l'Eq. (2.1) et du discriminateur D définis dans l'Eq. (2.2).

Le paramètre λ contrôle le niveau de réalisme de l'image générée. En utilisant $\lambda = 0$, le modèle devient similaire à Drozdzal *et al.* (2018) où aucune contrainte de réalisme n'est appliqué sur le générateur. Celui-ci est alors libre de générer une image optimisée pour la tâche de segmentation. En revanche, pour un λ plus grand, le modèle devient similaire à un classificateur antagoniste de domaines présenté dans Ciga *et al.* (2019).

Algorithme 2.1 L'algorithme d'entraînement de notre méthode antagoniste avec la mise à jour des paramètres du réseau générateur, du discriminateur et du réseau de segmentation.

Input : Training set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i, z_i\}_{i=1}^{|\mathcal{D}|}$
Input : Batch size m, number epochs, iterations and steps (n_{epochs}, n_{iter}, n_{steps}), and learning rates η_G, η_S, η_D ;
Output : Network parameters $\theta_G, \theta_S, \theta_D$;

Randomly initialize network parameters $\theta_G, \theta_S, \theta_D$;
for $epoch = 1, \ldots, n_{epochs}$ **do**
  for $iteration = 1, \ldots, n_{iter}$ **do**
    for $step = 1, \ldots, n_{steps}$ **do**
      Sample batch of m examples from all domains $\{(\mathbf{x}_i, z_i)\}_{i=1}^{m}$;
      Update the Discriminator D :
      $\theta_D \leftarrow \theta_D - \frac{\eta_D}{m} \sum_{i=1}^{m} \Big(\nabla_D \mathcal{L}_{dis}\big(D(\mathbf{x}_i), z_i\big) + \nabla_D \mathcal{L}_{dis}\big(D(G(\mathbf{x}_i)), K{+}1\big) \Big)$;
    end for
    Sample batch m examples from all domains $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{m}$;
    Update the Segmenter S and Generator G :
    $\theta_S \leftarrow \theta_S - \frac{\eta_S}{m} \sum_{i=1}^{m} \nabla_S \mathcal{L}_{seg}\big(S(G(\mathbf{x}_i)), \mathbf{y}_i\big)$;
    $\theta_G \leftarrow \theta_G - \frac{\eta_G}{m} \sum_{i=1}^{m} \Big(\nabla_G \mathcal{L}_{seg}\big(S(G(\mathbf{x}_i)), \mathbf{y}_i\big) - \lambda \nabla_G \mathcal{L}_{dis}\big(D(G(\mathbf{x}_i)), K{+}1\big) \Big)$;
  end for
  Adjust learning rates η_G, η_S, η_D ;
end for
return $\theta_G, \theta_S, \theta_D$;

Comme dans une méthode antagoniste standard, l'entraînement s'effectue en deux étapes qui s'alternent. Dans un premier temps, les paramètres de S et G sont mis à jour. Dans un deuxième temps, les paramètres de D sont mis à jour. Ceux-ci sont mis à jour $n_{steps} = 3$ afin de maintenir une solution quasi optimale de la classification des domaines alors que G est mis à jour moins fréquemment. La procédure est présentée explicitement dans l'Algorithme 2.1

L'optimiseur Adam (Diederik & Ba, 2014) est utilisé. Cet optimiseur permet de calculer de manière individuelle un taux d'apprentissage pour chaque paramètre Θ constituant le modèle tout en permettant de minimiser une valeur d'une fonction $f(\Theta)$. L'optimisation est stochastique puisqu'elle est réalisée à partir de lots de données *(mini-batch)*. Le taux d'apprentissage est basé sur une estimation du premier (moyenne) et second moment (variance) des gradients. Adam met à jour une moyenne mobile exponentielle des gradients et du gradient au carré alors que deux hyperparamètres, β_1 et $\beta_2 \in [0, 1)$ contrôlent le taux de décroissance de ces moyennes mobiles. Ces particularités permettent notamment à Adam de faire converger le réseau plus rapidement en réduisant le nombre d'itérations requis.

2.2.3 Stratégie d'apprentissage

2.2.3.1 Taux d'apprentissage

Bien que l'optimiseur Adam puisse calculer un taux d'apprentissage indépendant pour chaque paramètre, la borne supérieure du taux d'apprentissage ne change pas. Ainsi, un paramètre Θ du modèle pourrait se voir attribuer la valeur maximale du taux d'apprentissage. Or, cette valeur pourrait ne pas être optimale, menant à une non-convergence du modèle. En adoptant une stratégie de taux d'apprentissage dynamique, cela permet au modèle de mieux converger vers une solution optimale tout en respectant un temps d'entraînement raisonnable.

Par conséquent, une stratégie de mise à jour du taux d'apprentissage fut adoptée. Il s'agit d'une stratégie par plateau. Ainsi, si la valeur de la fonction de coût en validation stagne ou augmente après une patience de 7 époques d'entraînement, la valeur du taux d'apprentissage sera diminuée d'un facteur 10 afin de permettre une meilleure convergence du modèle. Cette stratégie fut implémentée pour le réseau discriminateur. Quant au réseau générateur G et au réseau de segmentation S, les deux voient leur taux d'apprentissage descendre en synchronisme aux époques 50 et 70 dans le cas d'un apprentissage à deux ensembles de données et à l'époque 50 dans le cas de trois ensembles de données. Cela permet aux deux réseaux d'apprendre en synchronisme.

2.2.3.2 Augmentation des données

Afin de tester l'algorithme dans différentes conditions et utilisations, les données ont été augmentées et modifiées. Le générateur s'est vu présenter, avec une probabilité de 33%, des images auxquelles un bruit Ricien et un champ de polarisation (*bias field*). Ce champ est généralement un champ uniforme s'appliquant à toute l'image modifiant les intensités des tissus. Ce champ fut modelé à partir d'une fonction linéaire et ajouté de manière aléatoire dans les coins de l'image, là où il apparait le plus souvent.

Le bruit Ricien est quant à lui modéliser de la sorte (Gudbjartsson & Patz, 1995) :

$$p_m(M) = \frac{M}{\sigma^2} e^{-\frac{\left(M^2+A^2\right)}{2\sigma^2}} I_0\left(\frac{A \cdot M}{\sigma^2}\right) \tag{2.4}$$

où M est l'intensité du voxel, A est le signal original non dégradé, I_0 et la fonction modifiée à l'ordre zéro de Bessel et σ est la variance du bruit gaussien. Le rapport signal sur bruit (*RSB*) est prédéfini à 60 dB. Avec ce RSB, le bruit est clairement visible sur l'image.

CHAPITRE 3

EXPÉRIMENTATIONS

3.1 Données

Trois ensembles de données ont été sélectionnés afin de tester notre méthode de normalisation réaliste d'images médicales multi-domaines. Ces trois ensembles de données sont destinés à la segmentation d'images médicales en 4 classes distinctes : l'arrière-plan, la matière blanche (*white matter - WM*), la matière grise (*grey matter - GM*) et les liquides cérébraux spinaux (*cerebrospinal fluids - CSF*).

3.1.1 Ensembles de données

3.1.1.1 iSEG

iSEG (Wang *et al.*, 2019) est un ensemble de données composé de 10 images à résonance magnétique T1 et T2 d'enfants âgés entre 6 et 8 mois. À cet âge, l'enfant est dans une phase dite *isointence*, une phase durant laquelle le contraste des tissus du cerveau est réduit. Cela se traduit par des plages d'intensités de la matière blanche et de la matière grise qui se chevauchent grandement, augmentant la difficulté des réseaux de segmentation à bien classifier les voxels appartenant à ces classes. Ce contraste réduit est le résultat d'une concentration plus élevée d'eau dans les structures du cerveau et la présence de matière blanche non myélinisée (Xue, Srinivasan, Jiang, Rutherford, Edwards, Rueckert & Hajnal, 2007). Les images sont également plus bruitées en raison du temps d'imagerie réduit afin d'éviter les artefacts de mouvement. Finalement, un autre défi que posent ces images réside dans l'effet de volume partiel (*partial volume effect*) en raison du contraste réduit. Les images sont échantillonnées dans une résolution de 1.0 mm^3. La vérité terrain inclut les masques de segmentation de la matière blanche, la matière grise et les liquides cérébraux spinaux.

3.1.1.2 MRBrainS

MRBrains (Mendrik, Vincken, Kuijf, Breeuwer, Bouvy, de Bresser, Alansary, de Bruijne, Carass, El-Baz, Jog, Katyal, Khan, Lijn, Mahmood, Mukherjee, Opbroek, Paneri, Pereira & Viergever, 2015) contient les images de 5 cerveaux adultes pondérées en T1 et T2 FLAIR (*Fluid Attenuated Inversion Recovery*). Les images ont été acquises sur un imageur 3 Tesla et échantillonnées à une taille de voxel de 0.958 mm ×0.958 mm ×3.0 mm. Cet ensemble de données a les mêmes classes à titre de vérité terrain que l'ensemble iSEG.

La figure 3.1 montre les histogrammes d'intensités des différents tissus composant le cerveau pour les ensembles de données iSEG (gauche) comportant des sujets d'enfants de 6-8 mois et MRBrainS (droite) comportant des sujets adultes. Nous pouvons constater le chevauchement dans les images iSEG de la matière blanche et grise causant une performance moindre des algorithmes de classification. On peut également noter l'étendue très différente des intensités d'un ensemble de données à l'autre.

3.1.1.3 ABIDE

L'ensemble *Autism Brain Imaging Data Exchange*, ou ABIDE (Di Martino et al., Yan, Li, Denio, Castellanos, Alaerts, Anderson, Assaf, Bookheimer, Dapretto, Deen, Delmonte, Dinstein, Ertl-Wagner, Fair, Gallagher, Kennedy, Keown, Keysers, Lainhart, Lord, Luna, Menon, Minshew, Monk, Mueller, Müller, Nebel, Nigg, O'Hearn, Pelphrey, Peltier, Rudie, Sunaert, Thioux, Tyszka, Uddin, Verhoeven, Wenderoth, Wiggins, Mostofsky & Milham, 2014), fut également utilisé afin de valider une fois de plus notre méthode. Cet ensemble est composé de 1112 images acquises sur 17 sites différents. Cet ensemble permet d'avoir une bonne variété d'images acquises avec des imageurs différents. Puisque ABIDE est une initiative multisite, les paramètres d'acquisition de chaque site sont disponibles sur le site web d'ABIDE [1]. Cet ensemble de données introduit également une variabilité au niveau de l'âge des sujets, celle-ci s'entendant de 7 à 64 ans. Les détails démographiques sont présentés dans la table 3.1. Il est à noter que 9 sujets ont été retirés

[1] http://fcon_1000.projects.nitrc.org/indi/abide/

de l'ensemble de données en raison d'une trop faible qualité de l'image. Puisque cet ensemble ne contient pas de masques de segmentation à titre de vérité terrain, nous avons considéré les masques de segmentation produits par le pipeline FreeSurfer *recon-all* [2].

