



HAL
open science

COLISEUM-3D : Une plate-forme innovante pour la segmentation d'images médicales par Raisonnement à Partir de Cas (RàPC) et méthodes d'apprentissage de type Deep Learning

Florent Marie

► **To cite this version:**

Florent Marie. COLISEUM-3D : Une plate-forme innovante pour la segmentation d'images médicales par Raisonnement à Partir de Cas (RàPC) et méthodes d'apprentissage de type Deep Learning. Intelligence artificielle [cs.AI]. Université Bourgogne Franche-Comté, 2019. Français. NNT : 2019UBFCD051 . tel-03001092

HAL Id: tel-03001092

<https://theses.hal.science/tel-03001092>

Submitted on 12 Nov 2020

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

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

SPIM

Thèse de Doctorat



COLISEUM-3D

Une plate-forme innovante pour la segmentation d'images médicales par Raisonnement à Partir de Cas (*RàPC*) et méthodes d'apprentissage de type Deep Learning

■ FLORENT MARIE

UBFC

UNIVERSITÉ
BOURGOGNE FRANCHE-COMTÉ



UNIVERSITÉ DE
FRANCHE-COMTÉ

THESE DE DOCTORAT DE L'ETABLISSEMENT UNIVERSITE BOURGOGNE FRANCHE-COMTE
PREPAREE A L'UNIVERSITE DE FRANCHE-COMTE

École doctorale n°37
Sciences Pour l'Ingénieur et Microtechniques
Spécialité : Informatique

par

FLORENT MARIE

COLISEUM-3D

Une plate-forme innovante pour la segmentation d'images médicales par
Raisonnement à Partir de Cas (*RàPC*) et méthodes d'apprentissage de type
Deep Learning

Unité de Recherche :
Femto-st/DISC

Soutenue publiquement le 19/12/2019 à Besançon,
devant le Jury composé de :

JEAN LIEBER

JEAN-PHILIPPE THIRAN

HUGUES TALBOT

FRÉDÉRIC AUBER

JEAN-CHRISTOPHE LAPAYRE

JULIEN HENRIET

Rapporteur

Rapporteur

Examineur

Examineur

Directeur

Co-encadrant

MCF HDR, LORIA Inria Lorraine

Prof., École Polytechnique Fédérale de
Lausanne (EPFL)

Prof., CentraleSupélec Inria Saclay

Prof., UBFC Centre Hospitalier Régional
Universitaire de Besançon (CHRUB)

Prof., Univ. de Bourgogne/Franche-Comté

MCF, Univ. de Bourgogne/Franche-Comté

REMERCIEMENTS

Tout d'abord, je remercie Jean LIEBER et Jean-Philippe THIRAN pour avoir accepté de rapporter mon travail ainsi que l'ensemble des membres du jury pour avoir accepté de l'évaluer.

Je remercie ensuite Jean Christophe LAPAYRE et Julien HENRIET pour la qualité de leur encadrement, leur soutien moral tout au long de ce travail de thèse ainsi que leur disponibilité et leur aide qui m'ont permis d'en surmonter les difficultés. Leurs conseils m'ont été précieux pour parvenir à livrer ce travail.

Je remercie également Yann CHAUSSY, praticien du CHU de Besançon, et Loredane VIEILLE, ingénieure en biomédicale, ainsi que l'ensemble de l'équipe hospitalière. Leur aide fut essentielle à la réussite de ce travail en fournissant l'expertise médicale requise au traitement des données.

Je désire ensuite particulièrement remercier mes amis et collègues du laboratoire DISC. Ces années de thèse n'auraient pas été les mêmes sans leur présence.

Tout d'abord, tous mes remerciements à Karla BRESCHI pour m'avoir fait découvrir un grand nombre de restaurants de Besançon et pour m'avoir permis de bien rire de ses gaffes. Mais surtout, je la remercie pour sa bonne humeur, son soutien et son attention dont elle a toujours fait preuve durant ces trois années et qui ont été importants pour moi. Des remerciements particuliers aussi pour Lucas LEAO qui a toujours été un très bon ami et un confident, présent dans les bons comme dans les mauvais moments, et ce même malgré la distance. Je remercie Arthur HUGEAT que je suis allé régulièrement embêter dans son bureau et qui a toujours pris le temps de répondre à mes questions. Je le remercie aussi pour nos soirées qui m'ont permis de me changer les idées lorsque j'en avais le plus besoin. Je remercie également Henri DE BOUTRAY pour sa bonne humeur et les quelques soirées que nous avons pu faire ensemble. Je remercie enfin Lisa CORBAT, amie et collègue de bureau, pour nos conversations (parfois animées) et pour m'avoir fait partager en toutes circonstances la vie de ses trois félins.

Je remercie l'ensemble de l'équipe du DISC pour leur sympathie et leur accueil.

Je remercie ma famille pour leur soutien malgré la distance et le peu de temps que j'ai pu leur consacrer ces dernières années.

Finalement, je remercie la Communauté d'Agglomération du Grand Besançon (CAGB) pour avoir financé mes travaux de thèse durant ces trois années.

SOMMAIRE

Introduction	7
Introduction générale	7
Plan du mémoire	9
Quelques indications pour la lecture de ce rapport	11
I Contexte et état de l'art	13
1 Le contexte de la segmentation d'images	17
1.1 Le contexte médical : l'aide au traitement de tumeurs infantiles	18
1.1.1 Le néphroblastome : définition, épidémiologie et traitement	18
1.1.2 L'Anatomie et les fonctions du rein	18
1.2 L'imagerie : quelques notions liminaires	24
1.2.1 La représentation numérique d'une image	24
1.2.2 Définition d'un problème de segmentation d'images	26
1.2.3 L'évaluation d'une segmentation	27
1.3 L'imagerie et la segmentation des structures abdominales	28
1.3.1 L'imagerie tomодensitométrique	28
1.3.2 La mise en évidence des structures abdominales d'intérêt	31
1.3.3 L'intérêt médical de la segmentation	32
1.3.4 Logiciels et méthodes de segmentation d'images utilisés en routine médicale	33
1.4 L'apprentissage machine dans le domaine de la segmentation	34
1.4.1 La représentation de données images	34
1.4.2 Les différents types d'apprentissage automatique	34
1.4.3 Les méthodes de segmentation	37
2 Les réseaux de neurones et l'apprentissage profond	43
2.1 Le perceptron	43
2.1.1 Notions de base	43
2.1.2 Le perceptron multicouche	45

2.2	L'apprentissage profond	46
2.2.1	Le principe général de l'apprentissage	46
2.2.2	La rétropropagation du gradient	47
2.3	Les réseaux de neurones convolutifs	48
2.3.1	Présentation générale	48
2.3.2	La convolution dans les réseaux de neurones	51
2.4	CNN pour la segmentation d'images	53
2.4.1	Fully Convolutional Network	53
2.4.2	Deconvolution Network	55
2.4.3	Decoupled Network	56
2.4.4	SegNet	57
2.4.5	U-Net	57
2.4.6	W-Net	59
2.4.7	DeepMedic	60
2.4.8	DeepLab	62
2.4.9	Comparatif des réseaux	64
3	Le raisonnement à partir de cas : une approche connaissance	69
3.1	Généralités	69
3.1.1	Historique	69
3.1.2	Le raisonnement par analogie et les bases du RàPC	70
3.2	Le fonctionnement du RàPC	72
3.2.1	Les différents types de connaissances	72
3.2.2	Le cycle du RàPC	78
3.3	Les RàPC pour la segmentation d'images	80
3.3.1	Une première architecture de P. Perner pour la segmentation par seuillage	82
3.3.2	Une nouvelle application du système de RàPC de P. Perner : segmentation par ligne de partage des eaux (<i>watershed</i>) de Frucci <i>et al.</i>	87
II	Contribution	93
4	COLISEUM-3D : une plate-forme innovante pour la segmentation d'images médicales	97
4.1	Présentation générale de la plate-forme	97
4.2	Les rôles et le fonctionnement des différentes couches	99

4.2.1	La couches de données	99
4.2.2	La couche de segmentation	100
4.2.3	La couche de fusion	101
4.3	Le recalage temps vasculaire/temps tardif	105
4.3.1	Détection automatique des points de repère anatomiques	106
4.3.2	Recalage des images utilisé dans cette première version	107
5	Contributions majeures sur la segmentation en utilisant l'Intelligence Artificielle d'une part avec le RàPC et d'autre part avec le Deep Learning	111
5.1	Raisonnement à Partir de Cas : Application à un problème de segmentation d'images par croissance de régions	111
5.1.1	L'unité de segmentation	112
5.1.2	La modélisation d'un cas	114
5.1.3	La phase de remémoration	116
5.1.4	La phase d'adaptation	117
5.1.5	Les phases de révision et de capitalisation	124
5.2	<i>OV²ASSION</i> pour la segmentation du néphroblastome	125
5.2.1	Présentation générale de <i>OV²ASSION</i>	126
5.2.2	Formalisation	127
5.2.3	Cardinalité	128
6	Résultats et discussions	131
6.1	Les métriques pour l'évaluation des segmentations	131
6.2	Expérimentations liminaires	133
6.2.1	Le choix du réseau de neurones convolutif	133
6.2.2	Le choix de l'approche à privilégier en fonction de la structure à segmenter	134
6.2.3	Évaluation des métriques de similarités	135
6.3	Évaluation du système de RàPC et plus particulièrement de sa phase d'adaptation	136
6.3.1	L'adaptation de la position des germes	138
6.3.2	L'adaptation des seuils des germes	143
6.4	L'évaluation d'un CNN entraîné suivant la méthode <i>OV²ASSION</i>	149
6.4.1	L'influence de l'écart g et du vecteur $(V_g)_i$ sur les performances	149
6.4.2	Vue d'ensemble des performances d' <i>OV²ASSION</i> sur plusieurs patients	153
6.5	Discussion générale	156

Conclusion et perspectives	161
Conclusion	161
Perspectives	162
Ma bibliographie personnelle	169
Bibliographie	179

INTRODUCTION

INTRODUCTION GÉNÉRALE

LES SYSTÈME DE VISION PAR ORDINATEUR

Dans notre société contemporaine, nous assistons à l'omniprésence de systèmes de vision par ordinateur. La vision par ordinateur recouvre l'ensemble des approches consistant à permettre à une machine de comprendre et d'interpréter le contenu d'une image obtenue par un système d'acquisition quelconque (caméra, appareil photo, sonar, ...). Les données acquises peuvent être de différentes formes telles que des images fixes, des séquences vidéo ou encore des données multidimensionnelles difficilement interprétables par un être humain. Dans ce contexte, la compréhension consiste en la transformation de ces données visuelles (images numériques) en données interprétables par un ordinateur. Ces données extraites peuvent être de nature statistique, géométrique ou encore physique et visent à concevoir un modèle permettant cette compréhension. Les approches sont souvent basées sur un biomimétisme, cherchant à reproduire la vision humaine par des processus conscients ou inconscients (comment un être humain parvient-il à reconnaître les objets? À délimiter leurs frontières? Comment le cerveau parvient à traiter les images?).

La vision par ordinateur est entre autre intimement liée à l'analyse automatique d'images et représente une problématique centrale dans tout développement d'Intelligences Artificielles et de robots qui doivent être capables de comprendre leur environnement par eux-mêmes. Les applications sont nombreuses, et vont du *tracking* vidéo à la reconnaissance d'objets (classification d'images) en passant par la segmentation. Ainsi, le développement des voitures autonomes impose de donner aux ordinateurs une haute compréhension du monde réel et surtout de leur environnement immédiat (routes, piétons, autres véhicules et obstacles quelconques, ...) afin de réagir de la manière la plus adéquates et garantir ainsi la sécurité des usagers. Dans le domaine médical, la multiplication de la quantité de données, liée au recours toujours plus important à l'imagerie ces dix dernières années, amène la question du développement de systèmes de vision informatique pour pouvoir traiter ces données de manière automatique et optimale, afin d'en exploiter toutes les informations pertinentes dans l'intérêt du patient. La segmentation d'image est l'un des aspects du traitement de ces données et consiste à extraire d'une image une information de forme pour une structure d'intérêt.

CONTEXTE DE LA THÈSE

Le rein est un organe ovoïde présent en deux exemplaires et assurant des fonctions essentielles au bon fonctionnement du corps humain. Il est responsable de l'épuration du sang par la production d'urine et a d'importantes fonctions hormonales. Il joue un rôle

majeur dans la régulation de la pression artérielle. L'urine est produite dans les tissus de l'organe (également appelé parenchyme, constitué de deux tissus distincts que sont le cortex et la médulla) et elle est déversée dans des cavités formant le début des voies urinaires.

Le néphroblastome est une tumeur infantile affectant le rein et représentant une urgence médicale à cause de son évolution rapide. Il s'agit d'une des tumeurs solides infantiles les plus fréquentes et représente donc un enjeu important. Il reste toutefois une pathologie avec un fort taux de guérison, pour les formes communes, grâce à un traitement alliant chimiothérapie, radiothérapie et chirurgie. L'acte chirurgical consiste en une ablation de la tumeur et éventuellement du rein touché. L'ablation du rein est conditionné par les recommandations de la SIOP (Société Internationale d'Oncologie Pédiatrique) qui a établi un certain nombre de critères à vérifier pour orienter la prise de décision. L'un des critères consiste à caractériser l'envahissement des cavités rénales (voies excrétrices) par les cellules tumorales. En cas de tumeur se limitant aux tissus périphériques du rein, une chirurgie conservatrice peut être envisagée avec la sauvegarde du rein pathologique. Si, au contraire, la tumeur a envahi les voies urinaires, alors la SIOP recommande une chirurgie totale avec une ablation des tissus rénaux. Cette volonté de sauvegarder au mieux le rein lorsqu'il est possible de le faire découle à la fois de la constatation d'une faible récurrence en cas de néphrectomie partielle, si ces critères sont respectés, et d'un risque accru de développer une insuffisance rénale au cours de la vie du patient en cas de rein unique.

Une façon d'évaluer cet envahissement par la tumeur des cavités rénales est de réaliser une segmentation des structures d'intérêt (parenchyme rénal, cavités rénales, tumeur et vaisseaux sanguins) sur un examen scanner, puis de s'appuyer sur ces segmentations pour construire une modélisation en 3D. Cette visualisation représente une aide intéressante pour caractériser le néphroblastome, vérifier les critères de la SIOP et planifier au mieux la chirurgie en fonction des patients, en permettant d'anticiper les difficultés (comme par exemple une tumeur proche d'un vaisseau sanguin). Cependant, les segmentations sont souvent réalisées manuellement ou à l'aide d'algorithmes automatiques guidés par un praticien ou un ingénieur biomédical mais nécessitant une forte intervention humaine. De plus, les solutions logicielles à disposition des radiologues ont tendance à être peu ergonomiques, complexes et nécessitent tout de même une expertise médicale avancée. Enfin, malgré ces outils, la réalisation des segmentations reste actuellement une tâche très chronophage (4 à 5H pour la segmentation d'un patient). La conséquence directe est que les segmentations sont rarement effectuées en routine, privant les médecins d'informations pertinentes pour orienter la thérapie des jeunes patients.