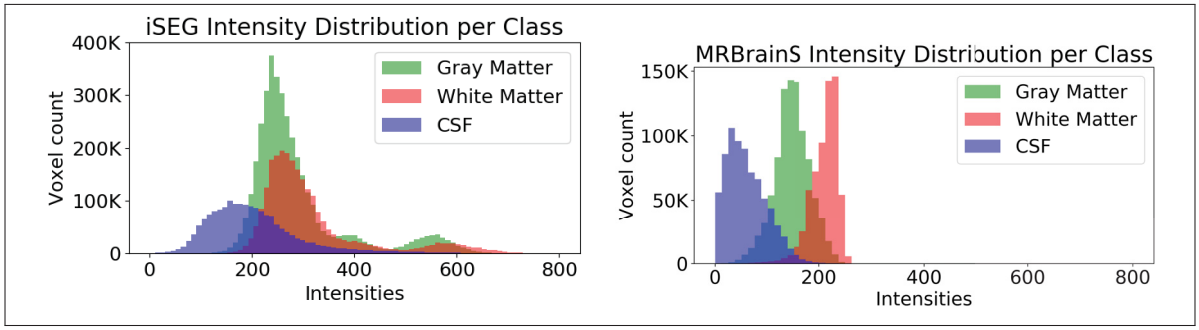

Figure 3.1 Histogrammes d'intensités des différents tissus composant le cerveau en fonction de l'ensemble de données

Tableau 3.1 Démographie des sujets présents dans l'ensemble de données ABIDE. Tiré de Di Martino et al. *et al.* (2014)

Group	**n**	**Homme**	**Femme**	**Âge (moyenne ± écart-type)**
Control	539	474	65	17.01 ± 8.36
ASD	573	474	99	17.08 ± 7.72

3.2 Protocole d'expérimentations

3.2.1 Prétraitement

Pour tous les ensembles de données, les images ont d'abord été recadrées aux dimensions du cerveau de l'image afin de minimiser la présence de l'arrière-plan dans l'image. Les images ont ensuite été redimensionnées à une taille normalisée qui les rend divisibles par la taille du segment (*patch*) de taille $1 \times 32 \times 32 \times 32$ et par le pas (*stride*) du segment de taille $1 \times 4 \times 4 \times 4$. Le retrait du crâne dans les images de MRBrainS fut effectué à l'aide de la carte de segmentation fournie avec l'ensemble de données. Les images ont également été rééchantillonnées afin d'avoir

[2] https://surfer.nmr.mgh.harvard.edu/fswiki/recon-all

un voxel de taille 1 mm^3 afin d'avoir un voxel de même taille que les ensembles de données iSEG et ABIDE. 40 000 segments d'images ont été choisis pseudo-aléatoirement au début de chaque processus d'entrainement pour chacun des ensembles de données. Les mêmes segments furent choisis à l'aide d'un *seed* prédéfini permettant une constance dans le choix des segments et permettant ainsi la répétabilité des expérimentations. 12 000 segments furent choisis de la même manière pour constituer les ensembles de validation et de test. Tous ces segments sont centrés sur un voxel appartenant à une classe autre que l'arrière-plan. La reconstruction des images est effectuée à chaque 10 époques. Les métriques Dice et la distance de Hausdorff moyenne (MHD) est calculée à partir de ces images reconstruites, normalisée et segmentée et sa carte de segmentation vérité terrain correspondante.

3.2.2 Détails d'implémentation

Pour iSEG, huit sujets ont été sélectionnés de manière pseudo-aléatoire pour l'entraînement, alors qu'une image a été consacrée à la validation. Le dernier sujet de l'ensemble de données fut utilisé à titre de test. Quant à MRBrainS, trois images furent sélectionnées de manière pseudo-aléatoire pour l'entraînement, alors que les deux autres ont été utilisées respectivement dans l'ensemble de validation et de test. Pour ABIDE, qui a beaucoup plus d'images, les images ont été séparées selon le modèle 60-20-20 couramment utilisé en apprentissage automatique. 60% des images ont été sélectionnées à des fins d'entraînement, alors que 20% furent réservées à la validation. Le dernier 20% fut conservé à des fins de test. Cinq expériences en parallèle avec des *seed* différents ont été réalisées afin de simuler une validation croisée.

Lors des expérimentations, le temps maximal d'entraînement fut limité à 5 jours sur un GPU NVIDIA Tesla V100 ayant 32 giga-octets de mémoire. Ce temps limité a donné lieu à un entraînement de 120 époques lorsque les ensembles de données de iSEG et MRBrainS furent utilisés et 70 époques lors d'un entraînement à trois ensembles de données. Une recherche approfondie afin de trouver le meilleur rapport entre la fonction de coût du réseau de segmentation et celle du discriminateur a été effectuée. La meilleure valeur donnant des résultats de génération d'image à la fois réaliste et donnant une performance de segmentation correcte est $\lambda = 1.5$.

Cette valeur recalibre les valeurs résultantes du calcul des fonctions de coût du Dice et de l'entropie croisée pour que ces deux valeurs soient dans une plage similaire. Une décroissance des poids (*weight decay*) de $0,001$ a été utilisé pour toutes les expérimentations. Un taux d'apprentissage de départ de $0,001$ pour le générateur et le réseau de segmentation fut utilisé, alors qu'un taux d'apprentissage de 0.0001 fut utilisé pour le discriminateur. Chaque modèle fut entrainé avec l'algorithme d'optimisation Adam. Les modèles furent implémentés avec la libraire d'apprentissage profond PyTorch [3].

3.2.3 Évaluation

Nous évaluons la performance de segmentation de notre méthode en utilisant le coefficient Dice moyen (DSC) qui mesure le degré de chevauchement entre la segmentation prédite $\mathbf{S}$ et la vérité terrain $\mathbf{G}$:

$$\mathrm{DSC}(\mathbf{S},\mathbf{G}) \;=\; \frac{2\,|\mathbf{S}\cap\mathbf{G}|}{|\mathbf{S}+\mathbf{G}|}. \tag{3.1}$$

Nous considérons également la distance moyenne de Hausdorff (MHD) afin de mesurer la qualité de la segmentation sur les frontières de celle-ci :

$$\mathrm{MHD}(\mathbf{S},\mathbf{G}) \;=\; \frac{1}{2}\big(\mathrm{dist}(\mathbf{S},\mathbf{G}) \;+\; \mathrm{dist}(\mathbf{G},\mathbf{S})\big). \tag{3.2}$$

Ici, $\mathrm{dist}(\cdot,\cdot)$ est la distance euclidienne maximale entre un point de la segmentation prédite et son point le plus proche dans la vérité terrain (ou vice-versa).

[3] http://pytorch.org

3.3 Résultats

3.3.1 Performance de base sur les ensembles de données

Nous établissons premièrement un résultat de base en mesurant la performance d'un modèle de segmentation évalué sur plusieurs ensembles de données (*cross-dataset baseline*). Un modèle adoptant l'architecture 3D U-Net fut entraîné sur des images T1 non normalisées. Cette architecture prend en entrée un segment d'images 3D et produit la carte de segmentation de ce segment. La fonction de coût Dice est calculée entre cette prédiction et la carte de segmentation représentant la vérité terrain de ce segment d'image. Cette fonction de coût définit la tâche du réseau, soit d'optimiser les poids de ce dernier de manière à obtenir la plus haute valeur Dice possible en minimisant l'erreur de chevauchement des classes. Quatre scénarios ont été testés :

1. Entrainement et test sur l'ensemble iSEG seulement ;
2. Entrainement et test sur l'ensemble MRBrainS seulement ;
3. Entraînement sur l'ensemble iSEG, test sur l'ensemble MRBrainS ;
4. Entraînement sur l'ensemble MRBrainS, test sur l'ensemble iSEG.

Les deux derniers scénarios évaluent la capacité du modèle à généraliser à travers des ensembles de données aux caractéristiques bien différentes, soit la distribution d'intensités, la démographie et les protocoles d'acquisition.

Le tableau 3.2 illustre les performances obtenues. Lorsqu'on entraîne et teste le réseau de segmentation avec le même ensemble de données, on remarque que la performance est relativement bonne. Cependant, on note que lorsqu'on effectue un test croisé, soit d'entraîner un réseau de segmentation avec un ensemble de données et de tester sur un second ensemble, le réseau de neurones pleinement convolutif peine à classifier correctement les voxels, accusant une baisse de performance pouvant aller jusqu'à 62.3%. La performance de segmentation est médiocre et le réseau est inutilisable sur les données de test. Cela est dû à la non-adaptation et la grande différence entre les distributions des données d'entraînement et de test. Le meilleur indice Dice moyen atteint en entraînant sur l'ensemble de données iSEG et en testant sur l'ensemble

MRBrainS fut de 42,5%, alors qu'un indice moyen de 31,3% a été obtenu en inversant les deux ensembles de données. Ces résultats montrent la très grande sensibilité des modèles d'apprentissage profond en fonction de leurs données d'entraînement, validant une fois de plus l'importance d'une fonction de normalisation spécifique à la tâche que l'on souhaite accomplir. Visuellement, la figure 3.2 illustre la segmentation d'un cerveau sur l'ensemble de données iSEG. À gauche, la vérité terrain d'une segmentation d'un cerveau d'enfant, au centre, la segmentation effectuée par un réseau de segmentation 3D U-Net. Lorsque testé sur la même distribution utilisée lors de l'entraînement, on obtient relativement une bonne segmentation. Finalement, à droite se trouve le résultat d'une segmentation lorsqu'on utilise une distribution différente lors de la phase d'entraînement. La segmentation est inutilisable.

Tableau 3.2 Résultats de base avec un réseau de segmentation

Ensemble d'entraînement	**Ensemble de test**	**Dice**			
		CSF	**GM**	**WM**	**Moyen**
iSEG	iSEG	0.920	0.857	0.828	0.868
MRBrainS	MRBrainS	0.861	0.789	0.839	0.830
iSEG	MRBrainS	0.401	0.354	0.519	0.425
MRBrainS	iSEG	0.293	0.082	0.563	0.313

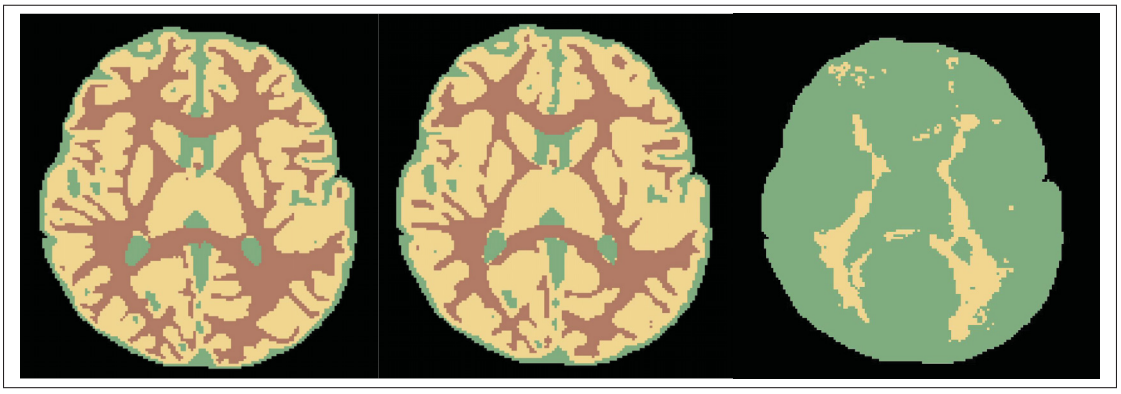

Figure 3.2 Segmentation d'un cerveau sur l'ensemble de données iSEG en fonction de l'ensemble de données d'entraînement

3.3.2 Évaluation sur deux domaines

Nous comparons ensuite notre méthode à une technique de normalisation communément utilisée, soit la standardisation (Birenbaum & Greenspan, 2016) et l'approche de fonction apprise de normalisation de Drozdzal *et al.* (2018). Ces méthodes ont été utilisées sur les ensembles de données iSEG et MRBrainS. Comme mentionné précédemment, ces deux ensembles comportent des caractéristiques différentes. iSEG contient des images T1 d'enfants dans leur phase isointense alors que MRBrainS contient des images T1 de sujets adultes. Par conséquent, une méthode de normalisation s'effectuant sur une base *par image* a une efficacité réduite. La technique de standardisation testée dans cette expérience normalise l'intensité de chaque voxel dans un volume en soustrayant au voxel la moyenne du volume et en le divisant par l'écart-type du volume. Le résultat de cette méthode est inscrit comme étant *standardisé* dans le tableau des résultats. La méthode de normalisation apprise, que nous appelons *préprocesseur*, contient le même pipeline que notre méthode (un générateur et un réseau de segmentation), mais omet l'addition du discriminateur. Ces résultats de base permettent de mieux apprécier le gain de notre méthode antagoniste. Puisque le *préprocesseur* n'applique aucune contrainte sur le réalisme et favorise l'ajustement des intensités des voxels en fonction de la performance de segmentation, il est normal de s'attendre à une meilleure coefficient de recouvrement au détriment du niveau de réalisme de l'image intermédiaire produite. En utilisant ces deux méthodes, la fonction de coût de l'indice Dice est propagée de la sortie du réseau de segmentation jusqu'au préprocesseur, produisant ainsi une image intermédiaire optimisée strictement pour la tâche de segmentation.