Le recours à l'Intelligence Artificielle (IA) apparaît donc comme un moyen intéressant qui pourrait décharger, au moins en partie, les radiologues de cette tâche et ainsi améliorer la prise en charge médicale des patients. Cette thèse est associée au projet nommé AS-Kid (**A**utomatic **S**egmentation using artificial intelligence of **KID**ney tumours in children) qui consiste à créer un système de segmentation automatique des images scanners pour l'ensemble des structures d'intérêt dans le cadre du traitement du néphroblastome chez de jeunes enfants, sans intervention humaine ou, *a minima*, avec une intervention aussi limitée que possible de la part d'un opérateur. La notion d'IA est indissociable de celle d'apprentissage automatique. Il existe différents types d'apprentissage, mais celui actuellement le plus utilisé est l'apprentissage dit supervisé. Dans ce type d'apprentissage, un programme apprend sur un ensemble de données étiquetées manuellement par un médecin afin d'obtenir la capacité d'effectuer ensuite lui-même cet étiquetage sur

de nouvelles données. Cependant, cette approche se heurte à un double problème. Tout d'abord, la quantité de données nécessaires pour obtenir de bonnes performances est élevée. Ensuite, le corollaire de ce premier problème est que faire effectuer par un humain l'étiquetage de ces données devient donc une tâche très coûteuse en temps.

OBJECTIF DE CES TRAVAUX

L'objectif de ce travail de thèse se confond en partie avec ceux du projet ASKid, à savoir la conception d'une plate-forme dédiée à la segmentation d'images obtenues par imagerie scanner. Cette plate-forme a été intitulée COLISEUM-3D (**COL**laborative platform with **artificial Intelligence for SEgmentation of tUmoral kidney in Medical images in 3D**). Elle a tout d'abord pour but de permettre la collaboration de différentes approches d'IA afin de réaliser la segmentation des différentes structures d'intérêt présentes dans les images (CT-scans). COLISEUM-3D doit ensuite fusionner les résultats des différentes segmentations pour représenter ces structures dans un même espace (c'est à dire dans un même jeu d'images) et faciliter ainsi leur interprétation. Nous avons choisi ici de concentrer nos efforts sur le formalisme de la plate-forme et la conception des modules de segmentations. La fusion des données représente une problématique à part entière et fait l'objet d'une seconde thèse complémentaire de celle-ci.

Au-delà de la tâche de segmentation en elle-même, l'une des principales problématiques d'ASKid est de parvenir à utiliser des approches d'IA malgré une quantité de données à disposition qui reste limitée, et ce pour plusieurs raisons. Premièrement, les données médicales sont des données sensibles nécessitant un certain nombre d'autorisations pour pouvoir être utilisées. Elles ne sont accessibles que par l'intermédiaire de partenariats avec des centres hospitaliers et doivent être impérativement anonymisées au préalable afin de protéger les informations concernant les patients. Deuxièmement, la quantité de données produites par un centre hospitalier donné est limitée, comparativement aux données produites dans d'autres contextes (comme les bases utilisées par Google pour la segmentation d'objets du quotidien), et il n'existe pas de base nationale concernant le néphroblastome qui serait approvisionnée par l'ensemble des centres hospitaliers de France et librement accessible. Il est donc nécessaire d'imaginer des approches palliant ce manque de données sur lesquelles entraîner des Intelligences Artificielles.

En parallèle de la conception globale de la plate-forme COLISEUM-3D, le travail de recherche présenté ici se concentre majoritairement sur la segmentation de deux des structures d'intérêt pour le projet ASKid que sont le parenchyme rénal et le néphroblastome.

PLAN DU MÉMOIRE

La première partie, composée de 3 chapitres est consacrée à la définition du contexte et à notre état de l'art. Dans un premier chapitre, nous abordons de manière plus complète le contexte de la thèse. Nous y détaillons la problématique médicale autour de l'aide au traitement du néphroblastome ainsi que son épidémiologie. Afin de favoriser la compréhension de la finalité de ce travail, nous y expliquons de manière succincte les grandes fonctions et l'importance du rein dans la physiologie humaine. Quelques notions liminaires sur l'imagerie y sont également introduites telles que la représentation numérique d'une image, la définition d'une segmentation et le processus physique permettant de

produire les données (CT-scans) traitées dans ce manuscrit. Ce chapitre s'achève sur une présentation de l'apprentissage machine et un état de l'art général sur les différentes approches de segmentation et d'IA disponibles dans la littérature.

Le second chapitre est consacré plus spécifiquement à la compréhension des Réseaux de Neurones Artificiels (RNA) et à leur utilisation dans le domaine du traitement d'images. Nous y rappelons les notions mathématiques derrière les RNA avec un rapide historique de leur développement, allant du perceptron à l'apprentissage profond qui est au cœur des réseaux exploités aujourd'hui. Nous expliquons le fonctionnement des *Convolutional Neural Networks* (CNN), qui sont une variante des RNA dédiée au traitement d'images. Enfin, la dernière partie du chapitre offre une vue d'ensemble des principales architectures de CNN utilisées dans la littérature pour la réalisation de segmentation automatiques.

Le troisième chapitre est dédié à une approche d'IA davantage orientée connaissance que les RNA, qui eux se basent exclusivement sur l'expérience lors de leur entraînement. Il s'agit du Raisonnement à Partir de Cas (RàPC). Les grands principes théoriques sur lesquels se repose le RàPC y sont présentés, accompagnés d'un résumé de son historique. Nous y décrivons en détails le fonctionnement général d'un système de RàPC avec les différents types de connaissances que l'on y trouve et les différentes étapes ou modules formant un tel système. La fin du chapitre se concentre sur l'exploration de la littérature concernant l'application du RàPC à la problématique plus spécifique de la segmentation d'images.

La deuxième partie de ce document est consacrée à nos contributions originales. Les premières contributions sont abordées dans le quatrième chapitre avec la conception et la modélisation de la plate-forme COLISEUM-3D. Les couches essentielles à son bon fonctionnement sont détaillées : la couche de données, la couche de segmentation et la couche de fusion. Certaines parties de la plate-forme ne sont pas l'objet de ces travaux et ne sont présentées que de manière succincte. Dans ce cas, nous nous limitons à expliquer leur fonctionnement théorique et à avancer des pistes de réflexions pour leurs implémentations futures. En particulier, nous nous attardons sur le module de recalage de COLISEUM-3D nécessaire à une future fusion des résultats.

Les 2 contributions majeures de ce travail de thèse sont réparties sur les 2 derniers chapitres. Le chapitre 5 est consacré à la méthodologie et à la formalisation mathématique de ces contributions. Tout d'abord, notre système de RàPC dédié à la segmentation du parenchyme rénal est présenté en détails. Nous abordons le fonctionnement de l'ensemble des modules du système. Une grande partie de la contribution à propos du RàPC concerne la conception d'une phase d'adaptation robuste pour adapter une solution existante et préalablement stockée à un nouveau problème. La seconde partie du chapitre 5 présente une méthode d'entraînement originale pour un RNA, que nous avons nommée *OV²ASSION* (*Overlearning Vector for Valid Sparse SegmentatIONS*), dédiée à la segmentation du néphroblastome.

Le chapitre 6 termine ce manuscrit avec la présentation des résultats de l'ensemble de nos expérimentations. Il débute par la description d'expériences liminaires nous permettant de justifier certains choix effectués dans ce travail de recherche. Puis nous mettons en évidence l'intérêt de la phase d'adaptation de notre système de RàPC et ses performances. Pour *OV²ASSION*, nous évaluons l'influence de ces différents paramètres sur la précision des segmentations de la tumeur et nous présentons ses performances sur plusieurs patients. La dernière section est une discussion générale sur les résultats et

sur les limites de nos expérimentations actuelles.

QUELQUES INDICATIONS POUR LA LECTURE DE CE RAPPORT

Le plan est organisé sur quatre niveaux : Parties, Chapitres, Sections et Sous-Sections. Il est utile de savoir que la Partie II, présentant les contributions et résultats, est indépendante de la première Partie, dédiée au contexte de la thèse et à l'état de l'art sur les différentes approches sélectionnées.

Chaque partie possède une introduction présentant son sujet, tout comme chaque chapitre. Une synthèse est disponible à la fin de chacun des chapitres afin de résumer son contenu et faciliter une lecture rapide du manuscrit.

Les résultats de nos travaux ont pour la plupart été publiés. La liste des publications personnelles est donnée avant la bibliographie placée en fin du document.

Par convenance personnelle, nous avons choisi d'utiliser la norme anglaise pour la représentation des nombres décimaux dans ce manuscrit.



CONTEXTE ET ÉTAT DE L'ART

Cette première partie du manuscrit de thèse nous permet tout d'abord d'introduire le travail dans son contexte, avec en particulier quelques notions médicales qui permettront au lecteur, béotien dans ce domaine, de mieux comprendre la finalité et les enjeux. Puis de manière plus classique nous présenterons notre état de l'art.

Le premier chapitre est consacré au contexte et à une présentation du projet *AsKid* (**A**utomatic **S**egmentation using artificial intelligence of **KID**ney tumours in children), dans lequel ce travail s'inscrit, avec les problématiques médicales et les contraintes associées. La physiologie du rein y est tout d'abord succinctement décrite afin d'en comprendre les enjeux médicaux importants pour les jeunes patients. Puis quelques notions liminaires en imagerie sont introduites concernant les images numériques et leur segmentation. Les concepts physiques sous-jacents à l'imagerie scanner (tomodensitométrie) sont ensuite présentés ainsi que le fonctionnement des appareillages les utilisant et les méthodes de visualisation des clichés produits. Pour clore ce chapitre, la définition des différents types d'apprentissage machine ainsi que leur application au domaine du traitement d'images est abordée, avec un tour d'horizon des états de l'art disponibles dans la littérature sur l'amélioration des techniques de segmentation automatique par l'usage de l'intelligence Artificielle (IA).

Le deuxième chapitre présente le fonctionnement des réseaux de neurones, à partir des notions de base comme le perceptron jusqu'aux réseaux profonds dédiés au traitement d'images, et le principe de l'apprentissage automatique permettant de les entraîner. Un état de l'art sur les principaux réseaux de neurones existants pour la segmentation d'images, et plus particulièrement appliqués à l'imagerie médicale, ainsi que leur architecture associée conclut ce chapitre.

Enfin, pour clore cette partie, le troisième chapitre expose le domaine du Raisonnement à Partir de Cas (*RàPC*) comme moyen d'ajouter de l'intelligence à des algorithmes de segmentation automatique. Ce chapitre aborde l'historique et la naissance du *RàPC* à partir du raisonnement par analogie afin d'en expliquer le fonctionnement et les principes. Enfin, un état de l'art sur la segmentation d'images utilisant cette technique d'Intelligence Artificielle est présenté, en se concentrant principalement sur deux travaux faisant référence dans le domaine des systèmes de *RàPC* dédiés à cette application.

LE CONTEXTE DE LA SEGMENTATION D'IMAGES

Les travaux de cette thèse s'inscrivent dans le cadre d'un projet nommé ASKid (**A**utomatic **S**egmentation using artificial intelligence of **KID**ney tumours in children) qui consiste à créer un système capable d'intelligence afin de segmenter automatiquement des images médicales sans intervention humaine ou, a minima, avec une intervention très limitée de la part d'un opérateur. Le projet s'intéresse aux segmentations des tumeurs rénales infantiles (plus particulièrement du néphroblastome, aussi appelé tumeur de Wilms). La segmentation est un procédé consistant à isoler un ou plusieurs objets d'intérêt sur une image. Dit autrement, cela consiste à associer à chaque pixel une étiquette ou classe par l'intermédiaire d'une carte de segmentation dans laquelle chacune de leur valeur est remplacée par cette étiquette.

Le dessein médical de ce projet d'envergure est de rendre systématique l'exploitation des informations apportées par l'imagerie médicale dans le cadre du traitement chirurgical des tumeurs du rein chez l'enfant. Ces images sont en effet un support capital pour orienter et planifier au mieux la chirurgie (et même plus généralement la thérapie) et décider si l'organe touché par la tumeur doit être sacrifié ou non : la résection de la tumeur entraînera-t-elle une néphrectomie partielle ou totale ?

La néphrectomie totale était jusqu'à récemment le geste recommandé pour le traitement du néphroblastome unilatéral chez un enfant ne présentant pas de facteur de risque particulier d'insuffisance rénale (recommandation de la Société International d'Oncologie Pédiatrique). Ces recommandations ont progressivement évolué en raison du bon pronostic oncologique de cette tumeur et de la volonté de réduire la morbidité du traitement chirurgical. Cette évolution rend l'exploitation de l'information image de plus en plus nécessaire dans l'intérêt du patient. Cependant, le temps important que représente le traitement des images les rend difficilement utilisables en routine.

Il est important, pour bien comprendre la finalité de ce travail, d'avoir un minimum de connaissances physiologiques et épidémiologique sur le rein et le néphroblastome. La section suivante permet de donner ces connaissances de base.

1.1/ LE CONTEXTE MÉDICAL : L'AIDE AU TRAITEMENT DE TUMEURS INFANTILES

Bien que la problématique du projet soit transposable à beaucoup d'applications différentes pour le traitement de tumeurs, le contexte médicale du projet *ASKID* se concentre sur le cas de jeunes enfants atteints pas une tumeur rénale infantile particulière nommée *néphroblastome*.

1.1.1/ LE NÉPHROBLASTOME : DÉFINITION, ÉPIDÉMIOLOGIE ET TRAITEMENT

Le néphroblastome (ou tumeur de Wilms) est une tumeur maligne provenant du blastème embryonnaire. C'est une tumeur très différente de celles que l'on trouve chez l'adulte : elle est spécifique à l'enfant. Le néphroblastome représente une urgence médicale à cause de sa rapide évolution. Heureusement, grâce au traitement approprié constitué à la fois d'une chimiothérapie, radiothérapie et d'un acte chirurgical, le taux de guérison est de 90% pour les formes communes [plantazTumWilms]. Ce fort taux de guérison est devenu possible grâce à une prise en charge spécifique et une connaissance approfondie de ce type de tumeur. Il cache néanmoins une forte disparité suivant son stade (100% pour des stades précoces et seulement 63% pour des stades à haut risque [siteWebGustRouss]). Le traitement comprend une chimiothérapie initiale, puis une résection de l'organe touché (néphrectomie en l'absence d'informations supplémentaires) en plus de la tumeur. Une chimiothérapie post-opératoire et une radiothérapie peuvent être pratiquées pour diminuer le risque de récurrence chez le patient et un suivi trimestriel de deux ans est réalisé afin de détecter l'apparition éventuelle de métastases [siteWebGustRouss].

Le diagnostic est suspecté devant certains signes cliniques tels qu'une hématurie ou une palpation d'une masse abdominale indurée. Il est ensuite corroboré sur les examens d'imagerie (le diagnostic final étant certifié sur l'analyse anatomopathologique de la pièce opératoire). Cette tumeur ne provoque que peu de symptômes, ce qui explique qu'elle soit souvent déjà très volumineuse lors du diagnostic.