3.3.2.1 Performance de segmentation

L'indice Dice obtenu pour la méthode *standardisée*, la méthode *préprocesseur* sans contrainte de réalisme et notre méthode est répertorié dans le tableau 3.3 (paramètre double-site). Comme nous l'avions prévu, l'ajout d'une méthode de standardisation permet une augmentation de la performance de segmentation. La méthode *préprocesseur* ainsi que notre méthode se démarque avec un indice Dice moyen plus haut que la méthode classique. L'histogramme d'une image produite par le *préprocesseur* montre des intensités qui suivent une courbe normale pratiquement

parfaite. Toutefois, en raison de ce changement trop drastique des intensités, cette même image n'est pas utilisable dans un cadre clinique. Elle est très difficilement interprétable pour un professionnel de la santé. Le résultat sur l'indice Dice montre une augmentation de la performance lors de la segmentation autant sur deux que sur trois ensembles de données, montrant ainsi un potentiel de la méthode lors d'un apprentissage avec de multiples ensembles de données. Cela est rendu possible en ajustant sévèrement les intensités des classes afin d'optimiser la tâche de segmentation. Il en résulte cependant une image distorsionnée et illisible.

Notre réseau discriminateur tente de classifier l'ensemble de données sources du segment d'image qui lui est présenté. Notre méthode s'appuie sur la tâche de segmentation afin d'effectuer la normalisation en ligne. Puisque la fonction de coût de l'indice Dice est utilisée et que cette valeur est propagée jusqu'au réseau générateur, les éléments structurels sont conservés lors de la génération de l'image. Les poids du générateur sont optimisés de manière à augmenter l'indice Dice, ce qui nécessite la génération d'une structure précise. Le discriminateur est quant à lui entrainé avec la fonction d'entropie croisée. Cette fonction permet de focaliser sur les caractéristiques globales des images. En définissant la tâche du discriminateur comme étant la classification du domaine d'imagerie, on tente ainsi d'optimiser les poids de façon à capter les différences entre les domaines. Cependant, le jeu *minimax* qu'offre l'architecture GAN permet d'appliquer au générateur l'inverse de cette fonction de coût. Ainsi, le générateur produit des images dont la distribution des intensités permet de duper le discriminateur. Lorsque la distribution générée ne peut être classifiée correctement par le discriminateur (moment ou la précision de la prédiction est à son plus bas) et que l'évolution de la fonction d'entropie croisée montre une erreur maximale, le système a convergé. L'avantage de cette méthode réside dans la possibilité d'entraîner le système avec des images issues de domaines très différents, en l'occurrence des cerveaux d'adultes et des cerveaux d'enfants tout en conservant des performances de segmentation respectable compte tenu de la difficulté du problème.

Notre méthode n'accuse qu'un déficit marginal 0.03% par rapport à la méthode de Drozdzal *et al.* (2018). Cela montre qu'imposer une contrainte de réalisme tout en considérant la tâche de segmentation n'impacte pas vraiment les performances de cette tâche. Nous avons également

remarqué que notre architecture, lorsqu'elle est entraînée avec les ensembles de données des deux domaines d'imagerie, obtient de meilleures performances de segmentation que notre résultat de base qui consistait en l'entraînement et le test sur des ensembles de données de manière indépendante (tableau 3.2). Effectivement, notre méthode obtient un résultat moyen supérieur de 54,2% par rapport à cette expérience. Notre méthode est capable de normaliser l'image d'entrée tout en conservant l'image intermédiaire réaliste. La performance est légèrement meilleure lorsqu'on prend en considération les deux modalités disponibles des ensembles de données, atteignant un indice Dice moyen de 88,7%. Des exemples de segmentation sur des images de test iSEG et MRBrainS sont montrés à la figure 3.4. Nos résultats confirment également que le gain de notre méthode sur l'indice Dice moyen est significatif avec une valeur P de 0.036 par rapport à l'expérience comprenant la normalisation Min-Max avec un réseau de segmentation U-Net.

Tableau 3.3 Indice Dice en fonction de l'architecture du modèle et des données

Paramètre	**Méthode**	**CSF**		**GM**		**WM**		**Moyen**	
		DSC	**MHD**	**DSC**	**MHD**	**DSC**	**MHD**	**DSC**	**MHD**
Dual-site (iSEG + MRBrainS)	Min-max Scaling	0.707	1.13	0.833	0.546	0.900	0.275	0.813	0.650
	Domain Standardization	0.836	**0.498**	0.790	0.734	0.897	0.792	0.841	0.675
	Drozdzal *et al.* (2018)	**0.860**	0.517	0.831	0.702	**0.919**	**0.227**	**0.870**	0.482
	Notre méthode	0.848	0.517	**0.836**	**0.627**	0.902	0.284	0.862	**0.476**
Multi-site (iSEG + MRBrainS + ABIDE)	Min-max Scaling	0.865	0.439	0.827	0.690	0.828	3.09	0.840	1.41
	Domain Standardization	0.881	0.684	0.856	0.812	0.860	0.264	0.866	0.587
	Drozdzal *et al.* (2018)	**0.895**	**0.392**	0.870	**0.530**	**0.922**	**0.251**	**0.896**	**0.390**
	Notre méthode	0.887	0.422	**0.870**	0.598	0.913	0.293	0.890	0.438

3.3.2.2 Performance de normalisation

L'avantage de notre méthode est présenté à la figure 3.3. On peut y voir des segments d'images sélectionnés de manière aléatoire qui correspondent aux images d'entrées et générées (normalisés). Dans cette figure, l'entrée est normalisée avec la technique Min-Max alors qu'à droite se trouve le résultat de notre méthode de normalisation. Les segments normalisés présentent une homogénéité accrue, une distribution des intensités plus uniforme tout en préservant le réalisme et les détails de l'image. Les contrastes des images générées sont augmentés, facilitant ainsi la lecture de l'image. Cela est rendu possible grâce à notre approche dirigée par une tâche (*task-driven*) qui minimise la fonction de coût de l'indice Dice, ce qui augmente les contrastes aux abords des frontières. Ce

résultat est rendu possible par l'ajout du discriminateur qui permet d'apprendre une fonction de transfert en rejetant les fonctions de normalisation irréalistes. Ainsi, les histogrammes des images intermédiaires montrent des courbes ayant une forme plus gaussienne pour chaque classe. Les intensités des voxels d'une même classe sont également réalignées autour d'une plage de valeurs communes. Cela est une conséquence directe de l'approche dirigée par une tâche puisqu'il est plus facile de classifier des intervalles d'intensités communes à chaque classe. Les segments d'images en entrée ont une distribution plus large des intensités avec des modes plus distincts, alors que les images générées ont une distribution plus étroite, attestant une variance intraclasse réduite. Cette variance réduite permet à son tour une classification plus précise des voxels. Ce résultat montre également que l'exploitation de l'information conjointe et normalisée de deux ensembles de données permet une meilleure performance globale de l'algorithme de segmentation.

3.3.3 Évaluation multi-site

Puisque le but recherché est d'apprendre une représentation commune au sein de plusieurs domaines d'imagerie, un troisième domaine a été ajouté au problème. L'ensemble de données ABIDE est constitué d'images provenant de 17 sites différents, tous ayant des protocoles d'acquisition différents. Des segments d'images sont sélectionnés au hasard parmi les 1 103 sujets qui le composent. Dans notre cas, ABIDE est perçu toutefois comme un seul domaine. En effet, les images que nous avions à notre disposition sont les images déjà normalisées avec les statistiques des 17 sites mis en commun.

La performance de segmentation des méthodes de standardisation classique, *préprocesseur* de Drozdzal *et al.* (2018) et notre méthode entraînée avec les trois ensembles de données est montrée au tableau 3.3. Une fois de plus, la méthode de Drozdzal *et al.* (2018) et notre méthode atteignent des résultats comparables et témoignent d'une importante augmentation de la performance de segmentation par rapport à la méthode de standardisation classique. Notre méthode proposée atteint avec ce troisième domaine un indice Dice moyen de 89,0%, soit 2,40% de plus que l'expérience avec deux domaines d'imagerie. Ce score plus élevé révèle l'avantage d'avoir plus

de données et plus de variété dans les distributions d'intensités. Les images restent tout de même proches de leur représentation visuelle originale.

La divergence de Jensen-Shannon (JSD) mesure la divergence moyenne de Kullback-Leibler entre une distribution (par exemple un histogramme) et la moyenne des distributions. Le tableau 3.4 montre la valeur de la JSD entre les images originales en entrées et celles normalisées par le générateur. Une valeur moindre signifie que les distributions sont plus similaires entre les segments présentés en entrée au générateur et les segments générés par celui-ci. On peut mieux apprécier l'effet de la normalisation sur l'histogramme des intensités. La valeur JSD moindre se traduit dans la figure par une distribution plus étroite centrée sur un mode, réduisant ainsi la variance intraclasse des intensités. Cette variance réduite se traduit par une performance supérieure de la tâche de segmentation. Cette valeur moindre suggère également une distribution plus uniforme entre les différents domaines d'imagerie. Toutefois, cette métrique ne mesure pas le réalisme de notre image, ce point étant le principal avantage de notre méthode. La divergence des distributions de notre méthode est égale à celle utilisée dans Drozdzal *et al.* (2018), mais conserve le réalisme des images.

Tableau 3.4 La distance de Jensen-Shannon des données brutes et normalisées par le générateur en fonction des ensembles de données

Ensembles de données	**Données brutes**	**Préprocesseur**	**Notre méthode**
iSEG + MRBrainS	2.0839	0.2793	0.2788
iSEG + MRBrainS + ABIDE	12.2212	0.4184	0.4180

3.3.4 Évaluation multimodale

Des approches récentes ont montré qu'utiliser plus d'une modalité peut accroître la précision de la segmentation produite par un algorithme d'apprentissage profond (Dolz *et al.*, 2019). Par exemple, des images T1 produisent des contrastes plus prononcés entre la matière blanche et grise, alors que des images pondérées en T2 offrent un meilleur contraste entre les liquides cérébraux spinaux et les tissus qui composent le cerveau. Il est donc plausible de croire que la combinaison des images T1 et T2 permet d'améliorer la précision de la segmentation. Notre

méthode supporte les entrées multimodales. Nous évaluons l'amélioration des résultats de notre algorithme antagoniste avec des entrées multimodales dans le tableau 3.5. Un indice Dice plus grand signifie une meilleure performance de la segmentation. Une distance moyenne de Hausdorff plus petite correspond à une distance moyenne plus petite entre la segmentation et la vérité-terrain, ce qui se traduit en une meilleure performance. La précision de la segmentation est améliorée avec un indice Dice moyen supérieur de 0.020 et une distance moyenne de Hausdorff diminuée de 0.081 mm.

Tableau 3.5 Indice Dice et la distance de Hausdorff en fonction des types de tissus du cerveau et des modalités d'imagerie utilisées

	CSF		**GM**		**WM**		**Moyen**	
Modality	**DSC**	**MHD**	**DSC**	**MHD**	**DSC**	**MHD**	**DSC**	**MHD**
T1 only	0.848	0.517	0.836	0.627	0.902	0.284	0.862	0.476
T1 + T2	**0.889**	**0.431**	**0.862**	**0.493**	**0.910**	**0.165**	**0.887**	**0.363**

3.3.5 Robustesse à la dégradation de l'image

Notre méthode dirigée par la tâche permet également d'améliorer la qualité d'une image dégradée avec des intensités non homogènes. Afin d'évaluer cette capacité de notre algorithme, nous avons entraîné ce dernier avec 50% d'images augmentées avec un effet de champ (*bias field*). Puisque le discriminateur discerne la sortie du générateur de ces images dégradée des images réelles, il encourage le générateur à produire des images dépourvues de cet effet de champ tout en continuant de préserver le réalisme. Le générateur apprendra des cartes de caractéristiques (*feature maps*) utiles à l'élimination du champ de polarisation suite au retour de l'information du discriminateur.

Nous mesurons l'habileté de notre méthode à corriger cette dégradation en calculant le coefficient de corrélation Pearson entre les intensités et la position des voxels sur l'axe de l'effet de champ (ici l'axe des y). Puisque l'effet de champ modélisé est linéaire, une corrélation plus grande correspond à une dégradation plus sévère de l'image. Le tableau 3.6 donne la corrélation moyenne des images de test pour les ensembles de données iSEG et MRBrainS en fonction d'un

effet de champ croissant $\alpha \in 0,3,0,5,0,7,0,9$. L'intensité moyenne en fonction de la position sur l'axe des y pour une valeur $\alpha = 0,5$ et $\alpha = 0,9$ est montrée à la figure 3.6. Nous pouvons observer que, pour les deux ensembles de données, l'intensité est notamment moins corrélée à sa position sur l'axe des y dans une image normalisée, illustrant la capacité de notre méthode à corriger ce type de dégradation typique d'une image médicale. Notre méthode montre son plein potentiel avec des valeurs plus grandes de α telles que $\alpha = 0,7$ et $\alpha = 0,9$. La figure 3.7 montre une image de test avec un effet de champ $\alpha = 0,5$ et la normalisation correspondante donnée par le générateur. On peut constater sur cette dernière image des intensités plus uniformes, une meilleure homogénéité et un effet de champ réduit.

Un bruit Ricien fut également ajouté à l'image originale de l'ensemble de données. Différents rapports signal-bruit (RSB) ont été choisis afin d'altérer la qualité de l'image. Afin de mesurer le bénéfice de notre méthode à enlever ce bruit, l'erreur moyenne au carré (MSE) fut utilisée. Une erreur plus basse signifie une image plus proche de l'originale non dégradée et ayant moins de bruit. Il en résulte une image d'une meilleure qualité. Dans le tableau 3.7, on peut constater que notre méthode est performante lorsqu'on teste sur des images ayant un rapport signal-bruit plus élevé. Toutefois, lorsque peu de bruit est présent dans l'image, l'interpolation par voisin le plus près du réseau générateur U-Net montre sa limite. En effet, ce type d'interpolation peu sophistiquer a tendance à répliquer le bruit présent dans les cartes de caractéristiques et le sur-échantillonne. Il aurait été intéressant de comparer avec une portion décodeur du U-Net pourvue de convolutions transposées à la place d'opération de sur-échantillonnage. Cette alternative, bien que toutefois beaucoup plus exigente en puissance de calcul, aurait directement adressé ce problème.

3.3.6 Impact de l'hyperparamètre Lambda

Tel que défini à l'équation 2.3, l'hyperparamètre λ a un impact direct sur le niveau de réalisme de l'image. Une valeur plus basse accentue la précision de la segmentation alors qu'une valeur plus haute priorise la génération d'images plus réalistes et invariant au domaine. Dans cette expérience, nous analysons l'impact de cet hyperparamètre important de notre architecture.

Tableau 3.6 Coefficient de corrélation de Pearson (ρ) entre l'intensité d'un voxel et sa position dans l'axe des y avant et après normalisation dans une tranche axiale d'un sujet de test pour les ensembles de données MRBrainS et iSEG pour différentes valeurs d'intensités α

	MRBrainS		**iSEG**	
α	**Dégradée**	**Normalisée**	**Dégradée**	**Normalisée**
Original	−0.0199	–	0.0271	–
0.3	0.0982	0.0534	0.2560	0.1192
0.5	0.1963	0.1422	0.4264	0.2430
0.7	0.3103	0.2393	0.5914	0.4599
0.9	0.4368	0.3649	0.7316	0.5928

Tableau 3.7 Distance moyenne au carrée (MSE) entre une tranche axiale originale dégradée d'un sujet et une tranche axiale d'un sujet après normalisation pour différents rapports signal-bruit (RSB)

	MRBrainS		**iSEG**	
SNR (dB)	**Dégradée**	**Normalisée**	**Dégradée**	**Normalisée**
100	0.0004	0.0653	0.0003	0.0089
80	0.0045	0.0553	0.0029	0.0064
60	0.0154	0.0456	0.0099	0.0069
40	0.0380	0.0449	0.0251	0.0159
20	0.0808	0.0627	0.0549	0.0282

Le tableau 3.8 donne l'indice Dice moyen et l'exactitude du discriminateur pour les échantillons d'entraînement et de test avec $\lambda \in 0.1, 1.5, 5.0$. Un λ plus petit priorise la qualité de la segmentation, alors qu'une valeur plus grande insiste sur la génération d'images réalises et invariant au domaine. Comme nous nous y attendions, la qualité de la segmentation décroit avec une valeur plus élevée du paramètre puisque le modèle focalise sur la génération d'images réalistes et moins sur l'adaptation des intensités en faveur du réseau de segmentation. Ainsi, on observe une baisse de 2.4% dans l'indice Dice lorsqu'on passe de $\lambda = 0.1$ à $\lambda = 5.0$. Le réalisme des images générées peut-être évalué avec l'exactitude du réseau discriminateur. Une valeur plus haute indique que l'image générée peut être différenciée des vraies images plus facilement par le

discriminateur. Pour une valeur plus petite $\lambda = 0.1$, le discriminateur peut facilement classifier les images lors de l'entraînement, indiquant que les images produites sont très différentes des images issues des ensembles de données réels. Au fur et à mesure que λ augmente, les images générées deviennent de plus en plus similaires aux images réelles en conservant toutefois leur optimisation provenant de la fonction de coût du réseau de segmentation. Il en résulte un décroissement de l'exactitude du discriminateur. Finalement, comme montré dans les matrices de confusion de la figure 3.8, le discriminateur est incapable prédire correctement le domaine de l'image. Notez que dans le cas du résultat où $\lambda = 0.1$, la disparité entre la valeur de la précision entre l'entrainement et le test vient du fait que le réseau est en état de sur-apprentissage, un état pratiquement impossible à éviter pour une telle valeur du paramètre.

Tableau 3.8 Indice Dice moyen pour différentes valeurs de l'hyperparamètre λ

Lambda (λ)	**Indice Dice Moyen**	**Précision du discriminateur (%)**	
		Entraînement	**Test**
0.1	0.875	98.95	34.49
1.5	0.866	44.10	32.58
5.0	0.851	44.07	28.58

3.3.7 Impact de l'architecture du discriminateur

En addition du discriminateur basé sur l'architecture ResNet, d'autres réseaux discriminateurs ont été testés. L'architecture plus simple du discriminateur présent dans DCGAN (Radford *et al.*, 2016) a été testée. La fonction de coût a également été modifiée de façon à satisfaire la méthode LSGAN (Mao, Li, Xie, Lau, Wang & Smolley, 2017b) qui utilise la fonction objective des moindres carrés (*Least Squares*) au lieu de l'entropie croisée. Dans ce cas, le modèle ResNet-18 fut conservé tout en adoptant la fonction de coût des moindres carrés.

Sur la figure 3.9, on peut constater que le modèle adoptant la fonction de coût des moindres carrés est pourvu d'une plus grande stabilité lors de l'entraînement. Cela est dû notamment à la meilleure évaluation de la distance d'un échantillon par rapport à la frontière de décision que donne la fonction des moindres carrés. Malheureusement, dû au nombre de patients trop limité

(12 patients en entraînement), le nombre de profils d'intensités disponibles pour l'apprentissage de la fonction de classification du discriminateur est considérablement réduit. Ainsi, on voit le réseau ResNet en état de surapprentissage très rapidement, même lorsque la fonction des moindres carrés est utilisée. Le modèle qui souffre le moins du phénomène de surapprentissage est le classificateur présent dans DCGAN. Ce modèle contient moins de paramètres que la plus petite variante de ResNet, ce qui aide grandement à réduire le sur-apprentissage et améliore ainsi la capacité de généralisation du réseau.

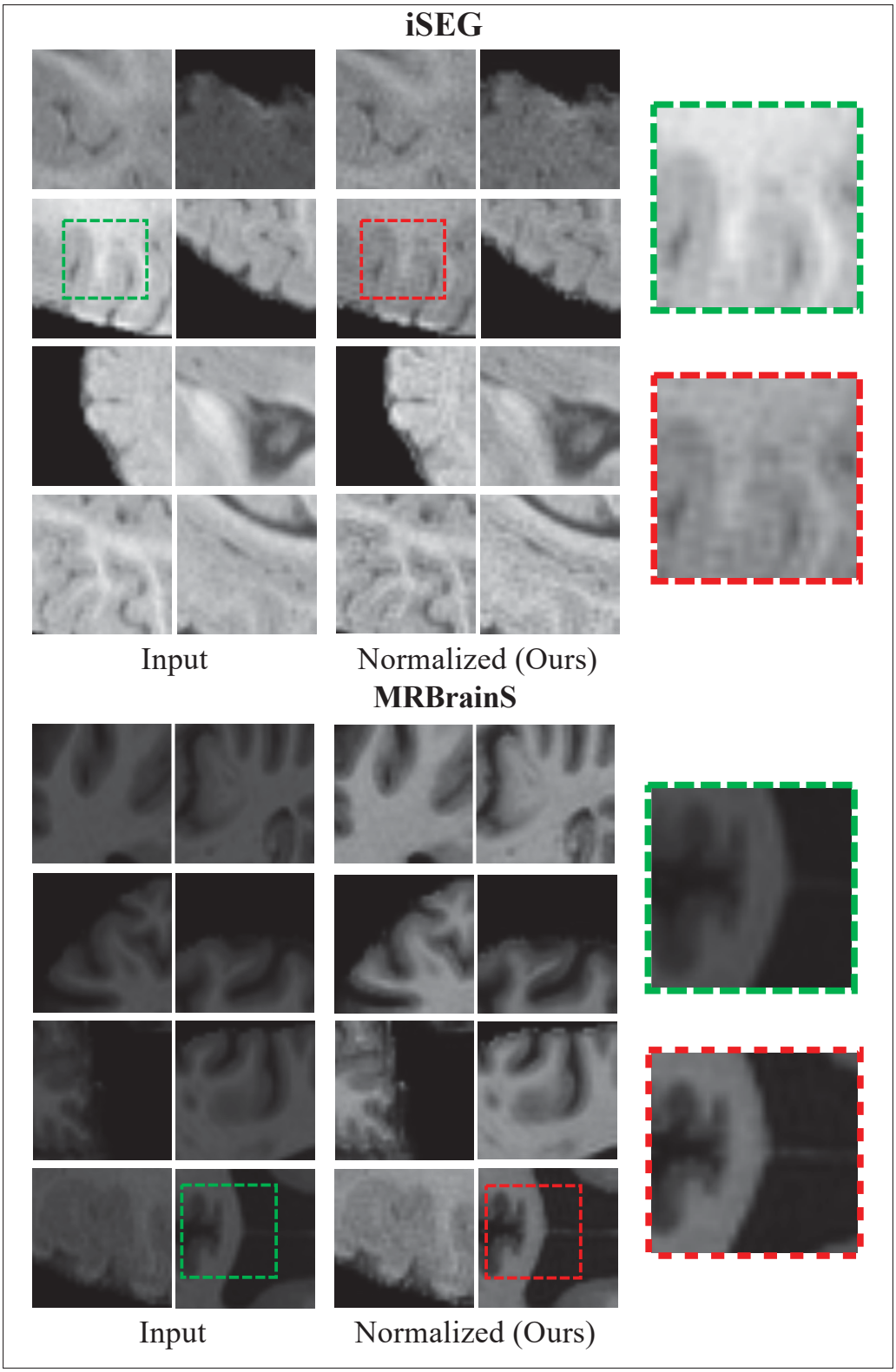

Figure 3.3 Résultats de normalisation avec notre méthode

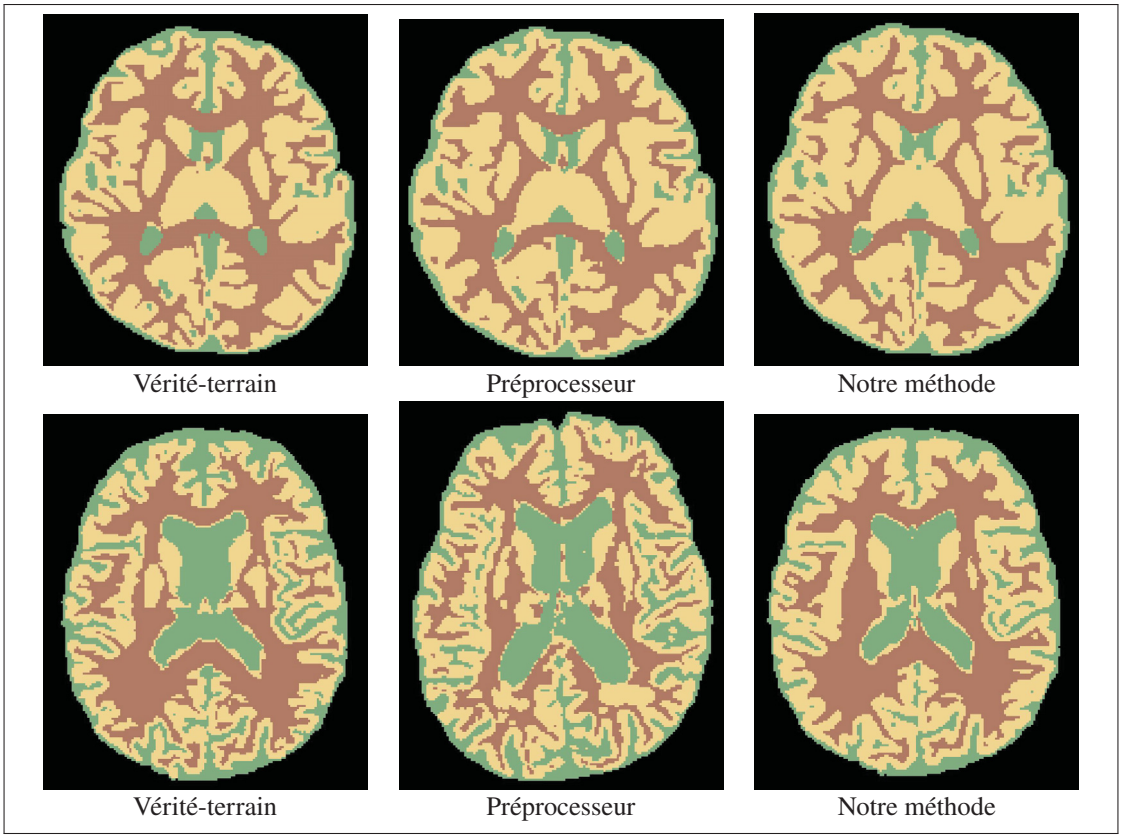

Figure 3.4 Visualisation du résultat de segmentation pour deux images de test. **Ligne du haut** : image iSEG. **Ligne du bas** : image MRBrainS.