La tumeur de Wilms constitue 5 à 10% des tumeurs malignes chez l'enfant, ce qui en fait l'une des tumeurs solides infantiles les plus fréquentes (maintenant légèrement dépassée par le neuroblastome). L'âge moyen des patients touchés est de 3 ans avec une fréquence maximale entre 1 et 5 ans. Il existe des facteurs de risque qui semblent augmenter la prévalence du néphroblastome, mais la plupart des patients sont dénués d'antécédents [plantazTumWilms].

1.1.2/ L'ANATOMIE ET LES FONCTIONS DU REIN

Les reins sont des organes essentiels du corps humain se trouvant en deux exemplaires et assurant des fonctions variées : épuration du sang, production de l'urine, régulation de la pression artérielle ainsi que des fonctions hormonales. Dans cette section, nous décrirons leur morphologie ainsi que leur organisation tissulaire. Les fonctions du rein étant intimement liées à son anatomie, nous traiterons les deux sujets de manière concomitante.

1.1.2.1/ LA MORPHOLOGIE ET L'ORGANISATION TISSULAIRE

Les reins sont des organes ovoïdes, situés en position rétro-péritonéale. La face externe est convexe et la face interne est concave. La surface du rein est lisse et de couleur rouge-brun avec une forte variabilité des mensurations selon les individus.

La Figure 1.1 présente un schéma de l'organisation d'un rein humain et de sa vascularisation. Cet organe est formé de différents tissus distinctement visibles en imagerie.

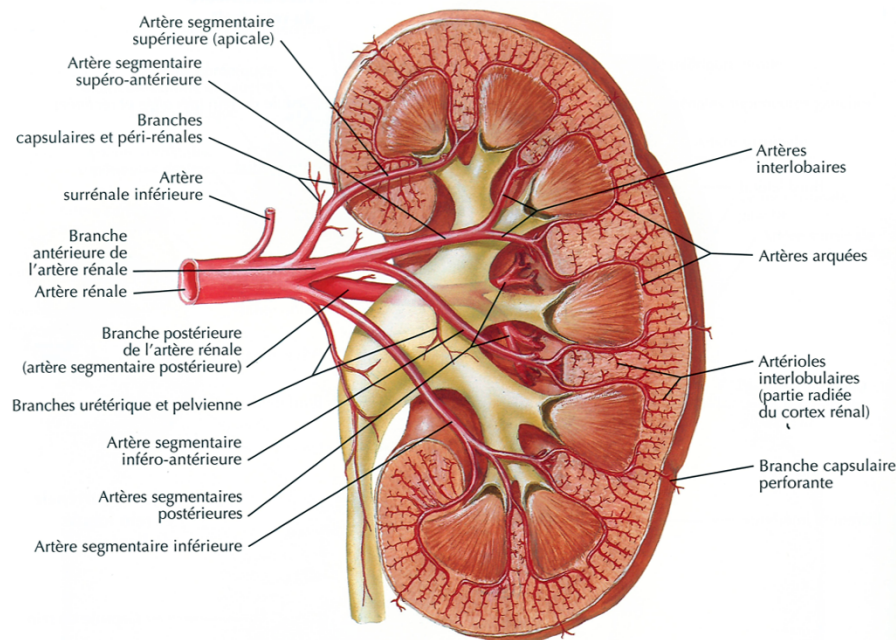


FIGURE 1.1 – Schéma du rein et de sa vascularisation [netter2014atlas]

Le rein peut être divisé en deux grandes zones tissulaires formant **le parenchyme rénal**. On y distingue tout d'abord en périphérie **le cortex rénal** qui abrite les unités fonctionnelles du rein, appelées *néphrons*, responsables entre autres de la production de l'urine ; puis **la médulla** avec des structures très reconnaissables appelées *pyramides de Malpighi* (ou plus simplement pyramides rénales), visibles sur la Figure 1.1, dont la pointe est orientée vers le centre de l'organe. Ces pyramides sont constituées de tubules déversant l'urine produite par les néphrons dans les différents calices, puis dans le pelvis rénal, l'ensemble correspondant aux **cavités rénales**. Ces pyramides sont séparées les unes des autres par des extensions du cortex appelées *colonnes de Bertin* (ou colonnes rénales) permettant de mieux ancrer ce dernier et étant le chemin d'accès pour les vaisseaux sanguins irriguant le cortex (vaisseaux interlobaires).

1.1.2.2/ L'ÉPURATION DU SANG ET LA PRODUCTION DE L'URINE

Cette fonction est principalement assurée par le néphron. Il prend sa source dans le cortex rénal au niveau du corpuscule, facilement observable en histologie, illustré par la Figure 1.2. On y trouve une partie vasculaire avec le *glomérule* et une partie épithéliale avec la *capsule de Bowman*. Le glomérule est une *pelote* de capillaires à haute pression encastrés dans la capsule. L'ensemble forme ainsi un filtre composé de l'*endothélium*

fenestré (fine paroi des capillaires avec des espaces facilitant les échanges) et des *podocytes*, des cellules épithéliales différenciées et ramifiées, enlaçant les capillaires du glomérule tout en laissant seulement de petits espaces pour le passage de molécules : les fentes de filtration. Ce filtre est capable de laisser passer de petites molécules tout en retenant les principaux constituants du sang et est à la base de sa fonction **d'épuration du plasma sanguin**. Le sang provient de l'artériole afférente et quitte le glomérule par l'artériole efférente. Le filtrage se réalise par un simple différentiel de pression. La pression intraglomérulaire est plus élevée que la pression dans la capsule. L'eau et les petites molécules traversent ainsi l'endothélium pour déboucher dans la lumière de la capsule de Bowman. L'ultra filtrat (ou **urine primitive**) court ensuite à travers une structure tubulaire constituant la majeure partie du néphron, comme présentée dans la Figure 1.3.

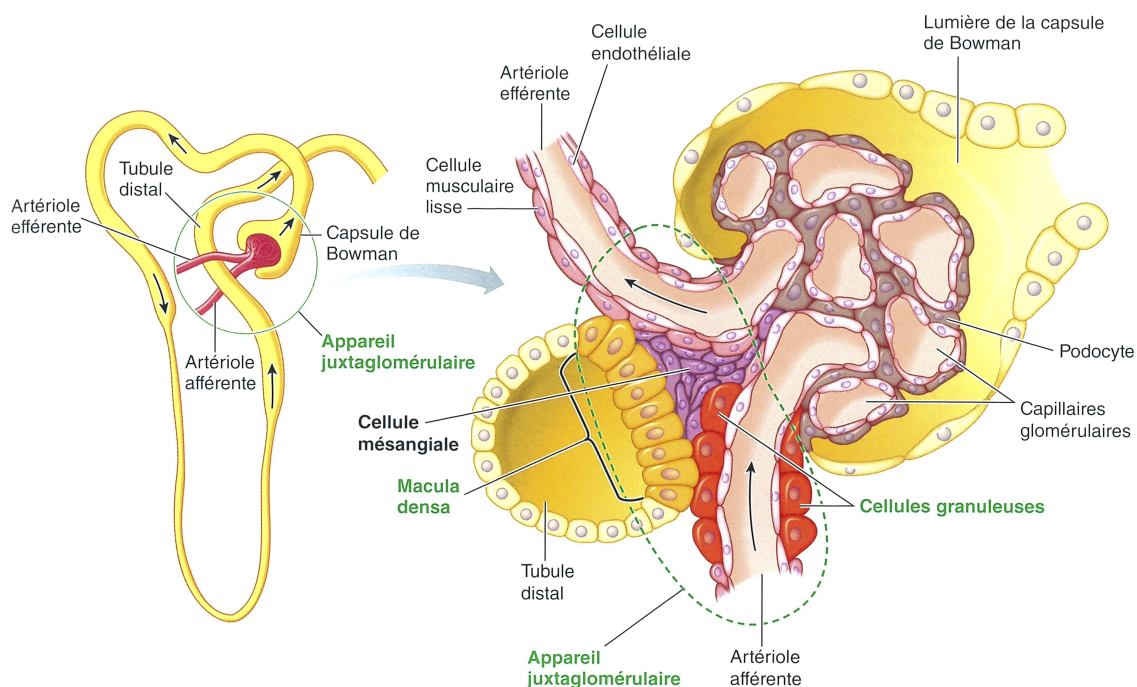


FIGURE 1.2 – Schéma du corpuscule rénal et de l'appareil juxta-glomérulaire [sherwood2016physiologie]

Cette structure tubulaire est divisée en trois parties : le *tube contourné proximal*, l'*anse de Henlé* et le *tube contourné distal*. Elle a pour rôle principal de concentrer l'urine par un jeu de réabsorption d'eau et d'ions **pour produire l'urine définitive** qui est ensuite déversée dans les tubes collecteurs et conduit jusqu'aux cavités rénales. Ces échanges avec le sang sont permis grâce à un gradient osmotique entre le cortex et la médulla, ainsi que par la présence d'un réseau capillaire qui accompagne l'anse de Henlé, issu de l'artériole efférente du glomérule : les *vasa recta*. Les tubes collecteurs possèdent eux-mêmes un rôle dans la régulation de l'élimination de l'eau en permettant sa réabsorption plus ou moins importante sous influence hormonale. **La vascularisation se concentre dans le cortex et, dans une moindre mesure, dans la médulla** via les *vasa recta*. Une fois l'urine déversée dans les cavités rénales (les différents calices et le pelvis rénal), elle suit les voies urinaires pour être stockée dans la vessie grâce aux uretères en attente de son expulsion dans le milieu extérieur.

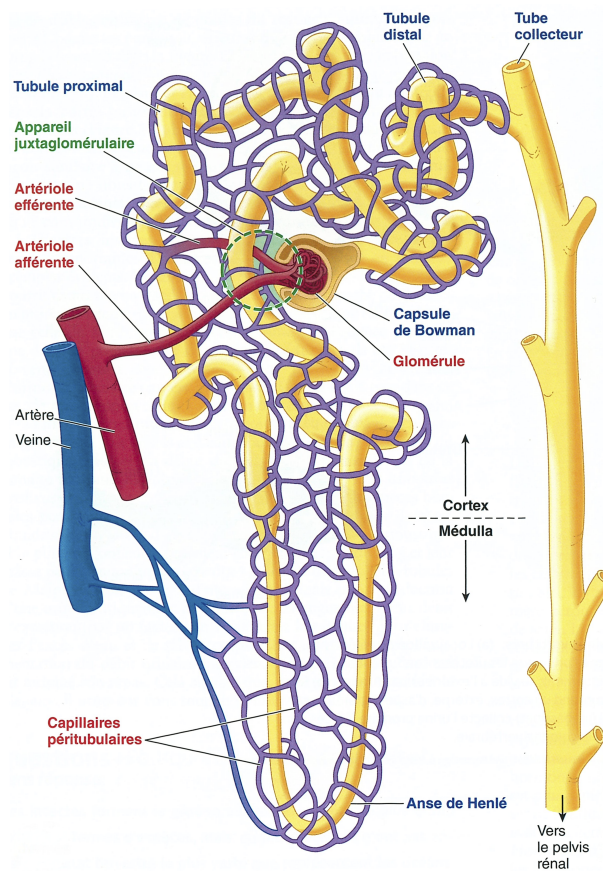


FIGURE 1.3 – Schéma anatomique d'un néphron [sherwood2016physiologie]

1.1.2.3/ LA RÉGULATION DE LA PRESSION SANGUINE ET DE LA VOLÉMIE

Le rein joue également un rôle central dans la régulation de la pression artérielle, par l'intermédiaire de la régulation de la *volémie*. La volémie correspond au volume de sang en circulation. Dans un système fermé, agir sur ce volume est l'un des nombreux moyens de modifier la pression du liquide circulant. De manière passive, plus la pression artérielle sera importante, plus le volume de liquide filtré au niveau du glomérule par unité de temps augmentera. Il s'agit du taux de filtration glomérulaire (TFG). De plus, une augmentation du TFG permet d'éliminer davantage de sodium dans les urines. Ce composant étant fortement osmotique, ceci favorise d'autant plus la perte d'eau. Ce mécanisme permet donc de diminuer une pression artérielle trop élevée. Cependant le rein possède également une activité réceptrice et hormonale qui influence la hausse de la pression sanguine. Un tissu en particulier est essentiel pour remplir cette fonction : l'appareil *juxta-glomérulaire*.

L'appareil juxta-glomérulaire

L'appareil juxta-glomérulaire est présenté dans la Figure 1.2. Il s'agit d'un ensemble de trois types de cellules accolé aux corpuscules rénaux. Chaque type cellulaire possède des fonctions propres :

- **Les cellules juxta-glomérulaires** : Également appelées cellules myo-

épithélioïdes ou encore cellules granulaires, elles sont situées dans la média des artères afférentes du glomérule. Elles possèdent des propriétés à la fois contractiles et endocrines. Elles ont également un rôle de baro-récepteurs car elle sont capables de détecter des changements de pressions dans la circulation sanguine. En réponse à une basse pression rénale, elles secrètent de la rénine pour permettre un rétablissement de la pression.

- **La Macula Densa** : Ces cellules forment la paroi du tube contourné distal du néphron. Elles se trouvent en contact direct à la fois avec l'urine produite circulant dans le tube et avec la lame basale des cellules juxta-glomérulaires. Elles ont un rôle d'osmo-récepteurs et sont capables de détecter la concentration de chlorure de sodium dans l'urine (NaCl). Suivant cette concentration, elles exercent une activité paracrine sur les cellules environnantes pour induire une vasoconstriction de l'artériole afférente et ainsi modifier le taux de filtration glomérulaire (TFG).
- **Les cellules mésingiales** : Également appelées les *cellules de Lacis*, leur rôle précis est encore inconnu. Elles se trouvent entre les capillaires du glomérule (cellules grises dans la Figure 1.2) et à l'extérieur de celui-ci, en contact avec l'ensemble des cellules de l'appareil juxta-glomérulaire. Elles ont une fonction structurelle en permettant l'ancrage des capillaires du glomérule.

L'action du rein sur la régulation de la pression se fait au sein du système *rénine-angiotensine-aldostérone*, s'appuyant sur les cellules de l'appareil juxta-glomérulaire.

Le système rénine-angiotensine-aldostérone

Ce système désigne une régulation hormonale en partie localisé dans le rein et dont le rôle est le maintien de l'*homéostasie hyposodée*. C'est en agissant sur cet équilibre que le rein peut influencer sur la volémie et la pression artérielle. Ce système est résumé dans la Figure 1.4.

Son point de départ est la synthèse de l'*angiotensinogène* par le foie et son relargage dans le sang. Le rein, par l'intermédiaire des cellules juxta-glomérulaires, sécrète de la rénine qui est une enzyme dont le rôle est de cliver l'angiotensinogène en une forme intermédiaire nommée *angiotensine I*. Cette molécule est finalement transformée en sa forme active grâce à l'action d'une enzyme de conversion sécrétée par les cellules endothéliales du poumon.