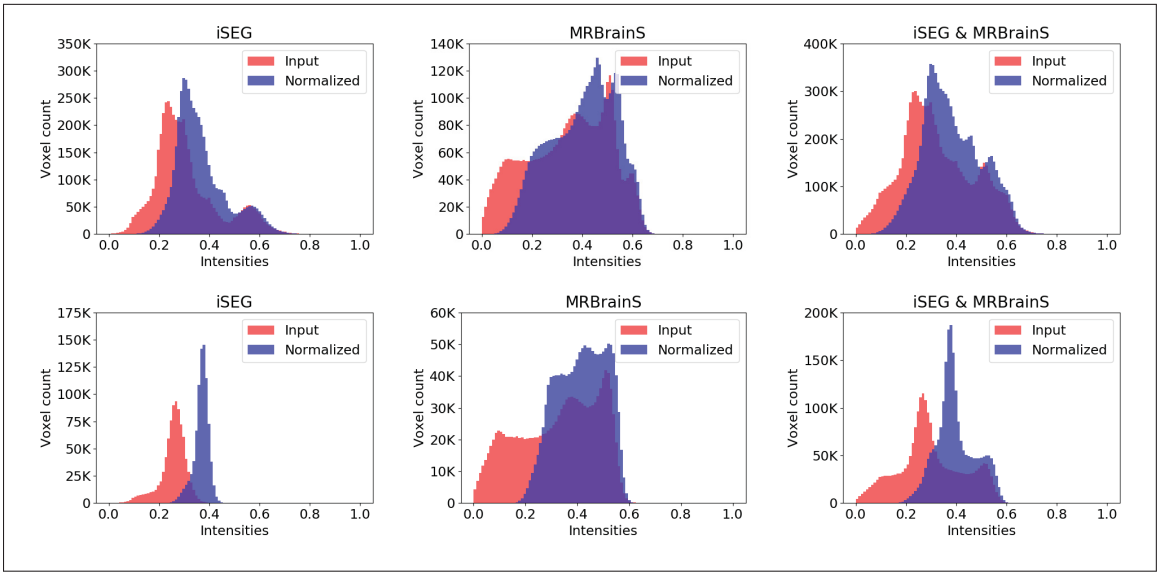

Figure 3.5 Histogrammes des images de tests avec contrainte de réalisme (notre méthode)

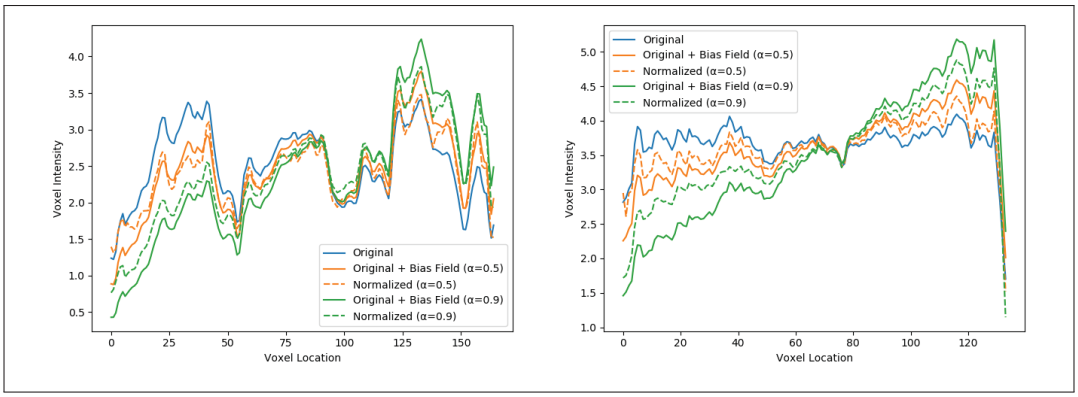

Figure 3.6 Intensité moyenne d'un voxel d'une trache axiale pour un sujet de l'ensemble de données MRBrainS (gauche) et iSEG (droite) pour un effet de champ $\alpha = 0.5$ et $\alpha = 0.9$

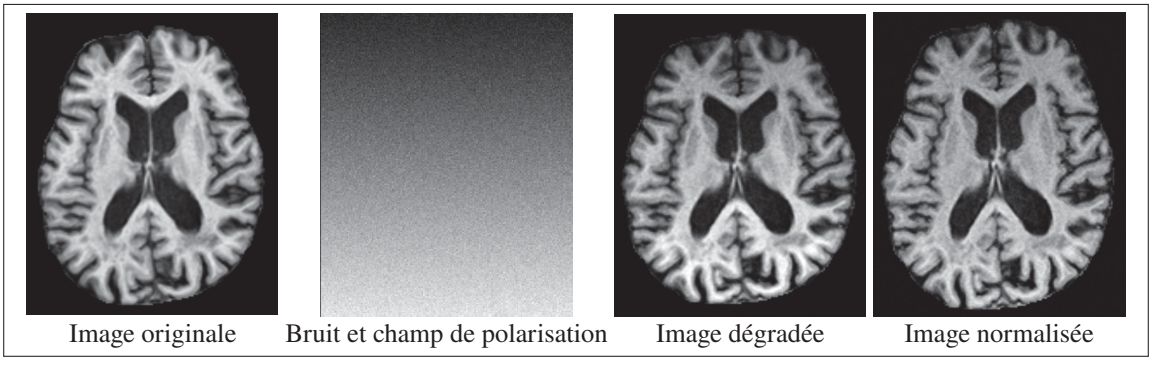

Figure 3.7 Résultat d'une prédiction du générateur avec une image de test dégradée

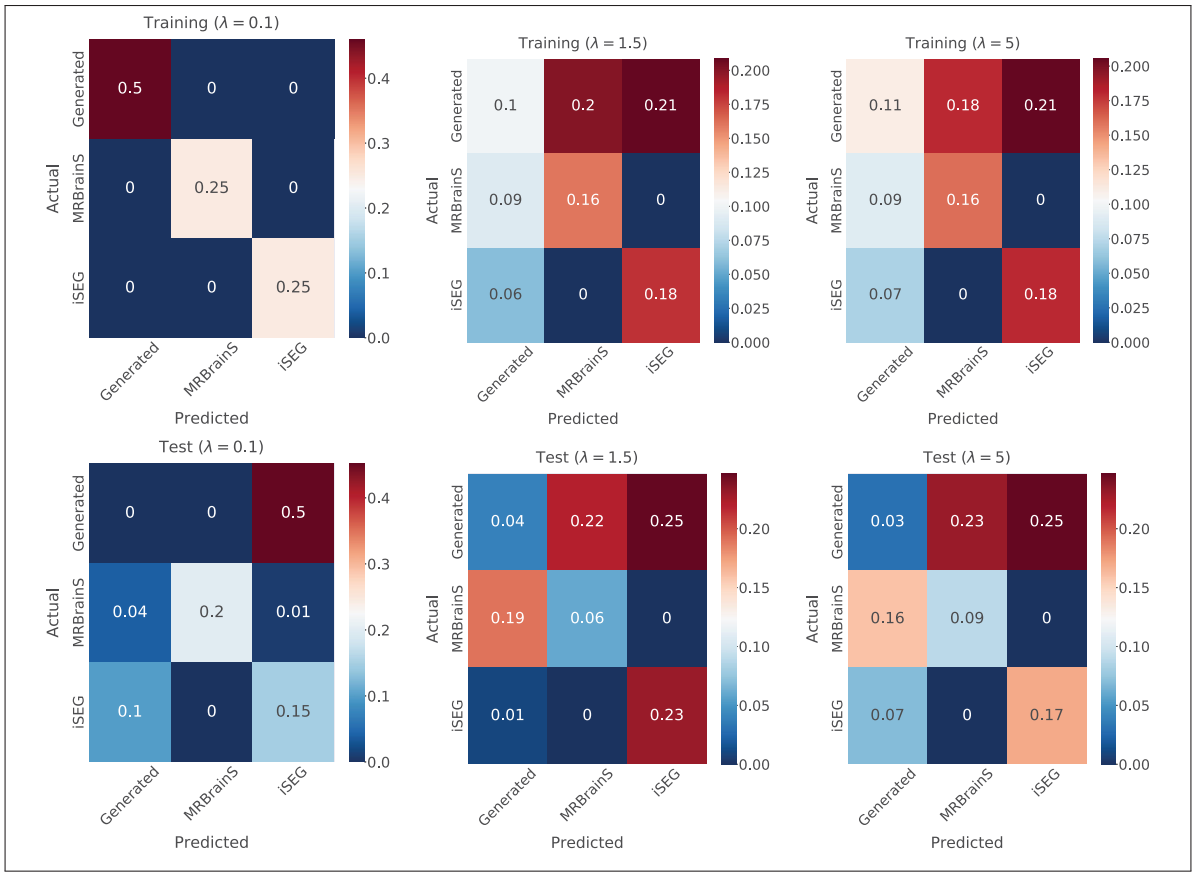

Figure 3.8 Matrices de confusions normalisées pour différentes valeurs de l'hyperparamètre λ

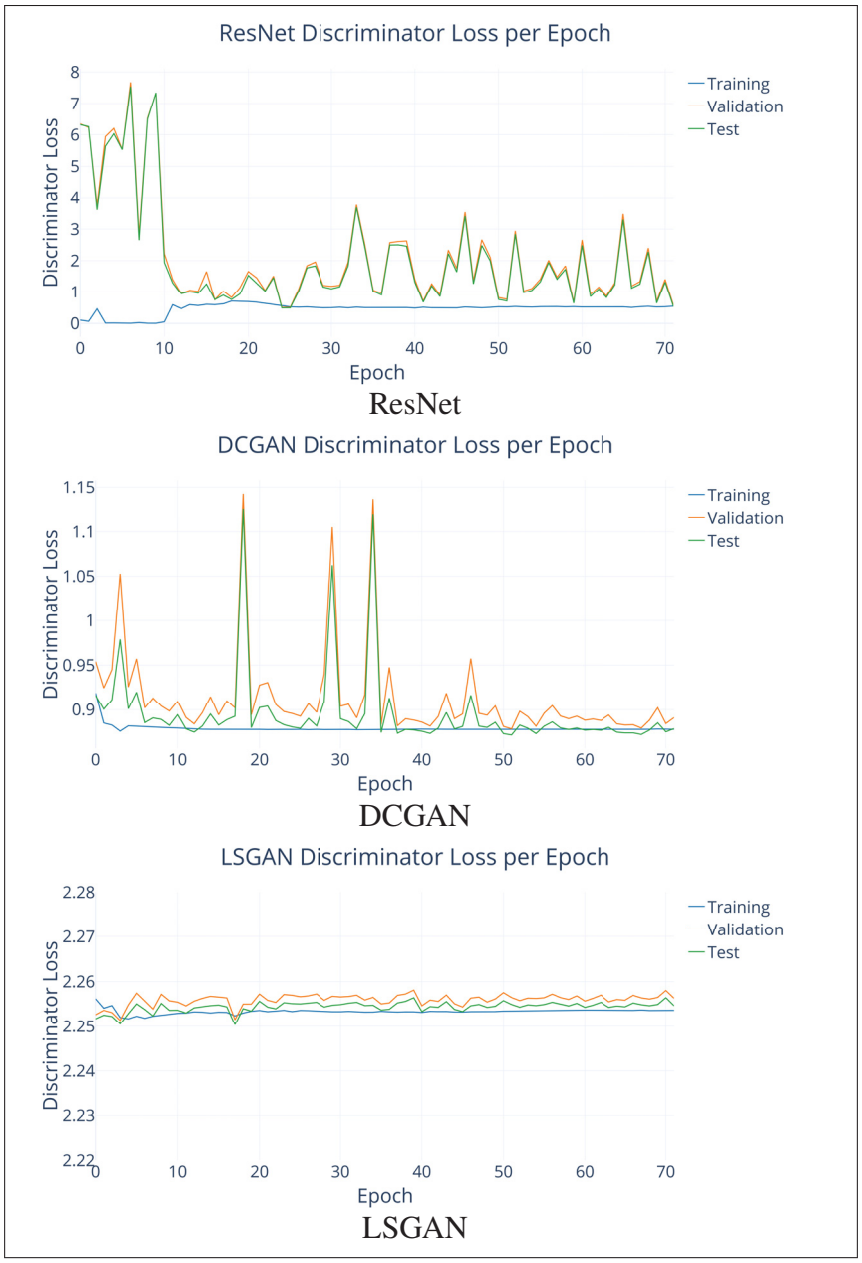

Figure 3.9 Visualisation de l'évolution de la valeur des fonctions de coûts en fonction de l'époque