L'*angiotensine II* entraîne ensuite une cascade d'effets aboutissant à une augmentation de la pression. Cette hormone induit tout d'abord une vasoconstriction des artérioles dans l'organisme. D'un autre côté, elle agit également sur le rein en augmentant la réabsorption d'eau depuis les urines, et ce de deux manières différentes :

- Premièrement, elle augmente la réabsorption tubulaire de sodium, principalement au niveau du tube contourné proximal, et stimule la sécrétion d'aldostérone par les glandes surrénales avec le même effet. Si plus de sodium est réabsorbé, la rétention d'eau augmente par un phénomène osmotique. On parle de réabsorption passive.
- Deuxièmement, elle provoque la sécrétion de vasopressine (ou ADH pour *Antidiuretic Hormone*) par l'hypophyse. Cette autre hormone agit au niveau des tubes col-

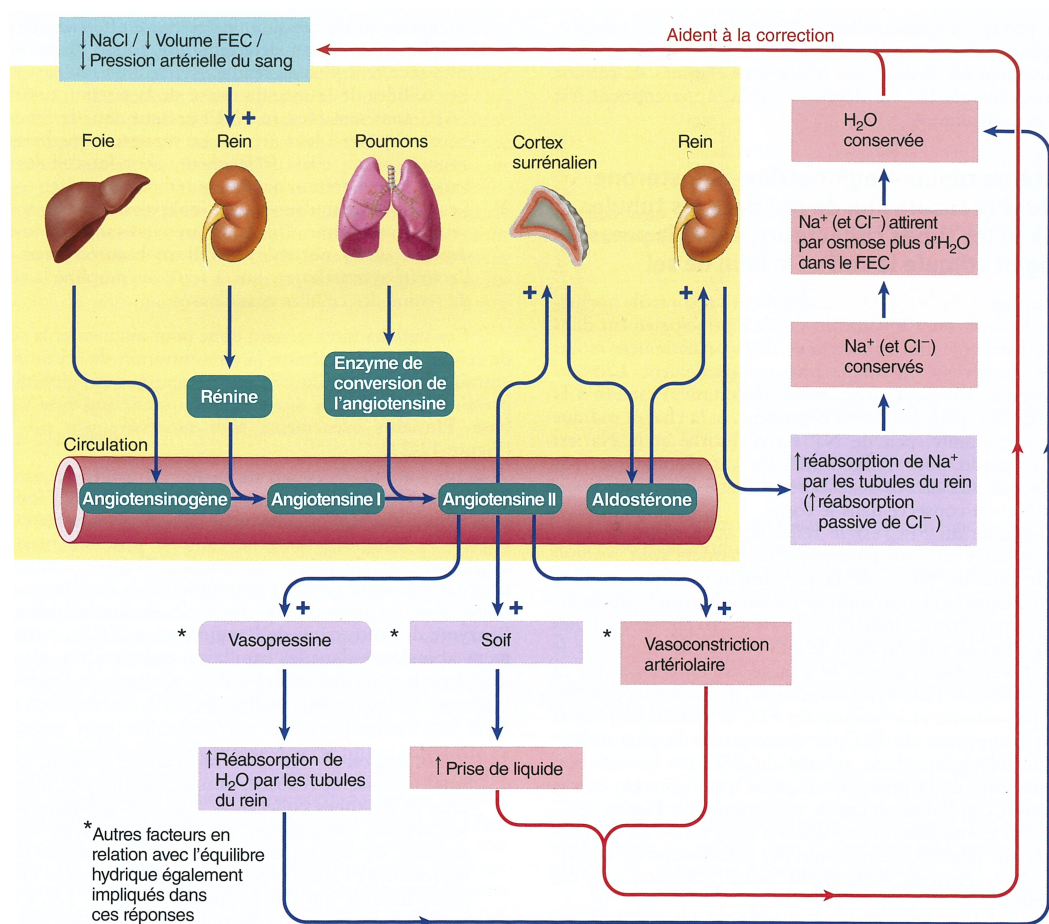


FIGURE 1.4 – Vue d'ensemble du système Rénine-Angiotensine-Aldostérone [sherwood2016physiologie]

lecteurs pour augmenter la réabsorption active d'eau via l'activation de la synthèse d'*aquaporine 2* (transporteur transmembranaire) et sa translocation au niveau de la membrane urinaire.

Le système est soumis à un rétro-contrôle négatif grâce à la rénine. Si sa sécrétion permet d'enclencher l'activation de l'angiotensinogène, elle agit en parallèle sur le foie en inhibant sa synthèse. Ainsi, lorsqu'il n'y a plus d'angiotensinogène, l'augmentation de la pression artérielle cesse. Un équilibre permanent entre rétro-contrôle négatif et action de l'angiotensine permet de conserver une pression physiologiquement optimale.

1.1.2.4/ LES AUTRES FONCTIONS ENDOCRINES

Parmi les fonctions endocrines du rein, nous retrouvons la *sécrétion d'érythropoïétine* (EPO) qui stimule l'*érythropoïèse*. Elle initie la différenciation et la prolifération des pré-curseurs érythrocytaires donnant les hématies du sang (globules rouges). Sa sécrétion par le cortex rénal est provoquée par une hypoxie au niveau des artères rénales. Le rein assume 90% de sa synthèse contre seulement 10% pour le foie.

Une autre fonction endocrine du rein entre en considération dans la régulation du méta-

bolisme calcique : la concentration plasmatique en calcium est soumise à un contrôle fin. Sous l'action de la *parathormone* (PTH) lorsque cette concentration devient trop faible, le rein augmente la réabsorption du calcium dans les urines pour compenser cette diminution. La PTH entraîne également la sécrétion par le rein du *calcitriol*, produit à partir de la vitamine D : une hormone qui augmente l'absorption intestinale de calcium. Le calcitriol inhibe en parallèle la production de PTH, assurant un rétro-contrôle négatif.

1.2/ L'IMAGERIE : QUELQUES NOTIONS LIMINAIRES

1.2.1/ LA REPRÉSENTATION NUMÉRIQUE D'UNE IMAGE

Un ordinateur ne peut pas traiter directement une image analogique. Il est nécessaire d'effectuer au préalable une numérisation pour que l'image devienne manipulable par un système informatique.

1.2.1.1/ SIGNAL NUMÉRIQUE VS SIGNAL ANALOGIQUE

Un signal numérique se distingue d'un signal analogique par l'absence de support physique. Le monde réel est continu, contrairement au monde informatique qui ne gère que des valeurs discrètes. Le passage d'un signal analogique à un signal numérique se fait par un échantillonnage (aussi appelé discrétisation). Cette étape consiste à mesurer la valeur d'un signal analogique donné de manière ponctuelle afin de le transformer en un ensemble fini de valeurs comme illustré dans la Figure 1.5. La qualité de l'échantillonnage est conditionnée par la fréquence d'échantillonnage. Plus cette fréquence est élevée, plus le signal reproduit numériquement est fidèle au signal d'origine. À l'inverse, une fréquence trop faible ne permet pas de reconstituer correctement le signal numérique, entraînant ainsi une perte définitive d'information lors du processus.

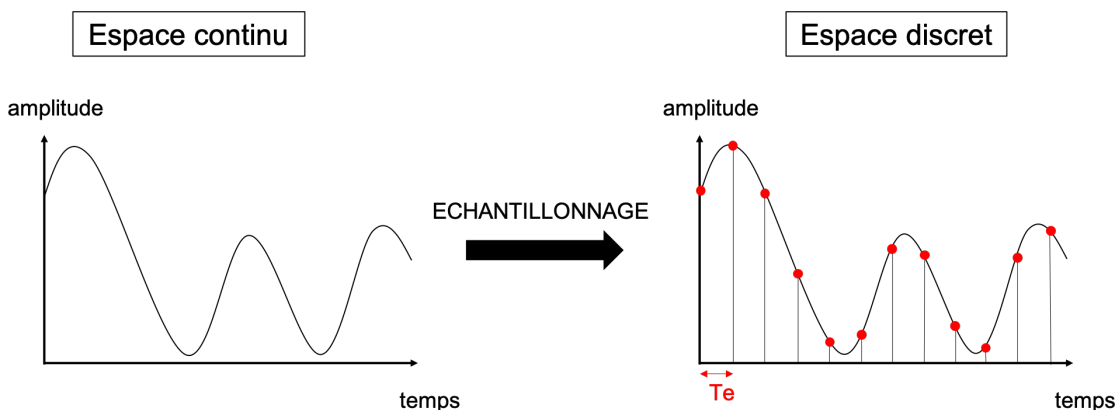


FIGURE 1.5 – Exemple d'échantillonnage d'un signal analogique

1.2.1.2/ L'IMAGE MATRICIELLE

Une image est couramment un signal en deux dimensions (éventuellement en trois, voire quatre pour le cas particulier d'un flux vidéo). L'échantillonnage est donc cette fois-ci

une discrétisation spatiale. L'unité d'une image numérique en 2 dimensions est le pixel. Chaque pixel possède une valeur entière. La taille et le nombre de pixels définissent la qualité de l'image. Plus les pixels sont nombreux et fins, plus la fréquence d'échantillonnage a été élevée et plus l'image numérique est proche de l'image analogique. Dans l'exemple donné par la Figure 1.6, il n'y a pas eu de perte d'information grâce à la simplicité de l'image d'origine (uniquement deux couleurs, blanc et gris, et une forme simple) et au nombre suffisant de pixels pour la représenter.

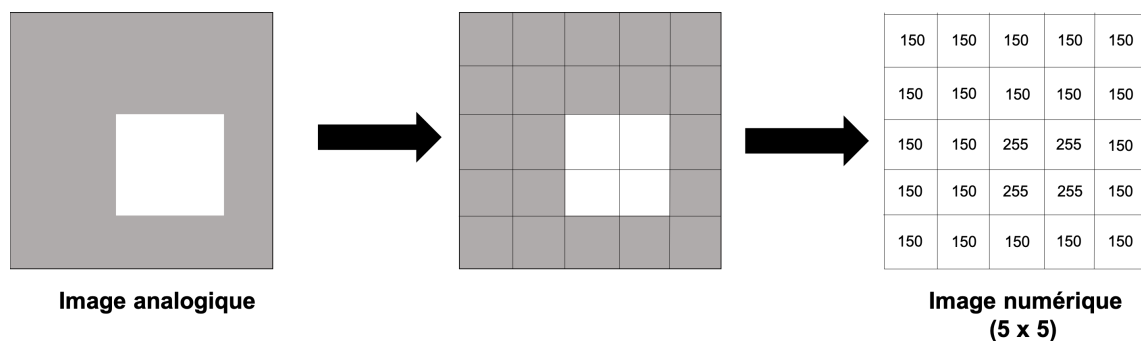


FIGURE 1.6 – Illustration de l'échantillonnage d'une image

La qualité de l'image numérique est quantifiée par sa résolution et sa définition.

Définitions

La résolution d'une image numérique représente la distance en dessous de laquelle deux points ne sont plus visualisables distinctement sur une image. Il s'agit d'une densité de pixels et elle s'exprime en dpi (*dots per inch*) ou en ppi (*pixels per inch*). Elle se calcule d'après le nombre de pixels de l'image et sa surface. Elle est donc directement liée à la taille des pixels. Elle ne doit pas être confondue avec la définition.

La définition d'une image correspond au nombre de pixels utilisés pour la représenter. Elle s'exprime généralement par le produit du nombre de pixels sur la hauteur de l'image par le nombre de pixels sur sa largeur (ex : 512 x 512 pour une image carrée).

Une image matricielle peut posséder plusieurs canaux, ce qui correspond au nombre de valeurs que possède chaque pixel. La représentation la plus simple est l'image **en niveaux de gris** dans laquelle chaque pixel a une seule et unique valeur (et un seul canal). Mais il est également très courant de travailler avec des images à trois canaux pour représenter les couleurs, par exemple dans un espace RVB (Rouge-Vert-Bleu), dans d'autres domaines que l'imagerie scanner ou pour ajouter une fausse coloration.

1.2.1.3/ L'ENCODAGE D'UNE IMAGE

Une image, comme toute information en informatique, est encodée sous la forme d'une série de 0 et de 1. Au sein d'un système binaire, un nombre est exprimé par un certain

nombre de bits et chaque bit peut prendre deux valeurs distinctes : soit 0, soit 1. Plus le nombre de bits est important, plus il est possible d'encoder de grands nombres. L'encodage d'une image correspond au nombre de bits utilisés pour encoder la valeur d'un pixel : 2^n valeurs possibles avec n bits utilisés pour l'encodage.

Les images en niveaux de gris sont communément encodées sur 8 bits. Ainsi les pixels peuvent prendre $2^8 = 256$ valeurs différentes dans l'intervalle d'entiers $[0, 255]$. En imagerie médicale, il est courant d'obtenir des images avec un encodage sur un nombre de bits beaucoup plus grand, notamment pour les images scanner dans lesquelles le nombre de valeurs possibles pour chaque pixel est de plusieurs milliers.

1.2.1.4/ LE FORMAT DICOM

Digital Imaging and Communications in Medicine, dont l'acronyme est *DICOM*, est une norme standard pour la gestion informatique des données issues des modalités (appareils d'acquisition) de l'imagerie médicale. Le but de ce format particulier, propre au milieu clinique et hospitalier, est de faciliter la communication et le transfert des images entre les différentes modalités et les différents hôpitaux. De plus, il offre l'avantage de contenir toutes les informations concernant l'acquisition de ces images (données du patient, paramètres d'acquisition, ...).

Un fichier DICOM est constitué d'une liste de champs caractérisant les images tels que la date et le temps d'acquisition, le nom du patient, son âge, le type de système d'acquisition (scanner, échographe,...) ou encore le modèle de l'appareil utilisé. La matrice de pixels constitue l'un de ces champs.

1.2.2/ DÉFINITION D'UN PROBLÈME DE SEGMENTATION D'IMAGES

La segmentation consiste à regrouper/partitionner les pixels d'une image en plusieurs classes ou régions en leur attribuant un label particulier. Le regroupement se fait de manière à ce que des pixels d'une même classe aient des caractéristiques en commun. Ces caractéristiques peuvent être très variées suivant l'approche utilisée : il peut s'agir d'un critère de texture, de niveaux de gris ou encore de distance. Dit autrement, le but est de regrouper les pixels en régions suivant un critère de similarité.

Définition

Une segmentation consiste à partitionner une image en plusieurs régions R_i telles que l'union de ces régions forme l'image segmentée :

$$\bigcup_{i=1}^K R_i = \Omega \quad \text{avec} \quad \Omega \text{ l'image segmentée, } R_i \text{ région } i \text{ de la segmentation}$$

La Figure 1.7 illustre une segmentation d'un cliché scanner (CT-scan). L'image de gauche est une image scanner en niveau de gris de l'abdomen d'un patient en coupe transversale. Le processus de segmentation consiste à délimiter les zones dans le cliché correspondant aux structures d'intérêt. Il s'agit ici de segmenter le parenchyme rénal (en

marron), les cavités rénales (en jaune), les vaisseaux sanguins (rouge et bleu) et la tumeur (en vert).

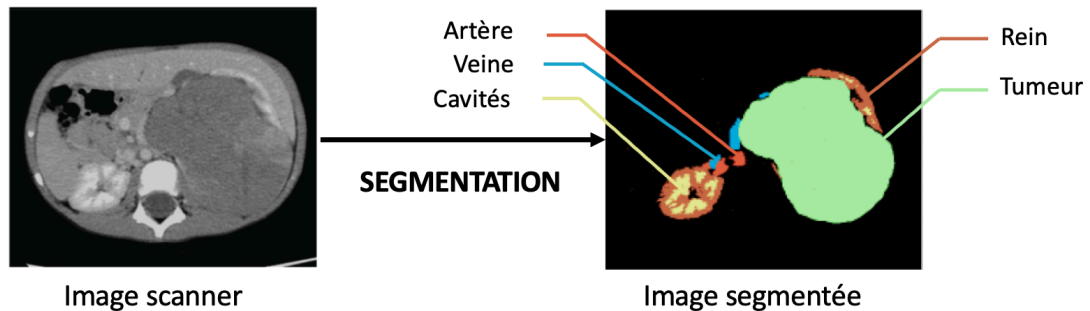


FIGURE 1.7 – Exemple d'une segmentation d'images sur une image scanner

La segmentation est généralement utilisée afin d'extraire un ou plusieurs objets d'une image. Le but peut être d'en extraire *a posteriori* des caractéristiques comme la surface, la localisation d'un objet ou encore d'effectuer une modélisation 3D (dans le cas où la segmentation a été faite en 3D ou sur une succession de coupes 2D).

1.2.3/ L'ÉVALUATION D'UNE SEGMENTATION

A partir du moment où un système produit un résultat, il est nécessaire d'être en mesure d'évaluer ce résultat. Il existe plusieurs scores pouvant être calculés pour définir la précision et la pertinence d'une segmentation. Les plus communs sont le Dice et l'IoU.

Le score Dice est communément utilisé par les experts en imagerie médicale et l'IoU est un coefficient courant dans la communauté du traitement d'images. Ils donnent tous deux une valeur de similarité (sur [0, 1]) entre 2 ensembles X et Y . Dans le contexte de l'évaluation d'une segmentation, X représente les pixels de la vérité terrain (la segmentation désirée) et Y représente les pixels de la segmentation calculée donné par le système.

Définition

Soit deux ensembles X et Y , le coefficient Dice, aussi connu comme le coefficient de Sorensen, consiste à calculer le taux de recouvrement entre ces deux ensembles selon la formule suivante :

$$Dice = \frac{2 * |X \cap Y|}{|X| + |Y|}$$

L'indice IoU est un coefficient plus pénalisant que le Dice, notamment en ce qui concerne les erreurs de segmentation (attribution d'une mauvaise étiquette à un pixel).

Définition

Soit deux ensembles X et Y , l'indice IoU (*Intersection over Union*), aussi appelé indice de Jaccard, est la moyenne des indices locaux IoU_i de toutes les régions i de X et Y . L'indice IoU_i est donné par la formule suivante :

$$IoU_i = \frac{|X \cap Y|}{|X \cup Y|} = \frac{n_{ii}}{n_{ii} + n_{ji} + n_{ij}}$$

- n_{ii} est le nombre de pixels correctement placés dans la région i ,
- n_{ji} est le nombre de pixels qui devraient être placés dans la région i mais qui ne le sont pas,
- n_{ij} est le nombre de pixels qui se trouvent dans la région i mais qui ne devraient pas y être.

1.3/ L'IMAGERIE ET LA SEGMENTATION DES STRUCTURES ABDOMINALES

La pathologie tumorale (présence de tumeur) du rein est tout d'abord confirmée par échographie et permet d'obtenir des caractéristiques générales sur le diamètre et/ou la position de la tumeur. La visualisation précise des structures d'intérêt est permise grâce à un examen tomодensitométrique, couramment appelé scanner (ou encore *CT-scan*), couplé à l'utilisation d'un produit de contraste (produit iodé hydrosoluble). Cette section nous permet de définir le contexte d'imagerie médicale dans lequel ces travaux de thèse ont été effectués.

1.3.1/ L'IMAGERIE TOMODENSITOMÉTRIQUE

L'imagerie tomодensitométrique se base sur la production et l'utilisation de rayons X pour visualiser, grâce aux interactions de ces rayons avec la matière, ce qui ne peut pas l'être directement.

1.3.1.1/ L'ATTÉNUATION DES PHOTONS X EN IMAGERIE

Le phénomène physique qui est exploité en imagerie médicale est l'absorption partielle par les tissus biologiques des rayons X, ce qui provoque leur atténuation. Cette atténuation résulte de deux phénomènes, à savoir l'effet photo-électrique et l'effet Compton. Soit l'exemple d'un rayon X incident Φ_0 traversant un matériau et Φ le rayon X résultant tel qu'illustré dans la Figure 1.8. Φ correspond à un faisceau atténué par la matière traversée. L'importance de l'atténuation dépend à la fois de l'épaisseur des tissus x et de leur nature qui est liée à un coefficient d'atténuation linéaire μ .

Ce coefficient dépend lui-même du numéro atomique Z du tissu et de sa densité ρ , ainsi que de l'énergie des photons X incidents Φ_0 .

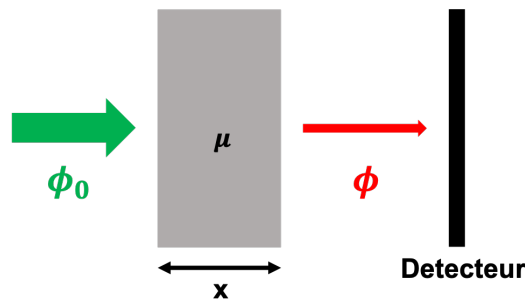


FIGURE 1.8 – Exemple d'une atténuation d'un rayon X

Définition

Soit un rayon X incident Φ_0 , traversant un matériau d'une épaisseur x avec un coefficient d'atténuation linéaire μ . Alors le rayon X résultant Φ a pour expression :

$$\Phi = \Phi_0 \cdot e^{-\mu \cdot x}$$

Dans une situation réelle, la matière traversée n'est souvent pas homogène et est constituée de différents tissus avec des coefficients d'atténuation propres. Dans ce cas, le coefficient global est la somme des différents coefficients.

Définition

Soit un rayon incident Φ_0 traversant un ensemble de n tissus consécutifs. Alors, le rayon résultant a pour expression :

$$\Phi = \Phi_0 \cdot e^{-\sum_1^n \mu_i \cdot x_i}$$

1.3.1.2/ LE FONCTIONNEMENT D'UN SCANNER ET SON PARAMÉTRAGE

Un scanner est une modalité d'imagerie médicale permettant de réaliser des clichés en coupe (généralement transversale) d'un patient en utilisant le phénomène d'atténuation par les tissus biologiques des rayons X. Le principe est qu'un ensemble formé d'un tube à rayons X et d'un détecteur tourne autour du patient comme illustré dans la Figure 1.9 afin d'effectuer un grand nombre de mesures. Ces mesures permettront à l'ordinateur de calculer une image. Un collimateur en sortie du tube à rayons X permet d'obtenir un faisceau de photons.

Il existe de nombreux paramètres influençant la qualité de l'examen et la qualité de l'image produite.

La résolution spatiale est contrôlée par l'épaisseur des coupes, la taille de la matrice de pixels et par le champ de reconstruction. Lors de l'examen, la planche sur laquelle repose

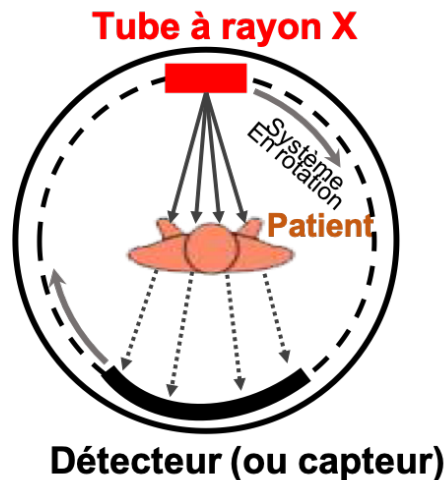


FIGURE 1.9 – Schéma du fonctionnement d'un scanner

le patient avance régulièrement. Le pas de la planche entre chaque rotation complète déterminera l'**épaisseur des coupes** calculées. Plus le pas est élevé, plus l'épaisseur l'est également et moins il y a de précision sur l'axe z (axe de la longueur du scanner). Pour les scanners les plus modernes, l'acquisition volumique a remplacé l'acquisition séquentielle. La translation de la planche est continue et le mouvement du tube/détecteur est hélicoïdal avec un temps d'acquisition devenu très court.

La résolution en contraste de l'image est principalement influencée par le **différentiel de potentiel** appliqué dans le tube à rayons X pour accélérer les électrons. Plus la différence de tension est importante entre les deux bornes (cathode et anode), plus les électrons accumulent de l'énergie cinétique et plus les photons X produits sont énergétiques. Pour un différentiel très faible (< 30 kV), l'effet photo-électrique est dominant. Les photons X sont tous absorbés par les tissus du patient et peu sont transmis au détecteur. Malgré une irradiation du patient, il n'y a donc pas d'image interprétable. Si au contraire le différentiel est très élevé (> 150 kV), il y a une très faible atténuation uniquement par l'effet dit de *Compton* et l'image est non interprétable car sans contraste. Le patient serait ainsi fortement irradié sans résultat. Il est donc nécessaire d'adapter la tension en fonction des tissus d'intérêt et du contraste nécessaire pour visualiser ces tissus.

Le nombre d'images pour un examen varie beaucoup, de 60 à 200 coupes, suivant son objectif et la zone de l'anatomie à explorer (examen corps entier, examen de l'abdomen seulement, ...). Il dépend également de l'épaisseur des coupes. Un examen plus fin contiendra plus d'images pour une même zone.

1.3.1.3/ LA VISUALISATION DES IMAGES

La visualisation des images ne se fait pas directement à partir des coefficients d'atténuation (\approx densités) calculés mais en utilisant l'échelle des unités Hounsfield (UH). Il s'agit d'une transformation linéaire des coefficients d'atténuation par rapport à la densité de l'eau distillée et la densité de l'air. Ceci revient à effectuer une normalisation dans laquelle ces deux densités sont respectivement fixées à 0 UH et -1000 UH.

Définition

Soit μ_x , le coefficient d'atténuation associé au matériau x , et μ_{eau} , le coefficient d'atténuation de l'eau, l'unité Hounsfield de ce même matériau a pour expression :

$$UH = 1000 \cdot \frac{\mu_x - \mu_{eau}}{\mu_{eau}}$$

Compte tenu du grand nombre de valeurs que peut prendre un pixel dans une image, exprimées en UH, l'encodage ne peut pas se faire sur seulement 8 bits (qui ne permettent que 256 valeurs différentes) : 12 bits (pour 4096 valeurs) sont généralement utilisés.

Le fenêtrage

L'œil humain n'est capable de distinguer qu'un nombre limité de gris différents. Il y a donc un surplus d'information dans les images en UH qui n'est pas visuellement interprétable. Le rôle du fenêtrage est d'ajuster l'échelle des niveaux de gris en fonction des structures d'intérêt que l'utilisateur souhaite visualiser. Il consiste à définir une « zone utile » par un intervalle de valeurs. Les pixels ayant une valeur au-dessus de la borne supérieure de cet intervalle reçoivent la valeur maximum. Ceux ayant une valeur en-dessous de la borne inférieure reçoivent la valeur minimum. Ceci s'apparente à un double seuillage et permet d'exploiter l'ensemble des niveaux de gris pour faciliter la distinction des structures dans la zone utile. Un fenêtrage est caractérisé par un centre et une largeur, tel qu'illustré par la Figure 1.10.

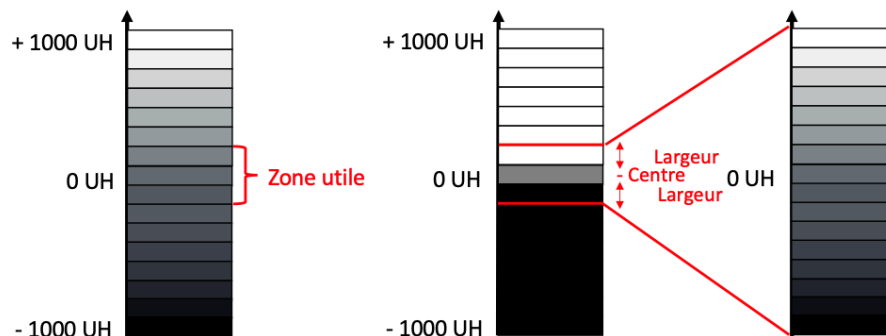


FIGURE 1.10 – Illustration de l'intérêt d'un fenêtrage

1.3.2/ LA MISE EN ÉVIDENCE DES STRUCTURES ABDOMINALES D'INTÉRÊT

Les structures que l'on cherche à déterminer sont les suivantes : le parenchyme rénal, le néphroblastome, les cavités et les vaisseaux sanguins (artère et veine rénales). Suivant la localisation du produit de contraste dans les tissus, il est donc possible d'améliorer la visualisation des structures hôtes. Pour ce faire, l'examen se fait en deux temps tels qu'illustrés dans la Figure 1.11 : **le temps vasculaire** et **le temps tardif**. Tout d'abord durant **le temps vasculaire** (image de gauche), le produit de contraste se limite **au système**

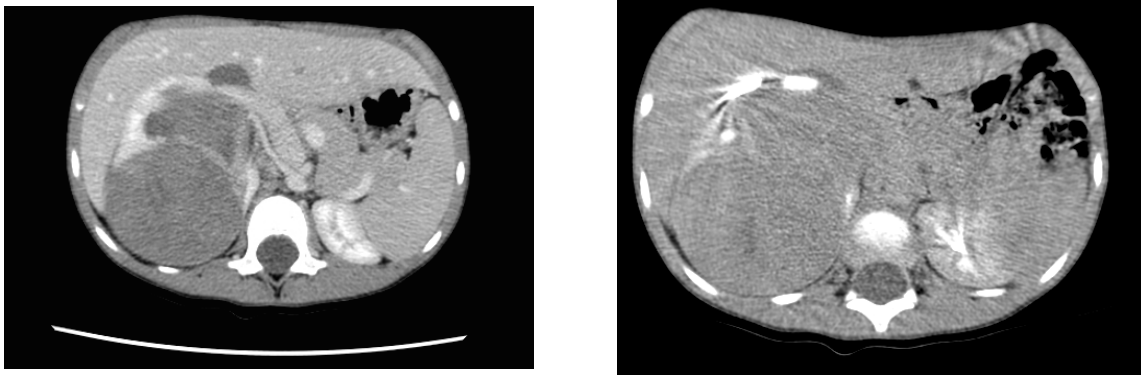


FIGURE 1.11 – Images scanner en coupe d'un rein et du néphroblastome associées en temps vasculaire (à gauche) et en temps tardif (à droite)

sanguin. Les structures fortement vascularisées, telles que le cortex rénal, les veines, les artères et dans une moindre mesure la tumeur, sont mieux discernables et apparaissent avec des intensités de pixels remarquables. **En temps tardif** (image de droite), plusieurs minutes après l'injection, le produit de contraste a été filtré par les reins et se retrouve en forte concentration dans **les voies urinaires**. Ainsi, les cavités rénales apparaissent avec une forte intensité sur les clichés scanner, permettant leur segmentation a posteriori. Les différences anatomiques entre les deux images s'expliquent par le laps de temps entre l'acquisition des deux clichés, temps durant lequel malgré les consignes données par le personnel soignant, il est difficile pour le jeune patient de ne pas bouger. Ainsi la correspondance entre les coupes n'est pas parfaite.