CONCLUSION ET RECOMMANDATIONS

4.1 Résumé des contributions

Ce travail présente une méthode de normalisation dirigée par les données et une tâche de segmentation qui produit à la fois des images réalistes et optimisées qui améliorent la qualité de la segmentation. Notre méthode met à profit une stratégie d'apprentissage antagoniste qui implique trois réseaux de neurones dont les poids sont optimisés de manière simultanée. Un générateur normalise l'image en entrée tout en préservant le réalisme de l'image. Un réseau appliqué à une tâche, dans notre cas un réseau de segmentation, prend ces images normalisées et produit une carte de segmentation précise. Finalement, un discriminateur classifie le domaine de ces images. Contrairement aux approches antagonistes plus traditionnelles où le discriminateur distingue les images réelles des images générées, notre discriminateur est entraîné selon un problème à $(K + 1)$ classes avec K domaines d'imagerie réels correspondant aux ensembles de données et une classe supplémentaire correspondant aux images générées. En maximisant la fonction de coût du discriminateur tout en minimisant celle du réseau de segmentation, le générateur apprend à produire une image qui est à la fois harmonisée à travers les domaines d'imagerie, réaliste et optimisée pour la tâche de la segmentation. Comparativement à des techniques récentes d'harmonisation des données (Modanwal *et al.*, 2020), notre méthode comporte moins d'hyperparamètres à optimiser et peut facilement s'adapter à plusieurs domaines d'imagerie différents sans modification majeure. Notre méthode a également été publiée dans le cadre de la conférence *International Symposium on Biomedical Imaging - ISBI 2020* (Delisle *et al.*, 2020).

4.2 Résultats

Les avantages de notre méthode ont été démontrés dans un ensemble d'expérimentations comprenant trois ensembles de données d'imagerie du cerveau par résonance magnétique :

iSEG, MRBrainS et ABIDE. Dans une première expérience, nous établissons un standard de performance où nous constatons qu'un modèle entrainé sur un ensemble de données puis testé sur un second performe de façon médiocre sur ce dernier (tableau 3.2). Notre méthode de normalisation atteint constamment des performances supérieures, et ce même en la soumettant à des distributions de données très différentes (tableau 3.3 et Fig. 3.4). De plus, notre méthode de normalisation permet d'obtenir des images visuellement réalistes et plausibles tout en offrant des performances très proches de l'état de l'art (Drozdzal *et al.*, 2018) comme le montre la figure 3.3.

Nos expériences ont également montré que notre stratégie de normalisation atteint une bonne performance sur plus de deux domaines ou sur de multiples modalités d'imagerie. Lorsqu'entrainée sur trois ensembles de données, notre méthode atteint un indice Dice plus élevé de 89% comparativement à 86.7% lorsque notre méthode est entraînée sur les ensembles iSEG et MRBrainS 3.3. De plus, la variabilité des intensités dans les images normalisées est considérablement réduite avec une distance de Jenson-Shannon de 0.418 comparativement à 12.221 dans le cas des images brutes (tableau 3.4). Par ailleurs, l'ajout de la modalité T2 permet d'accroître la performance de segmentation de 2% sur l'indice Dice par rapport à un entraînement avec seulement la modalité T1.

Nous évaluons également la robustesse de notre méthode lorsqu'elle se voit présenter des images dégradées. En plus de normaliser une image de différents sites, notre méthode de normalisation antagoniste permet de retirer un effet de champ d'une image et de diminuer le bruit Ricien d'une image dégradée sans nécessiter de traitement supplémentaire (3.6, 3.7, 3.7). Ceci pourrait aider à effectuer des analyses à grande échelle beaucoup plus rapidement et de manière plus fiable comparativement à des pipelines de traitement d'image classiques. L'étude de l'hyperparamètre λ de notre modèle nous a permis d'étudier le comportement de celui-ci quant au compromis

entre l'exactitude de la segmentation et le niveau de réalisme dans l'image. Notre meilleur résultat sur l'indice Dice fut atteint avec $\lambda = 1,5$.

Notre approche de normalisation dirigée par la tâche de segmentation permet l'entraînement d'algorithmes d'apprentissage profond avec des données issues de plusieurs domaines d'imagerie tout en conservant le réalisme et en améliorant la précision de la tâche.

4.3 Limitations et améliorations futures

Une limitation potentielle de notre méthode réside dans le traitement de segments d'images au lieu de l'image dans son entièreté. Cela limite le contexte auquel notre algorithme a droit afin de normaliser les intensités et d'effectuer la classification des voxels. Bien que nous obtenons des images normalisées aux intensités et un résultat de segmentation spatialement lisse, les résultats globaux peuvent être sous-optimaux car le contexte global n'est pas pleinement considéré lorsqu'on travaille avec des segments de l'image. Dans un travail futur, nous proposons d'aborder ce problème en incorporant les statistiques globales de l'image dans la fonction de coût et en explorant une approche 2.5D (Xue, Farhat, Boukrina, Barrett, Binder, Roshan & Graves, 2020) où des tranches des différents plans sont traitées simultanément.

Aussi, notre méthode s'est avérée très difficile à entraîner afin d'avoir des résultats reproduisibles et stables. En effet, les approches antagonistes sont réputées pour leurs difficultés à converger correctement et sont souvent confrontées à des problèmes de disparition des gradients (*gradient vanishing*) ou d'effondrement des modes (*mode collapse*). Dans un travail futur, nous proposons d'étudier une méthode antagoniste basée sur la distance de Wasserstein (Arjovsky, Chintala & Bottou, 2017), une méthode moins sujette à ces problèmes. Il serait également intéressant de modifier l'architecture U-Net en concaténant la première couche avec la couche de sortie. Ainsi, la dernière couche n'apprendrait que le résiduel servant à la normalisation des contrastes au lieu d'apprendre à reproduire intégralement les intensités.

Afin de produire un résultat avec un niveau de confiance plus élevé, la théorie des ensembles pourrait être appliquée. Ainsi, on pourrait étendre la solution présentée à plusieurs réseaux de segmentation et élaborer une stratégie de vote. L'extension à un ensemble de modèles rendrait la segmentation plus robuste. Puisque la génération de l'image normalisée se base intrinsèquement sur la tâche accomplie, en l'occurrence la segmentation, la fonction de normalisation ainsi produite serait naturellement plus robuste.

Il serait également pertinent d'adapter notre méthode de normalisation aux images de diffusion. Cette modalité permet entre autres la modélisation des fibres dans le cerveau. Cette modalité souffre également d'une disponibilité limitée. Avoir une méthode de normalisation adaptée spécifiquement à cette modalité permettrait notamment d'augmenter le nombre d'échantillons dans les études utilisant cette modalité, ce qui rendrait celles-ci plus significatives. La fonction de normalisation pourrait être imbriquée dans un cadriciel riemannien permettant de synthétiser des images haute résolution (Anctil-Robitaille, Desrosiers & Lombaert, 2020), ce qui produirait une image normalisée répondant aux caractéristiques d'une variété riemannienne propre aux images de diffusions.

Une autre avenue possible serait d'explorer davantage le meta-learning, une méthode où l'algorithme *apprend à apprendre*. Il serait intéressant d'intégrer notre approche de normalisation et d'optimisation de la segmentation à même une approche d'apprentissage sur les métadonnées des images. De par le design fondamental de cette approche, l'apprentissage basée sur les métadonnées pourrait aider à diminuer le décalage de domaines d'imageries (*domain shift*).

Finalement, nous aimerions démontrer notre méthode dans une plus grande variété d'applications, incluant la segmentation de lésions cérébrales. Ces régions, beaucoup plus petites, posent un grand défi afin d'avoir une segmentation précise. Il serait intéressant de voir le comportement d'une méthode de normalisation comme la nôtre par rapport à ce type de tâche.

ANNEXE I

PREUVE DU THÉORÈME 1

Soit $p_r(\mathbf{x} \mid z)$ et $p_g(\mathbf{x} \mid z)$ les probabilités que **x** *est une image réelle ou générée, respectivement, provenant de l'ensemble de données z. L'optimisation minimax de l'Eq. (2.3) sans le terme de la segmentation correspond à la minimisation de la divergence entre $p_g(\mathbf{x} \mid z)$ pour chaque z et la distribution moyenne des images réelles $\overline{p}_r(\mathbf{x}) = \frac{1}{k} \sum_{z=1}^{K} p_r(\mathbf{x} \mid z)$.*

Démonstration. Si on ignore le coût de la segmentation $\mathcal{L}_{seg}$, le problème d'optimisation est posé par

$$\begin{aligned} \min_G \max_D \mathcal{L}(G, D) &= \mathbb{E}_{\mathbf{x},z}\Big[\log D_z(\mathbf{x}) + \log D_{K+1}(G(\mathbf{x}))\Big] \\ &= \mathbb{E}_{\mathbf{x},z}\Big[\log D_z(\mathbf{x}) + \log\Big(1 - \sum_{z'=1}^{K} D_{z'}(G(\mathbf{x}))\Big)\Big] \end{aligned} \tag{A I-1}$$

Supposons que le générateur G est static, le discriminateur optimal D^* peut être défini en minimisant

$$\mathcal{L}_G(D) = -\sum_{z=1}^{K} p(z) \int_{\mathbf{x}} \Big[p_r(\mathbf{x} \mid z) \log D_z(\mathbf{x}) + p_g(\mathbf{x} \mid z) \log\Big(1 - \sum_{z'=1}^{K} D_{z'}(\mathbf{x})\Big)\Big] d\mathbf{x}. \tag{A I-2}$$

Nous obtenons l'optimum pour chaque **x** en dérivant cette fonction par rapport à $D_z(\mathbf{x})$

$$\frac{\partial \mathcal{L}_D}{\partial D_z(\mathbf{x})} = -\frac{p(z)\, p_r(\mathbf{x} \mid z)}{D_z(\mathbf{x})} + \frac{p(z)\, p_g(\mathbf{x} \mid z)}{1 - \sum_{z'=1}^{K} D_{z'}(\mathbf{x})}. \tag{A I-3}$$

En fixant à zéro, cela donne

$$\frac{D_z^*(\mathbf{x})}{1 - \sum_{z'=1}^{K} D_{z'}^*(\mathbf{x})} = \frac{p_r(\mathbf{x} \mid z)}{p_g(\mathbf{x} \mid z)}. \tag{A I-4}$$