1.3.3/ L'INTÉRÊT MÉDICAL DE LA SEGMENTATION

Il existe deux grandes approches chirurgicales concernant le traitement d'un néphroblastome : **la néphrectomie partielle** (consistant à ne retirer que la tumeur tout en laissant au mieux les parties saines du rein touché ; on parle dans ce cas de chirurgie conservatrice) et **la néphrectomie totale** (la tumeur ainsi que la totalité du rein touché sont enlevés). Le choix de l'acte se base sur les recommandations de la Société Internationale d'Oncologie Pédiatrique (SIOP) qui a établi un certain nombre de critères à vérifier tels que la taille de la tumeur ou encore sa localisation. Parmi ces critères se trouve celui de l'envahissement tumoral des voies urinaires. Si la tumeur est circonscrite au cortex, alors une chirurgie conservatrice est envisageable pour sauver une partie du rein touché. Au contraire, si un envahissement des cavités urinaires est constaté, alors une exérèse totale s'impose pour éviter tous risques de récurrence.

Cette volonté de tenter une sauvegarde du rein touché ne s'appliquait, jusqu'à récemment, qu'au cas d'une atteinte bilatérale (ou d'une atteinte unilatérale sur rein unique). Cependant, il a été constaté d'une part que le taux de guérison est proche de 90% en cas de néphrectomie partielle respectant les critères décrits ci-dessus, et d'autre part qu'il y a un risque prépondérant de développer une insuffisance rénale pour les patients traités par néphrectomie totale. Cette insuffisance rénale possède plusieurs causes. La première est qu'un corps humain possède à la naissance un capital néphronique fixe (nombre de néphrons) qui est utilisé durant toute sa vie ; une néphrectomie totale induit donc une perte sèche de 50% de ce capital. Ensuite, la chimiothérapie pré-chirurgicale

a une néphrotoxicité sur le rein restant, couplée au fait que la charge à supporter par ce rein est doublée, durant toute une vie car les patients sont très jeunes. Ainsi, les recommandations de la SIOP ont évolué pour privilégier une sauvegarde du rein dans tous les cas possibles, y compris lors d'une atteinte unilatérale.

Cette décision n'est pas anodine car même s'il est possible de vivre avec un seul et unique rein, cela rend le patient plus vulnérable à de potentiels futurs dysfonctionnements ou pathologies du rein restant. De plus, dans le cas d'une pathologie bilatérale (si les deux reins sont touchés), il est primordial de savoir si l'un des reins peut être sauvegardé afin de ne pas condamner hâtivement le patient à vivre sous dialyse jusqu'à une éventuelle greffe d'organe.

L'une des façons de pouvoir évaluer l'envahissement des cavités rénales par la tumeur est de recourir à de l'imagerie médicale et d'effectuer une segmentation, suivie d'une modélisation en trois dimensions du rein pathologique comme illustré par la Figure 1.12. Cette modélisation permet, en modifiant la transparence des structures, de mieux visualiser cet envahissement tumoral.

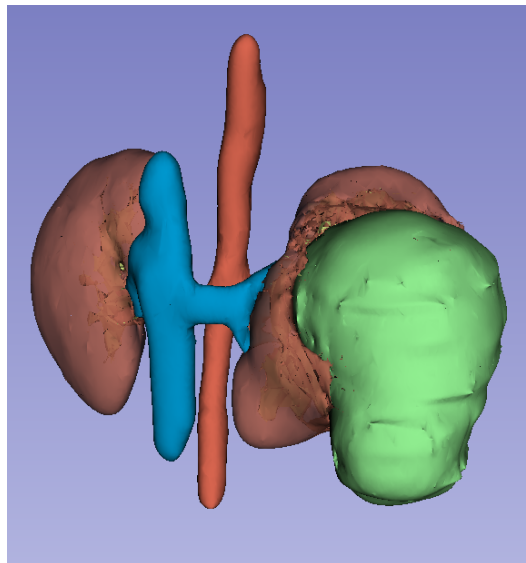


FIGURE 1.12 – Résultat d'une modélisation 3D (avec le logiciel 3D-Slicer) obtenue à partir des segmentations successives des images scanner (des différentes coupes) : en marron les reins, en vert la tumeur, en rouge l'aorte et les artères rénales et en bleu la veine cave inférieure et les veines rénales

1.3.4/ LOGICIELS ET MÉTHODES DE SEGMENTATION D'IMAGES UTILISÉS EN ROUTINE MÉDICALE

Les segmentations sont encore souvent réalisées manuellement. Certains algorithmes automatiques, guidés par un praticien ou un ingénieur biomédical, sont également utilisés au sein d'une solution logicielle. Parmi ces solutions, ITK Snap [siteWebitk] ou encore 3D-Slicer [siteWebslicer] peuvent être cités. Les algorithmes les plus souvent implémentés sont des algorithmes simples comme le seuillage ou encore la croissance de régions. Cependant, ce sont souvent des logiciels peu ergonomiques et complexes. De plus, ces logiciels et algorithmes nécessitent une certaine expérience de la part de l'utilisateur pour

être correctement utilisés. Enfin, malgré leur aide, la réalisation des segmentations reste chronophage pour le personnel hospitalier.

1.4/ L'APPRENTISSAGE MACHINE DANS LE DOMAINE DE LA SEGMENTATION

L'apprentissage machine, ou apprentissage automatique, dans ce domaine est la résolution par un programme d'une tâche de classification d'un ensemble de données grâce à un processus d'apprentissage permettant de déterminer des règles à appliquer. Dit autrement, il s'agit de produire un modèle prédictif qui apprend sur des données existantes afin de prévoir des tendances ou des comportements futurs. Il s'agit d'une définition quelque peu restrictive mais correspondant au contexte d'une segmentation d'image. La segmentation n'est en fait rien de plus qu'un problème de classification de données. L'objectif est d'attribuer une étiquette à chaque pixel composant l'image suivant l'objet qu'il représente (ou plutôt l'objet auquel il appartient). Le résultat est une simple visualisation des étiquettes, qui ont été attribuées à chacun des pixels, que l'on nomme familièrement « *segmentation* ».

1.4.1/ LA REPRÉSENTATION DE DONNÉES IMAGES

Les données utilisées sont un peu particulières puisqu'il s'agit d'images et même plus précisément de pixels, mais elles peuvent être représentées dans un espace à n dimensions sous la forme d'un nuage de points comme n'importe quel type de données. L'espace choisi pour représenter les données a son importance et influencera fortement l'efficacité de la classification. Chaque dimension de cet espace correspond à une information supplémentaire qui sera utilisée pour effectuer le partitionnement. Par exemple, une représentation assez naturelle des pixels dans une image en couleur est un espace à trois dimensions dans lequel les axes correspondent aux trois composantes Rouge-Vert-Bleu (*RVB*). Il est ensuite possible d'augmenter la dimension de cet espace en rajoutant des informations comme la position spatiale de chaque pixel. Pour une image en niveau de gris, une représentation simple est d'utiliser un axe pour l'intensité du pixel, un deuxième pour la coordonnée x et un dernier pour la coordonnée y (voir Figure 1.13). Plus de détails sur la manière de représenter les pixels d'une image, notamment dans un espace de couleur, sont disponibles dans le manuscrit de thèse de Cyril Meurie portant sur la segmentation d'images par classification pixellaire [meurie2005segmentation].

1.4.2/ LES DIFFÉRENTS TYPES D'APPRENTISSAGE AUTOMATIQUE

On distingue plusieurs types d'apprentissage (Figure 1.14) en fonction des approches utilisées, dont les deux principaux sont l'apprentissage non supervisé et l'apprentissage supervisé. À ceux-ci se rajoutent les apprentissages intermédiaires (semi-supervisés) et l'apprentissage par renforcement.

On trouve des applications au traitement d'images pour chacune de ces catégories et la suite de ce chapitre explicite leurs différences. Cependant, dans la littérature la grande

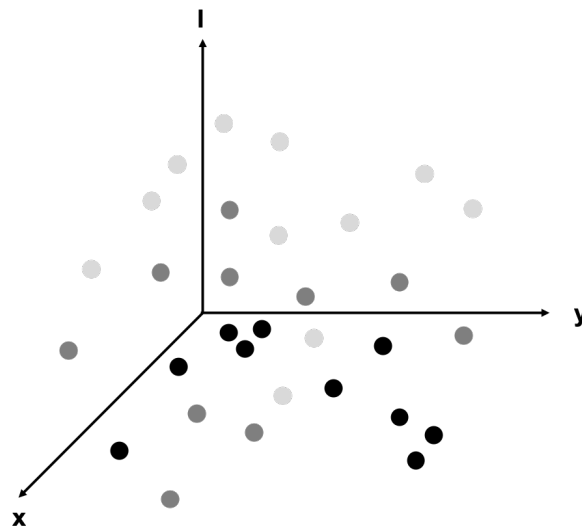


FIGURE 1.13 – Exemple d'une représentation possible des pixels d'une image 2D en niveaux de gris dans un espace à trois dimensions. L'axe vertical est la valeur du pixel et les deux autres axes sont ses coordonnées spatiales dans l'image

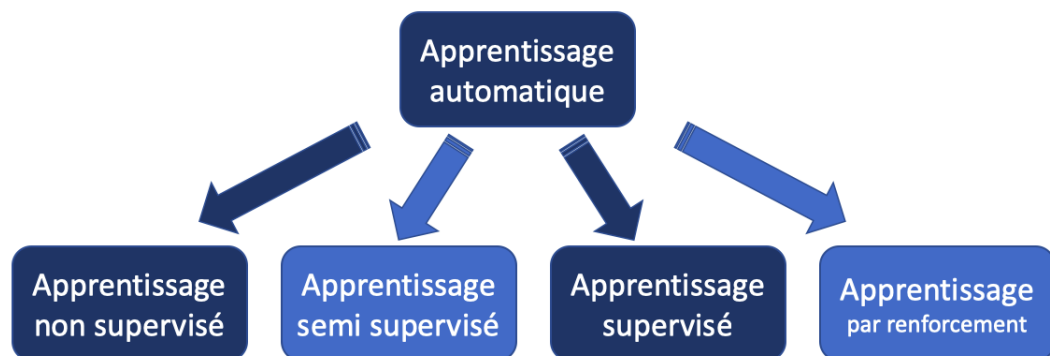


FIGURE 1.14 – Les différentes catégories d'apprentissage automatique

majorité des méthodes d'intelligence artificielle utilisées est rattachée soit à l'apprentissage supervisé, soit à l'apprentissage non supervisé.

1.4.2.1/ L'APPRENTISSAGE NON SUPERVISÉ

L'apprentissage non supervisé ne repose théoriquement sur aucune intervention humaine. L'ordinateur est capable de lui-même d'analyser des données brutes (sans information complémentaire comme un étiquetage), d'en déterminer la structure et d'inférer des connaissances qu'il pourra ensuite réutiliser. Les avantages de cette approche sont nombreux. Le premier est bien entendu l'absence de recours à un humain pour étiqueter les données, ce qui peut être très coûteux en temps pour traiter un nombre de données suffisant. De plus, cet étiquetage peut nécessiter une certaine compétence suivant le contexte et oblige donc le recours à un expert du domaine pour sa réalisation. Le deuxième grand avantage est qu'un ordinateur peut potentiellement trouver de l'information inaccessible à un humain (une structure cachée par exemple) alors que l'étiquetage

rend ceci impossible puisqu'il conditionne le processus.

Chez l'humain, il s'agit d'un type d'apprentissage naturellement utilisé. Par exemple, un enfant regardant une série de photographies d'animaux n'a pas besoin d'aide pour reconnaître un même animal sur plusieurs photos et le distinguer des autres, même s'il n'en connaîtra pas le nom sans l'indication d'un adulte. Bien que naturel chez l'être humain, ce type d'analyse est terriblement compliquée à reproduire chez un ordinateur. Si bien que les méthodes d'apprentissage non supervisées existantes se basent tout de même, à un moment ou à un autre, sur une intervention humaine (limitée mais présente). Ils constituent donc la plupart du temps seulement une aide et non pas un programme autonome capable de réaliser une tâche complète. *K-means* est l'une des méthodes les plus historiquement utilisées dans le cadre de la segmentation d'images ([ng2006kmeans], [ray1999colorSegmentation], [chen1998kmeans]) mais a depuis perdu en popularité. Traditionnellement à base d'apprentissage supervisé, certains réseaux de neurones peuvent également être modifiés afin d'apprendre à effectuer des segmentations d'images de manière non supervisées ([xia2017wnet]).

1.4.2.2/ L'APPRENTISSAGE SUPERVISÉ

L'apprentissage supervisé s'oppose au précédent et concerne un processus guidé par la connaissance humaine. Un expert créera un ensemble de données auxquelles il associera une étiquette afin de créer des couples (observation, label). Ces données serviront d'exemple pour permettre à l'ordinateur d'apprendre sur celles-ci afin d'en comprendre la structure et de construire un modèle de prédiction.

D'une manière plus formelle, la base d'apprentissage est un ensemble de couples entrées-sorties ou encore observations-prédictions $(x_n, y_n)_{1 \leq n \leq N}$ avec $x_n \in X$ et $y_n \in Y$ où X et Y sont des lois de tirages aléatoires inconnues. On distingue plusieurs cas selon le type d'ensemble auquel appartient y_n :

- $Y \subset \mathbb{R}$: la valeur (ou l'estimation) de sortie appartient à l'ensemble continu des réels. On parle alors de problème de régression et la fonction de prédiction est appelée un régresseur.
- $Y = \{1, \dots, K\}$: la valeur de sortie appartient à un ensemble fini. La prédiction revient donc à attribuer en sortie une étiquette à la valeur d'entrée. Dans ce cas on parle d'un problème de classification et la fonction de prédiction est appelée classifieur.

Les méthodes les plus connues dans la catégorie de l'apprentissage supervisé sont sans conteste les réseaux de neurones et l'apprentissage profond (*Deep Learning*) qui représentent un important domaine d'étude. En traitement d'images, ce sont les réseaux de neurones convolutifs (*Convolutionnal Neurone Network* ou CNN), une variante des réseaux classiques, qui sont massivement utilisés afin d'extraire des caractéristiques dans ces images susceptibles d'aider au calcul de la segmentation ([litjens2017Survey], [zhao2017survey]). La convolution est une opération mathématique permettant d'extraire des informations particulières d'un signal. Elle est à la base de la notion de filtrage linéaire. Ces réseaux particuliers consistent à appliquer en cascade des convolutions sur l'image en entrée pour produire une information plus abstraite, menant à une classification ou à une segmentation.

Cependant, on trouve également d'autres méthodes telles que les *Support Vector Machine* ([ricci2007svm], [zhou2006svm]), aussi appelés *Séparateurs à Vaste Marge*, ou les *k plus proches voisins* ([vrooman2007knearest]).