La sommation des deux côtés de l'équation sur z et en utilisant $D_{K+1}(\mathbf{x}) = 1 - \sum_z D_z(\mathbf{x})$, nous obtenons ainsi

$$\frac{\sum_{z=1}^{K} D_z^*(\mathbf{x})}{1 - \sum_{z'=1}^{K} D_{z'}^*(\mathbf{x})} \;=\; \sum_{z=1}^{K} \frac{p_r(\mathbf{x} \mid z)}{p_g(\mathbf{x} \mid z)} \;=\; \frac{1 - D_{K+1}^*(\mathbf{x})}{D_{K+1}^*(\mathbf{x})} \tag{A I-5}$$

et donc

$$D_z^*(\mathbf{x}) \;=\; \frac{\frac{p_r(\mathbf{x} \mid z)}{p_g(\mathbf{x} \mid z)}}{1 \;+\; \sum_{z'=1}^{K} \frac{p_r(\mathbf{x} \mid z')}{p_g(\mathbf{x} \mid z')}}, \quad z = 1, \ldots, K; \tag{A I-6}$$

$$D_{K+1}^*(\mathbf{x}) \;=\; \frac{1}{1 \;+\; \sum_{z'=1}^{K} \frac{p_r(\mathbf{x} \mid z')}{p_g(\mathbf{x} \mid z')}} \tag{A I-7}$$

Par la suite, nous utilisons ce résultat pour trouver les paramètres du générateur optimal. Pour atteindre ce but, nous utilisons (A I-7) dans la fonction de coût de l'équation Eq. (A I-1) et nous minimisons

$$\begin{aligned}
\mathcal{L}_{D^*}(G) \;&=\; \sum_{z=1}^{K} p(z) \int_{\mathbf{x}} p_g(\mathbf{x} \mid z) \;\log\; D_{K+1}^*(\mathbf{x}) \;\mathrm{d}\mathbf{x} \\
&=\; \sum_{z=1}^{K} p(z) \int_{\mathbf{x}} p_g(\mathbf{x} \mid z) \;\log \left(\frac{1}{1 \;+\; \sum_{z'=1}^{K} \frac{p_r(\mathbf{x} \mid z')}{p_g(\mathbf{x} \mid z')}} \right) \mathrm{d}\mathbf{x} \\
&=\; \sum_{z=1}^{K} p(z) \int_{\mathbf{x}} p_g(\mathbf{x} \mid z) \;\log \left(\frac{p_g(\mathbf{x} \mid z)}{p_g(\mathbf{x} \mid z) \;+\; \sum_{z'=1}^{K} \frac{p_g(\mathbf{x} \mid z)}{p_g(\mathbf{x} \mid z')} \, p_r(\mathbf{x} \mid z')} \right) \mathrm{d}\mathbf{x}
\end{aligned} \tag{A I-8}$$

Soit $q(\mathbf{x} \mid z)$ la distribution de probabilités définie par

$$q(\mathbf{x} \mid z) \;=\; \frac{1}{\mathcal{Z}(z)} \Big[p_g(\mathbf{x} \mid z) \;+\; \sum_{z'=1}^{K} \frac{p_g(\mathbf{x} \mid z)}{p_g(\mathbf{x} \mid z')} \, p_r(\mathbf{x} \mid z') \Big], \tag{A I-9}$$

où $\mathcal{Z}(z)$ est une constante de normalisation. Supposons que le générateur produit des images similaires aux images en entrée $p_g(\mathbf{x} \mid z') \approx p_r(\mathbf{x} \mid z')$, cette contante peut être estimée comme étant

$$\mathcal{Z}(z) \;=\; \int_{\mathbf{x}} p_g(\mathbf{x} \mid z) \, \mathrm{d}\mathbf{x} \;+\; \sum_{z'=1}^{K} \int_{x} \frac{p_g(\mathbf{x} \mid z)}{p_g(\mathbf{x} \mid z')} \, p_r(\mathbf{x} \mid z') \, \mathrm{d}\mathbf{x} \;\approx\; K{+}1. \tag{A I-10}$$

Ainsi, nous pouvons réécrire la fonction de coût du générateur dans (A I-8) comme

$$\begin{aligned} \mathcal{L}_{D^*}(G) &= \sum_{z=1}^{K} p(z) \int_{\mathbf{x}} p_g(\mathbf{x} \mid z) \log\left(\frac{p_g(\mathbf{x} \mid z)}{(K{+}1)\ q(\mathbf{x} \mid z)}\right) \mathrm{d}\mathbf{x} \\ &= \sum_{z=1}^{K} p(z)\, D_{\mathrm{KL}}\big(p_g(\cdot \mid z) \,||\, q(\cdot \mid z)\big) \ - \ \log(K{+}1), \end{aligned} \tag{A I-11}$$

où $D_{\mathrm{KL}}(p \,||\, q) \geq 0$ est la divergence de Kullback–Leibler. En assumant que $p(z)$ est uniforme, le générateur optimal G^* est $p_g(\cdot \mid z) = q(\cdot \mid z)$, pour $z = 1, \ldots, K$. En considérant l'Eq. (A I-9), ceci peut seulement être atteint si $p_g(\cdot \mid z) = p_g(\cdot \mid z')$, $\forall z, z'$. Ceci donne en retour

$$q(\mathbf{x} \mid z) \ = \ \frac{1}{K{+}1}\Big[p_g(\mathbf{x} \mid z) \ + \ \sum_{z'=1}^{K} p_r(\mathbf{x} \mid z')\Big]. \tag{A I-12}$$

En utilisant le fait que $p_g(\mathbf{x} \mid z) = q(\mathbf{x} \mid z)$, nous obtenons finalement

$$p_g(\mathbf{x} \mid z) \ = \ \frac{1}{K} \sum_{z'=1}^{K} p_r(\mathbf{x} \mid z') \ = \ \overline{p}_r(\mathbf{x}). \tag{A I-13}$$

Ainsi, G^* produira des sorties issues de la moyenne de la distribution des intensités des images réelles des différentes sources de données.

BIBLIOGRAPHIE

Anctil-Robitaille, B., Desrosiers, C. & Lombaert, H. (2020). Manifold-Aware CycleGAN for High Resolution Structural-to-DTI Synthesis. *arXiv preprint arXiv :2004.00173*, 1–4.

Arjovsky, M., Chintala, S. & Bottou, L. (2017). Wasserstein Generative Adversarial Networks. *Proceedings of Machine Learning Research*, 70, 214–223.

Atam P. Dhawan. (2011). *Medical Image Analysis, Second Edition*. Wiley-IEEE Press.

Attwell, D. & Laughlin, S. B. (2001). An energy budget for signaling in the grey matter of the brain. *Journal of Cerebral Blood Flow and Metabolism*, 21(10), 1133–1145.

Birenbaum, A. & Greenspan, H. (2016). Longitudinal Multiple Sclerosis Lesion Segmentation Using Multi-view Convolutional Neural Networks. *Deep Learning and Data Labeling for Medical Applications*, pp. 58–67.

Carass, A., Roy, S., Gherman, A., Reinhold, J. C., Jesson, A., Arbel, T., Maier, O., Handels, H., Ghafoorian, M., Platel, B., Birenbaum, A., Greenspan, H., Pham, D. L., Crainiceanu, C. M., Calabresi, P. A., Prince, J. L., Roncal, W. R. G., Shinohara, R. T. & Oguz, I. (2020). Evaluating White Matter Lesion Segmentations with Refined Sørensen-Dice Analysis. *Scientific Reports*, 10(1), 8242.

Casamitjana, A., Puch, S., Aduriz, A. & Vilaplana, V. (2016). 3D Convolutional Neural Networks for Brain Tumor Segmentation : A Comparison of Multi-resolution Architectures. *Brainlesion : Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*, pp. 150–161.

Chen, H., Dou, Q., Yu, L., Qin, J. & Heng, P.-A. (2018a). VoxResNet : Deep voxelwise residual networks for brain segmentation from 3D MR images. *NeuroImage*, 170, 446–455.

Chen, Y., Shi, F., Christodoulou, A., Zhou, Z., Xie, Y. & Li, D. (2018b). Efficient and Accurate MRI Super-Resolution Using a Generative Adversarial Network and 3D Multi-level Densely Connected Network. pp. 91-99.

Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T. & Ronneberger, O. (2016). 3D U-Net : Learning Dense Volumetric Segmentation from Sparse Annotation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2016*, pp. 424–432.

Ciga, O., Chen, J. & Martel, A. (2019). Multi-layer Domain Adaptation for Deep Convolutional Networks. *Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data*, pp. 20–27.

Clarke, W. T., Mougin, O., Driver, I. D., Rua, C., Morgan, A. T., Asghar, M., Clare, S., Francis, S., Wise, R. G., Rodgers, C. T., Carpenter, A., Muir, K. & Bowtell, R. (2020). Multi-site harmonization of 7 tesla MRI neuroimaging protocols. *NeuroImage*, 206(March 2019), 116335.

Cybenko, G. (1989). Approximation by Superpositions of a Sigmoidal Function. *Mathematics, Signals, Control and Systems*, 2, 303–314.

Dai, Z., Yang, Z., Yang, F., Cohen, W. W. & Salakhutdinov, R. (2017). Good semi-supervised learning that requires a bad GAN. *Advances in Neural Information Processing Systems*, 2017-December, 6511–6521.

Delisle, P., Anctil-Robitaille, B., Desrosiers, C. & Lombaert, H. (2020). Adversarial Normalization for Multi Domain Image Segmentation. *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, pp. 849–853.

Dewey, B. E., Zhao, C., Reinhold, J. C., Carass, A., Fitzgerald, K. C., Sotirchos, E. S., Saidha, S., Oh, J., Pham, D. L., Calabresi, P. A., van Zijl, P. C. & Prince, J. L. (2019). DeepHarmony : A deep learning approach to contrast harmonization across scanner changes. *Magnetic Resonance Imaging*, 64(May), 160–170.

Di Martino et al., A., Yan, C. G., Li, Q., Denio, E., Castellanos, F. X., Alaerts, K., Anderson, J. S., Assaf, M., Bookheimer, S. Y., Dapretto, M., Deen, B., Delmonte, S., Dinstein, I., Ertl-Wagner, B., Fair, D. A., Gallagher, L., Kennedy, D. P., Keown, C. L., Keysers, C., Lainhart, J. E., Lord, C., Luna, B., Menon, V., Minshew, N. J., Monk, C. S., Mueller, S., Müller, R. A., Nebel, M. B., Nigg, J. T., O'Hearn, K., Pelphrey, K. A., Peltier, S. J., Rudie, J. D., Sunaert, S., Thioux, M., Tyszka, J. M., Uddin, L. Q., Verhoeven, J. S., Wenderoth, N., Wiggins, J. L., Mostofsky, S. H. & Milham, M. P. (2014). The autism brain imaging data exchange : Towards a large-scale evaluation of the intrinsic brain architecture in autism. *Molecular Psychiatry*, 19(6), 659–667.

Diederik, K. & Ba, J. L. (2014). ADAM : A Method for Stochastic Optimization. *AIP Conference Proceedings*, 1631, 58–62.

Dolz, J., Gopinath, K., Yuan, J., Lombaert, H., Desrosiers, C. & Ben Ayed, I. (2019). HyperDense-Net : A Hyper-Densely Connected CNN for Multi-Modal Image Segmentation. *IEEE Transactions on Medical Imaging*, 38(5), 1116–1126.

Drozdzal, M., Vorontsov, E., Chartrand, G., Kadoury, S. & Pal, C. (2016). The importance of skip connections in biomedical image segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 179–187.

Drozdzal, M., Chartrand, G., Vorontsov, E., Shakeri, M., Di Jorio, L., Tang, A., Romero, A., Bengio, Y., Pal, C. & Kadoury, S. (2018). Learning normalized inputs for iterative estimation in medical image segmentation. *Medical Image Analysis*, 44, 1–13.

Fan, J., Cao, X., Xue, Z., Yap, P.-T. & Shen, D. (2018). Adversarial Similarity Network for Evaluating Image Alignment in Deep Learning based Registration. 11070, 739–746.

Feyjie, A. R., Azad, R., Pedersoli, M., Kauffman, C., Ayed, I. B. & Dolz, J. (2020). Semi-supervised few-shot learning for medical image segmentation. *arXiv preprint arXiv :2003.08462*, 1–10.

Fischl, B. (2012). FreeSurfer. *NeuroImage*, 62(2), 774-781.

Ganin, Y. & Lempitsky, V. (2015). Unsupervised domain adaptation by backpropagation. *32nd International Conference on Machine Learning, ICML 2015*, 2(i), 1180–1189.

Glorot, X. & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *Journal of Machine Learning Research*, 9, 249–256.

Glorot, X., Bordes, A. & Bengio, Y. (2010). Deep Sparse Rectifier Neural Networks. *Journal of Machine Learning Research*, 15, 315-323.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y. (2014). Generative Adversarial Nets. *Advances in Neural Information Processing Systems*, 27, 1–9.