1.4.2.3/ L'APPRENTISSAGE SEMI-SUPERVISÉ

L'apprentissage supervisé se montre certes efficace pour un bon nombre d'applications, mais il pose le problème de l'annotation des données. Cette annotation est longue et fastidieuse, surtout en considérant la grande taille des données nécessaire pour obtenir une efficacité suffisante. Il serait donc intéressant d'annoter partiellement des données et de compléter l'apprentissage supervisé en utilisant les données non annotées : c'est le principe de l'apprentissage semi-supervisé. L'étiquetage des données pour la segmentation d'images (ou encore segmentations manuelles) étant particulièrement chronophage, on trouve dans la littérature plusieurs voies pour alléger cette tâche ([papandreou2015semisupervised], [hong2015decoupledNet], [zheng2018semisupervised]).

1.4.2.4/ L'APPRENTISSAGE PAR RENFORCEMENT

L'apprentissage par renforcement repose sur un système de récompenses permettant de conditionner un comportement. Il tire ses origines des sciences comportementales. Par exemple, on peut conditionner un animal en lui donnant une récompense lorsqu'il effectue une bonne action. Celui-ci apprend à faire le lien entre les bonnes actions et les récompenses et cherchera à effectuer ces bonnes actions afin de maximiser le gain de récompenses. Le même principe est utilisé pour entraîner un programme. Une série d'actions est possible et à chaque action est associé un gain plus ou moins important ou une perte plus ou moins importante. Le but n'est pas seulement de créer un programme maximisant le gain de récompense sur le court terme (ce qui est plutôt simple si l'on connaît le gain ou la perte associé(e) à chaque action) mais aussi de maximiser ce gain sur le long terme. Dit autrement, le programme doit être capable de sacrifier un gain sur le court terme si cela lui permet de maximiser son gain sur le long terme.

Parmi les techniques d'apprentissage par renforcement, on peut citer le *Temporal Difference (TD) learning* ou encore le *Q-learning*. Les applications au traitement d'image sont beaucoup plus rares que pour les catégories d'apprentissages précédentes. L'apprentissage par renforcement peut toutefois être ponctuellement utilisé dans ce domaine ([goel2018reinforcementLearning]).

1.4.3/ LES MÉTHODES DE SEGMENTATION

Les méthodes de segmentation sont très variées allant du simple seuillage à des approches régions comme la croissance de régions, en passant par des analyses de textures ou encore de la morphologie mathématique. Parmi elles, les procédés d'intelligence artificielle semblent être des méthodes efficaces ouvrant la voie à une automatisation complète du processus.

1.4.3.1/ ÉTATS DE L'ART PRÉEXISTANTS SUR LA SEGMENTATION D'IMAGES PAR INTELLIGENCE ARTIFICIELLE

Hemanth et al. [hemanth2009survey] ont réalisé un état de l'art sur toutes les méthodes d'intelligence artificielle pour la classification et la segmentation de lésions cérébrales sur des images obtenues par Imagerie par Résonance Magnétique (*IRM*). Ce travail, bien que relativement ancien (2009, c'est à dire avant l'utilisation massive des réseaux de neurones profonds), permet d'avoir une vision générale du domaine. Pour la segmentation tumorale par IA, Hemanth et al. listent trois types de techniques :

- Les premières sont les réseaux de neurones, se limitant à l'époque à des perceptrons multi-couches (MLP) et à des réseaux de Kohonen (cartes auto-organisatrices).
- Les deuxièmes sont les techniques basées sur la logique floue comme la classification floue (*FCM*).
- Enfin les troisièmes sont des approches hybrides mariant les réseaux de neurones avec la logique floue.

Dans une étude plus large, car ne se limitant pas à une application précise, Jaiswal et al. [jaiswal2013survey] mettent en évidence des travaux portant sur des approches diverses pour la segmentation d'images, telles que les algorithmes génétiques [binh2012improving], l'optimisation de colonies de fourmis [jevtie2010adaptive] ou encore une segmentation par la méthode des *K-Means* [chandhok2012approach]. Cependant, tous les algorithmes développés donnent des résultats très dépendants du type d'images traitées. Les images médicales étant un type très particulier, car elles sont souvent assez bruitées, il est difficile d'anticiper les performances de ces techniques.

Les différents états de l'art publiés dans la littérature (Litjens et al. [litjens2017Survey], Zhao et al.[zhao2017survey]) témoignent du fort intérêt de la communauté scientifique en vision par ordinateur pour les réseaux de neurones profonds, de la richesse de la littérature sur le sujet et de l'efficacité de cette approche. Litjens et al. [litjens2017Survey] ont présenté l'ensemble des travaux en *Deep learning* dans un contexte médical, ce qui représente donc une source de choix pour orienter notre propre état de l'art sur le sujet.

La Figure 1.15 illustre notre proposition de classification de toutes ces techniques de segmentation, par Intelligence Artificielle ou non, issue de l'étude des travaux abordés dans la littérature. Cette classification nous permet d'offrir une vue d'ensemble du domaine, bien que non exhaustive. Dans un souci de lisibilité, nous avons mis des frontières qui peuvent être discutées entre certaines techniques existantes.

1.4.3.2/ LES APPROCHES PRIVILÉGIÉES

Une première approche est basée sur **l'expérience dans laquelle un système apprend par entraînement sur un ensemble de données**. Dans cette approche, soit le système est guidé par un humain dans son apprentissage grâce à des sorties *a priori* à respecter (on parle alors d'apprentissage supervisé), soit il doit lui-même déterminer les règles à suivre pour produire un résultat (on parle dans ce cas d'apprentissage non supervisé ou encore de *clustering*). L'apprentissage non supervisé constitue un terrain d'investigation

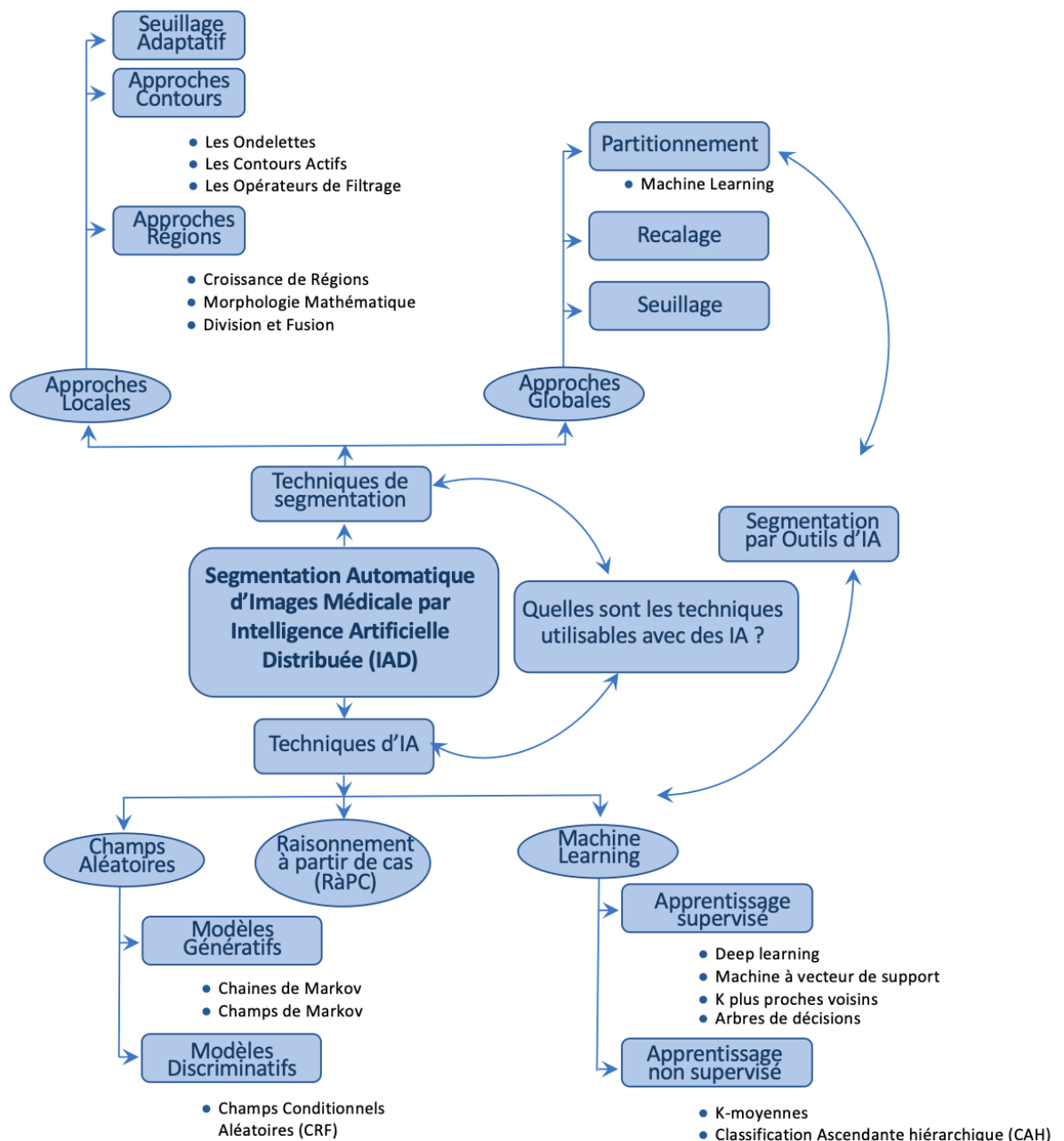


FIGURE 1.15 – Proposition de classification des différentes techniques de segmentation et des approches d'intelligence artificielle

très vaste dans le domaine de la recherche en IA et représente une sorte d'idéal qu'il est particulièrement difficile d'atteindre. Ainsi, les techniques actuellement les plus efficaces restent pour le moment de type supervisé, au premier rang desquelles on peut citer **les réseaux de neurones** qui suscitent un véritable engouement ces dernières années.

Une seconde approche prend un chemin un peu différent pour **exploiter une base de connaissance afin de résoudre un problème nouveau** : il s'agit des systèmes experts et du **Raisonnement à Partir de Cas (RàPC)** consistant en un système avec une capa-

citée de remémoration d'un cas déjà résolu similaire au nouveau problème rencontré afin de faciliter sa résolution. Bien que moins présent dans l'état de l'art que les réseaux de neurones profonds pour la segmentation d'images, le RàPC est un champ d'investigation pertinent pour compléter un système d'intelligence artificielle, avec un fort potentiel dans un contexte médical dans lequel l'accès à des données images est soumis à beaucoup plus de contraintes que dans un contexte plus générique. Couplé à de la logique floue et à l'utilisation d'une ontologie permettant de situer les organes les uns par rapport aux autres dans le corps humain, le RàPC offre encore plus de possibilités.

Les états de l'art des deux chapitres suivants de ce manuscrit portent sur ces deux approches, plus particulièrement sur les réseaux de neurones d'une part et sur le Raisonnement à Partir de Cas d'autre part. D'autres techniques permettent d'automatiser les segmentations d'images mais, moins utilisées et moins efficaces, elles ne seront pas abordées. Toutefois, **le plus gros potentiel réside dans l'hybridation de ces techniques entre elles** pour profiter des avantages de chacune tout en palliant leurs défauts respectifs.

Synthèse du chapitre

Ce premier chapitre a présenté le projet ASKid, dont la thématique est médicale, et qui consiste à réaliser de manière automatique la segmentation de structures de l'abdomen chez de jeunes enfants, dont principalement le parenchyme rénal et le néphroblastome.

Les images médicales segmentées sont des clichés tomodensitométriques exploitant l'utilisation de rayons X et leur absorption par les tissus biologiques suivant leur densité. Le principe de fonctionnement du scanner permet notamment d'obtenir des images en coupes transversales. Grâce à l'utilisation d'un produit de contraste, il est possible de différencier plus facilement sur les images produites les structures d'intérêts les unes par rapport aux autres et ainsi d'effectuer les segmentations nécessaires au diagnostic.

Un problème de segmentation d'images s'apparente à un problème de classification de données et peut être résolu par le recours à l'apprentissage machine. Il existe plusieurs types d'apprentissage, dont le plus utilisé est l'apprentissage supervisé. La quantité de données nécessaire à un apprentissage efficace forme cependant sa principale limite qu'il faut parvenir à surmonter. Les techniques d'Intelligence Artificielle applicables à une problématique de segmentation d'images sont nombreuses et la littérature est très riche sur ce sujet. Les deux grandes approches qui ont retenues notre attention sont les Réseaux de Neurones Convolutifs (*CNN, Convolutional Neural Networks*), pour leur efficacité et la littérature prolifique les concernant, ainsi que le Raisonnement à Partir de Cas (RàPC), qui offre une approche connaissance complémentaire à l'approche purement expérimentale des CNN.

Dans le chapitre suivant, nous présentons notre état de l'art concernant les réseaux de neurones convolutifs. Le chapitre décrira notamment les architectures des réseaux les plus présents dans la littérature pour la segmentation d'image et leur application dans un contexte médical.

LES RÉSEAUX DE NEURONES ET L'APPRENTISSAGE PROFOND

L'apprentissage machine regroupe des techniques diverses. Parmi celles-ci, les réseaux de neurones et l'apprentissage profond forment un champ de recherche particulièrement actif, en particulier depuis les années 2010 au cours desquelles les moyens matériels sont devenus suffisants pour rendre cette approche efficace (aussi bien en terme de *hardware* qu'en terme de bases de données pour l'apprentissage). La littérature sur le sujet est très riche et il existe peu de domaines dans lesquels le *Deep Learning* n'est pas expérimenté. Les réseaux de neurones dans le contexte de l'imagerie médicale ont trouvé un essor particulier compte tenu de tout ce que peut apporter l'utilisation de l'intelligence artificielle, en particulier pour la segmentation. En 2017, Litjens *et al.* ont produit un état de l'art très complet sur tous les travaux existants dans le domaine de l'analyse d'images médicales, regroupant plus de 300 références [litjens2017Survey].

Dans ce chapitre, les bases du fonctionnement d'un réseau de neurones seront présentées dans un premier temps avant d'aborder les architectures spécifiques pour le traitement d'images. Enfin, les réseaux dédiés à la segmentation les plus présents dans la littérature seront listés et détaillés dans un état de l'art.

2.1/ LE PERCEPTRON

2.1.1/ NOTIONS DE BASE

Le perceptron fonctionne par analogie avec un neurone biologique pour créer un neurone artificiel fonctionnant selon les mêmes principes. Imaginé en 1957 par Franck Rosenblatt [rosenblatt1958perceptron], il est représenté comme un système à n entrées et 1 sortie comme illustré sur la Figure 2.1.

Chaque entrée x_n est associée à un poids w_n . Une fonction, appelée fonction d'activation du neurone, est appliquée à la somme Σ des produits des entrées par leurs poids respectifs afin de produire une sortie. Les poids w forment les paramètres du perceptron. Un dernier paramètre est souvent ajouté par l'intermédiaire d'un biais b , additionné à Σ , avant l'application de la fonction. Le biais permet de contrôler le déplacement de la fonction d'activation selon l'axe des abscisses. Initialement, son rôle était de déterminer la valeur de seuil d'une fonction d'activation.