Gudbjartsson, H. & Patz, S. (1995). The Rician distribution of noisy MRI data. *Magnetic resonance in medicine*, 34(6), 910–914.

Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P. M. & Larochelle, H. (2017). Brain tumor segmentation with Deep Neural Networks. *Medical Image Analysis*, 35, 18–31.

He, K., Zhang, X., Ren, S. & Sun, J. (2015). Delving Deep into Rectifiers : Surpassing Human-Level Performance on ImageNet Classification. *IEEE International Conference on Computer Vision (ICCV 2015)*, 1502, 1026-1034.

He, K., Zhang, X., Ren, S. & Sun, J. (2016). Deep residual learning for image recognition. December, 770–778.

Hesse, L. S., Kuling, G., Veta, M. & Martel, A. (2020). Intensity augmentation to improve generalizability of breast segmentation across different MRI scan protocols. *IEEE Transactions on Biomedical Engineering*, 9294(c), 1–1.

Hu, B., Tang, Y., Eric, I., Chang, C., Fan, Y., Lai, M. & Xu, Y. (2018). Unsupervised learning for cell-level visual representation in histopathology images with generative adversarial networks. *IEEE journal of biomedical and health informatics*, 23(3), 1316–1328.

Ioffe, S. & Szegedy, C. (2015). Batch normalization : Accelerating deep network training by reducing internal covariate shift. *32nd International Conference on Machine Learning, ICML 2015*, 37, 448–456.

Kamnitsas, K., Bai, W., Ferrante, E., McDonagh, S., Sinclair, M., Pawlowski, N., Rajchl, M., Lee, M., Kainz, B., Rueckert, D. & Glocker, B. (2018). Ensembles of multiple models and architectures for robust brain tumour segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10670 LNCS, 450–462.

Kamnitsas, K., Ledig, C., Newcombe, V. F., Simpson, J. P., Kane, A. D., Menon, D. K., Rueckert, D. & Glocker, B. (2017). Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. *Medical Image Analysis*, 36, 61–78.

LeCun, Y., Bottou, L., Bengio, Y. & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2323.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A., van Ginneken, B. & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42, 60–88.

Logue, M. W., van Rooij, S. J., Dennis, E. L., Davis, S. L., Hayes, J. P., Stevens, J. S., Densmore, M., Haswell, C. C., Ipser, J., Koch, S. B., Korgaonkar, M., Lebois, L. A., Peverill, M., Baker, J. T., Boedhoe, P. S., Frijling, J. L., Gruber, S. A., Harpaz-Rotem, I., Jahanshad, N., Koopowitz, S., Levy, I., Nawijn, L., O'Connor, L., Olff, M., Salat, D. H., Sheridan, M. A., Spielberg, J. M., van Zuiden, M., Winternitz, S. R., Wolff, J. D., Wolf, E. J., Wang, X., Wrocklage, K., Abdallah, C. G., Bryant, R. A., Geuze, E., Jovanovic, T., Kaufman, M. L., King, A. P., Krystal, J. H., Lagopoulos, J., Bennett, M., Lanius, R., Liberzon, I., McGlinchey, R. E., McLaughlin, K. A., Milberg, W. P., Miller, M. W., Ressler, K. J., Veltman, D. J., Stein, D. J., Thomaes, K., Thompson, P. M. & Morey, R. A. (2017). Smaller Hippocampal Volume in Posttraumatic Stress Disorder : A Multisite ENIGMA-PGC Study : Subcortical Volumetry Results From Posttraumatic Stress Disorder Consortia. *Biological Psychiatry*, 83(3), 244–253.

Madani, A., Moradi, M., Karargyris, A. & Syeda-Mahmood, T. (2018). Semi-supervised learning with generative adversarial networks for chest X-ray classification with ability of data domain adaptation. *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 1038-1042.

Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z. & Paul Smolley, S. (2017a). Least squares generative adversarial networks. *Proceedings of the IEEE international conference on computer vision*, pp. 2794–2802.

Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z. & Smolley, S. P. (2017b). Least Squares Generative Adversarial Networks. *Proceedings of the IEEE International Conference on Computer Vision*, 2017-October, 2813–2821.

Mendrik, A., Vincken, K., Kuijf, H., Breeuwer, M., Bouvy, W., de Bresser, J., Alansary, A., de Bruijne, M., Carass, A., El-Baz, A., Jog, A., Katyal, R., Khan, A., Lijn, F., Mahmood, Q., Mukherjee, R., Opbroek, A., Paneri, S., Pereira, S. & Viergever, M. (2015). MRBrainS Challenge : Online Evaluation Framework for Brain Image Segmentation in 3T MRI Scans. *Computational Intelligence and Neuroscience*, 1–16.

Milletari, F., Navab, N. & Ahmadi, S. (2016). V-Net : Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. *2016 Fourth International Conference on 3D Vision (3DV)*, pp. 565–571.

Mirzaalian, H., Ning, L., Savadjiev, P., Pasternak, O., Bouix, S., Michailovich, O., Karmacharya, S., Grant, G., Marx, C. E., Morey, R. A., Flashman, L. A., George, M. S., McAllister, T. W., Andaluz, N., Shutter, L., Coimbra, R., Zafonte, R. D., Coleman, M. J., Kubicki, M., Westin, C. F., Stein, M. B., Shenton, M. E. & Rathi, Y. (2018). Multi-site harmonization of diffusion MRI data in a registration framework. *Brain Imaging and Behavior*, 12(1), 284–295.

Modanwal, G., Vellal, A., Buda, M. & Mazurowski, M. A. (2020). MRI image harmonization using cycle-consistent generative adversarial network. Dans Hahn, H. K. & Mazurowski, M. A. (Éds.), *Medical Imaging 2020 : Computer-Aided Diagnosis* (vol. 11314, pp. 259 – 264). SPIE.

Nyul, L. G., Udupa, J. K. & Xuan Zhang. (2000). New variants of a method of MRI scale standardization. *IEEE Transactions on Medical Imaging*, 19(2), 143–150.

Oguz, I., Malone, J. D., Atay, Y. & Tao, Y. K. (2020). Self-fusion for OCT noise reduction. *SPIE Medical Imaging*, 11313, 45–50.

Onofrey, J. A., Casetti-Dinescu, D. I., Lauritzen, A. D., Sarkar, S., Venkataraman, R., Fan, R. E., Sonn, G. A., Sprenkle, P. C., Staib, L. H. & Papademetris, X. (2019). Generalizable Multi-Site Training and Testing Of Deep Neural Networks Using Image Normalization. *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pp. 348–351.

Pereira, S., Pinto, A., Alves, V. & Silva, C. A. (2016). Brain Tumor Segmentation Using Convolutional Neural Networks in MRI Images. *IEEE transactions on medical imaging*, 35(5), 1240–1251.

Pomponio, R., Erus, G., Habes, M., Doshi, J., Srinivasan, D., Mamourian, E., Bashyam, V., Nasrallah, I. M., Satterthwaite, T. D., Fan, Y., Launer, L. J., Masters, C. L., Maruff, P., Zhuo, C., Völzke, H., Johnson, S. C., Fripp, J., Koutsouleris, N., Wolf, D. H., Gur, R., Gur, R., Morris, J., Albert, M. S., Grabe, H. J., Resnick, S. M., Bryan, R. N., Wolk, D. A., Shinohara, R. T., Shou, H. & Davatzikos, C. (2020). Harmonization of large MRI datasets for the analysis of brain imaging patterns throughout the lifespan. *NeuroImage*, 208(December 2019), 116450.

Radford, A., Metz, L. & Chintala, S. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. *4th International Conference on Learning Representations (ICLR) 2016*, 1–16.

Raghu, M., Zhang, C., Kleinberg, J. & Bengio, S. (2019). Transfusion : Understanding Transfer Learning for Medical Imaging. Dans Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. & Garnett, R. (Éds.), *Advances in Neural Information Processing Systems 32* (pp. 3347–3357). Curran Associates, Inc.

Ronneberger, O., Fischer, P. & Brox, T. (2015). U-Net : Convolutional Networks for Biomedical Image Segmentation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2015*, pp. 234–241.

S. C. Karayumak, S. Bouix, L. Ning, A. James, T. Crow, M. Shenton, M. Kubicki, Y. R. (2017). Retrospective harmonization of multi-site diffusion MRI data acquired with different acquisition parameters. *Physiology and Behavior*, 176(3), 139–148.

Sánchez, I. & Vilaplana, V. (2018). Brain MRI super-resolution using 3D generative adversarial networks. *arXiv preprint arXiv :1812.11440*, 1–8.

Shah, M., Xiao, Y., Subbanna, N., Francis, S., Arnold, D. L., Collins, D. L. & Arbel, T. (2011). Evaluating intensity normalization on MRIs of human brain with multiple sclerosis. *Medical Image Analysis*, 15(2), 267–282.

Shelhamer, E., Long, J. & Darrell, T. (2017). Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4), 640–651.

Shimodaira, H. (2000). Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2), 227–244.

Shinohara, R. T., Oh, J., Nair, G., Calabresi, P. A., Davatzikos, C., Doshi, J., Henry, R. G., Kim, G., Linn, K. A., Papinutto, N., Pelletier, D., Pham, D. L., Reich, D. S., Rooney, W., Roy, S., Stern, W., Tummala, S., Yousuf, F., Zhu, A., Sicotte, N. L. & Bakshi, R. (2017). Volumetric analysis from a harmonized multisite brain MRI study of a single subject with multiple sclerosis. *American Journal of Neuroradiology*, 38(8), 1501–1509.

Shinohara, R. T., Sweeney, E. M., Goldsmith, J., Shiee, N., Mateen, F. J., Calabresi, P. A., Jarso, S., Pham, D. L., Reich, D. S. & Crainiceanu, C. M. (2014). Statistical normalization techniques for magnetic resonance imaging. *NeuroImage : Clinical*, 6, 9–19.

Sinclair, A., Morrison, A., Young, C. & P., L. (2018, March). Canadian Medical Imaging Inventory | CADTH. [HTML]. Repéré https://cadth.ca/canadian-medical-imaging-inventory-2017.

Telgarsky, M. (2016). Benefits of depth in neural networks. *Journal of Machine Learning Research*, 49(June), 1517–1539.

Wang, L., Nie, D., Li, G., Puybareau, E., Dolz, J., Zhang, Q., Wang, F., Xia, J., Wu, Z., Chen, J. W., Thung, K. H., Bui, T. D., Shin, J., Zeng, G., Zheng, G., Fonov, V. S., Doyle, A., Xu, Y., Moeskops, P., Pluim, J. P. W., Desrosiers, C., Ayed, I. B., Sanroma, G., Benkarim, O. M., Casamitjana, A., Vilaplana, V., Lin, W., Li, G. & Shen, D. (2019). Benchmark on Automatic Six-Month-Old Infant Brain Segmentation Algorithms : The iSeg-2017 Challenge. *IEEE Transactions on Medical Imaging*, 38(9), 2219-2230.

Wang, M. & Deng, W. (2018). Deep visual domain adaptation : A survey. *Neurocomputing*, 312, 135–153.

Xue, H., Srinivasan, L., Jiang, S., Rutherford, M., Edwards, A. D., Rueckert, D. & Hajnal, J. V. (2007). Automatic segmentation and reconstruction of the cortex from neonatal MRI. *NeuroImage*, 38(3), 461–477.

Xue, Y., Farhat, F. G., Boukrina, O., Barrett, A., Binder, J. R., Roshan, U. W. & Graves, W. W. (2020). A multi-path 2.5 dimensional convolutional neural network system for segmenting stroke lesions in brain MRI images. *NeuroImage : Clinical*, 25, 102118.

Yang, D., Xu, D., Zhou, S. K., Georgescu, B., Chen, M., Grbic, S., Metaxas, D. & Comaniciu, D. (2017). Automatic liver segmentation using an adversarial image-to-image network. *Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2017*, pp. 507–515.

Yi, X., Walia, E. & Babyn, P. (2019). Generative adversarial network in medical imaging : A review. *Medical Image Analysis*, 58, 101552.

Zhang, Z., Yang, L. & Zheng, Y. (2018). Translating and segmenting multimodal medical volumes with cycle-and shape-consistency generative adversarial network. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9242–9251.

Zhu, J., Park, T., Isola, P. & Efros, A. A. (2017). Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2242-2251.

Zhu, X. & Goldberg, A. (2009). Introduction to Semi-Supervised Learning. Dans *Introduction to Semi-Supervised Learning*. Morgan & Claypool.