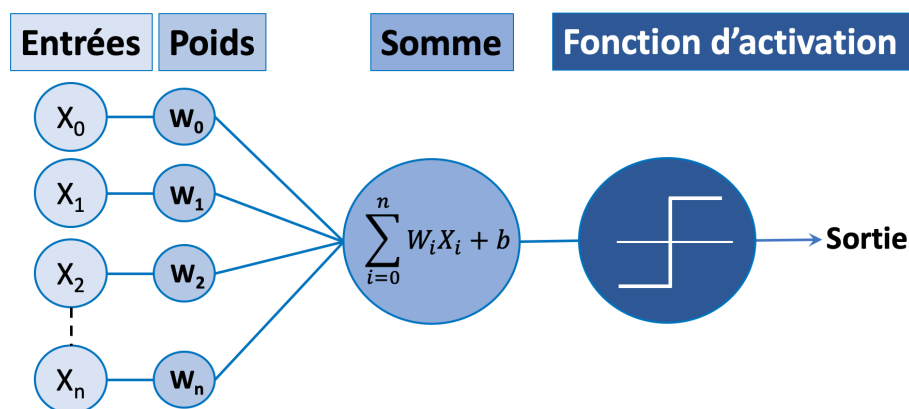


FIGURE 2.1 – Présentation d'un perceptron

Définition

Soit un neurone artificiel ayant pour paramètres un ensemble de poids w et un biais b . Sa sortie est calculée par l'application d'une fonction d'activation f sur la somme pondérée des entrées Σ telle que :

$$\Sigma = \sum_{i=1}^m x_i w_i$$

$$f(z) = f(\Sigma + b)$$

La fonction d'activation est historiquement une fonction seuil par analogie avec un neurone biologique (qui obéit à une loi « *du tout ou rien* »). Lorsque la somme pondérée dépasse un certain seuil, la fonction renvoie une valeur de 1, sinon 0. Afin de contrôler le seuil d'activation de la fonction, un biais b est additionné à la somme pondérée, ce qui permet en même temps de faire de ce seuil un paramètre du réseau (qui devient donc modifiable par apprentissage, cf 2.2 pour plus de détails). Généralement, l'entrée de la fonction d'activation est notée z telle que $z = \sum_{i=1}^m x^i w_i + b$. Dit autrement, le rôle de la fonction d'activation est d'appliquer une non linéarité en sortie du neurone.

Définitions

Différents types de fonctions d'activation peuvent être appliqués :

- La fonction seuil : $\sigma(x) = \begin{cases} 1 & \text{Si } \Sigma > \text{seuil} \\ 0 & \text{Sinon} \end{cases}$
- La fonction Sigmoidé : $\sigma(x) = \frac{1}{1+e^{-x}}$
- La fonction Tanh : $\sigma(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
- La fonction ReLU : $\sigma(x) = \max(0, x)$

Le choix de la fonction d'activation impacte également l'apprentissage au sein d'un réseau de neurones, notamment en accentuant ou en atténuant une disparition du gradient (*vanishing gradient*) qui peut nuire à son efficacité. De plus, elle doit être continue et dérivable pour permettre un apprentissage automatique, ce qui n'est par exemple pas le cas de la fonction seuil. Ces sujets ne sont pas davantage abordés dans ce manuscrit.

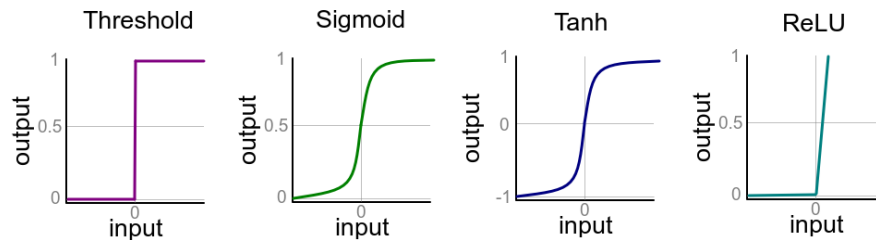


FIGURE 2.2 – Les fonctions d'activation

2.1.2/ LE PERCEPTRON MULTICOUCHE

Le perceptron multicouche a été inventé dans les années 1980. Il s'agit d'une succession de couches contenant plusieurs neurones (ou perceptrons simples). Chaque neurone d'une couche est relié à la totalité des neurones des couches adjacentes (cf Figure 2.3). Ainsi, le terme de couche *Fully Connected* est utilisé pour désigner les couches de ce type.

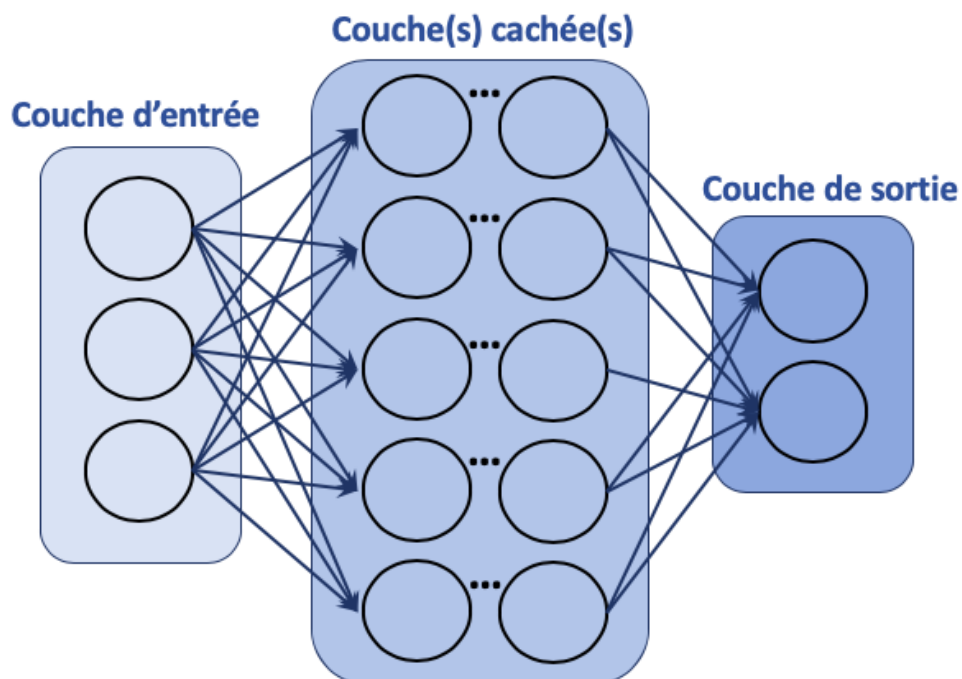


FIGURE 2.3 – Perceptron multicouche

Un perceptron multicouche se divise en une couche d'entrée, une ou plusieurs couches cachées et enfin une couche de sortie. Chaque résultat des fonctions d'activations des

neurones à la couche $n - 1$ sera alors transmis à tous les neurones de la couche n . Chaque neurone de la couche n effectue la somme de toutes leurs entrées pondérées, et transmet ce résultat à la fonction d'activation et envoie la sortie à tous les neurones de la couche $n + 1$, et ainsi de suite jusqu'à la dernière couche.

2.2/ L'APPRENTISSAGE PROFOND

Le concept de *Deep Learning* ou *apprentissage profond* n'apparaît qu'au début des années 2010. Il s'agit d'une manière particulière de réaliser le *Machine Learning*. Le *Deep Learning* consiste à réaliser un réseau de neurones profond, sur le même modèle que les perceptrons multicouches, mais avec de nombreuses couches cachées (plus de deux). Ce réseau est alors capable d'extraire des informations complexes de haut niveau des données lui étant présentées. Cependant, le nombre de paramètres est alors bien trop important pour pouvoir être déterminé empiriquement. Il est donc nécessaire d'avoir recours à un apprentissage automatique. Il s'agit d'un type d'apprentissage supervisé, c'est-à-dire que le réseau a besoin d'une base d'apprentissage pour s'entraîner et ainsi trouver les meilleures paramètres pour obtenir le résultat attendu.

2.2.1/ LE PRINCIPE GÉNÉRAL DE L'APPRENTISSAGE

Comme le montre la Figure 2.4, l'entrée est transmise au système, c'est-à-dire au réseau de neurones, ce qui permet de calculer une valeur de sortie. La sortie prédite est alors comparée à la sortie désirée grâce à une fonction de coût. Le rôle de cette fonction est de quantifier l'erreur du réseau. Puis, l'apprentissage consiste à ajuster les paramètres du système, via un processus itératif, afin de minimiser l'erreur mesurée en sortie. Ceci revient à étudier les variations de l'erreur et à déterminer pour quelles valeurs de paramètres celle-ci est minimisée. Il s'agit donc d'un problème d'optimisation mathématique dans lequel l'objectif est de trouver le minimum d'une fonction. La seule différence est que la fonction de coût est multidimensionnelle avec un grand nombre de variables en entrée et de paramètres dans le réseau (elle ne peut donc pas être représentée graphiquement dans un espace 2D ou 3D).

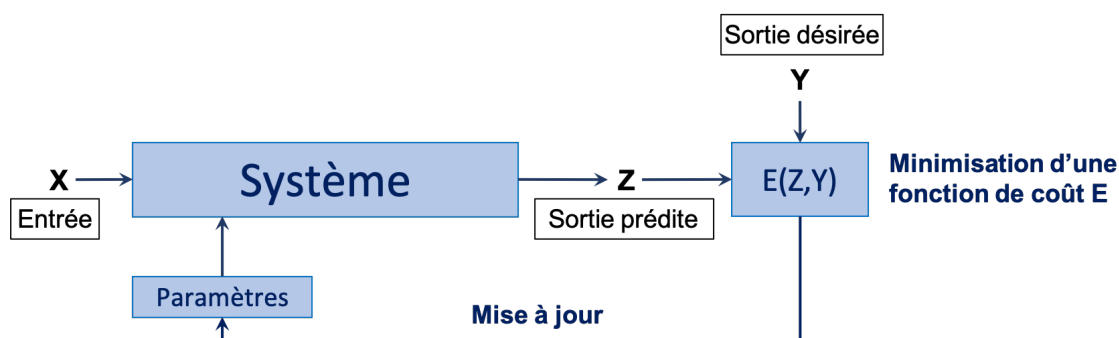


FIGURE 2.4 – Le principe général de l'apprentissage

L'ajustement des paramètres suit le principe de la *rétropropagation du gradient* (ou *back-propagation*) qui fut utilisée pour la première fois dans les réseaux de neurones par Rumelhart *et al.* [rumelhart1988learning] et par Yann Lecun [lecun1988backpropa]. Cette

méthode d'apprentissage s'appuie sur un algorithme de *descente du gradient*, qui est une approche simple et classique pour résoudre des problèmes d'optimisation, afin de rechercher le minimum d'une fonction de coût avec les paramètres associés.

2.2.2/ LA RÉTROPROPAGATION DU GRADIENT

Dans les réseaux de neurones multicouches, il faut désormais pouvoir ajuster les paramètres des différentes couches du réseau à partir de la sortie. La technique de rétropropagation du gradient est alors appliquée dans le cas du Deep Learning : technique prenant de l'engouement en 1988 grâce à David Rumelhart *et al.* [rumelhart1988learning]. Le principe est illustré par la Figure 2.5.

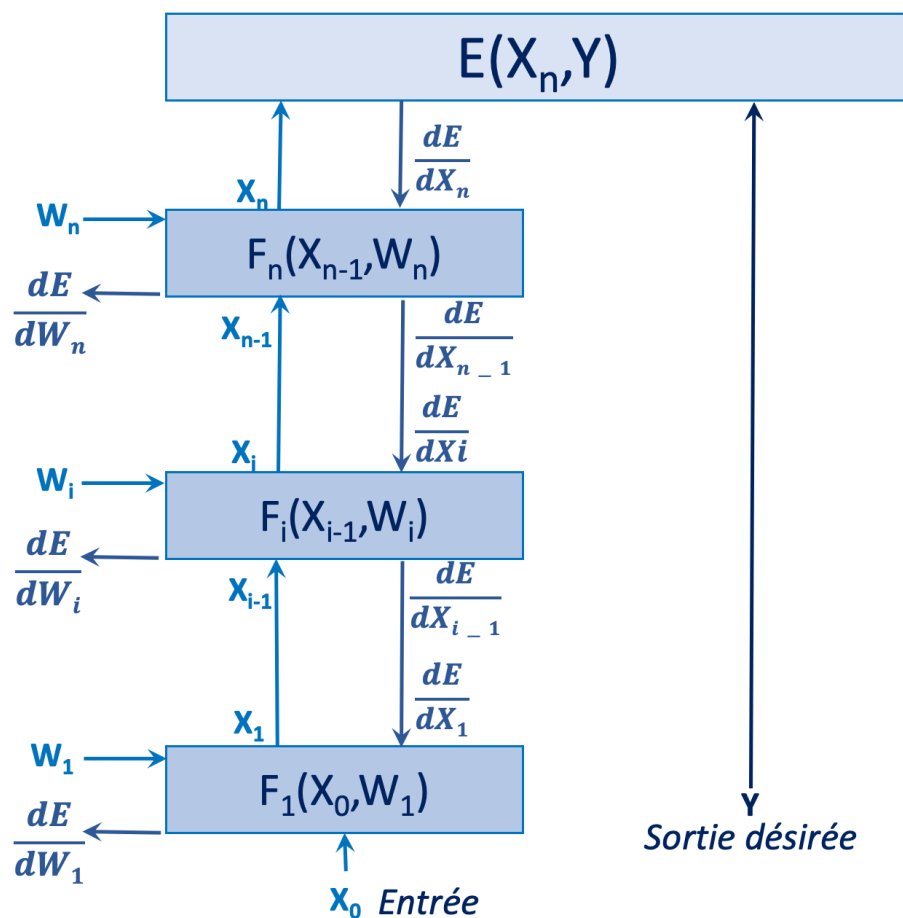


FIGURE 2.5 – Principe de rétropropagation du gradient

Tout d'abord, le système initialise les paramètres de chaque couche (poids + biais de chaque neurone) de façon aléatoire ou avec des valeurs définies. L'entrée est ensuite propagée à travers le réseau pour produire une sortie (*forward*). Cette sortie est comparée à une sortie désirée par l'intermédiaire d'une fonction de coût $E(X_n, Y)$ avec Y la sortie désirée et X_n la sortie du réseau (sortie de la couche n). Le gradient de cette fonction de coût par rapport à la sortie de la couche n $\frac{\partial E}{\partial X_n}$ est calculé. À partir de ce gradient en sortie de couche, on peut calculer le gradient en entrée de cette même couche et ainsi de suite jusqu'à remonter au niveau de l'entrée du réseau (*backward*). En parallèle, les gradients de la fonction E par rapport à chacun des paramètres W_i (poids et biais des

neurones) de la couche i , $\frac{\partial E}{\partial w_i}$, sont également déterminés.

Le gradient $\frac{\partial E}{\partial w_i}$ représente la part d'erreur attribuable au paramètre w_i . L'apprentissage consistera à modifier ces paramètres à chaque itération afin de minimiser l'erreur E . Cette modification est classiquement effectuée avec un algorithme de descente du gradient stochastique (souvent abrégé « SGD » dans la littérature : Stochastic Gradient Descent), mais d'autres méthodes d'optimisation peuvent également être utilisées pour davantage d'efficacité suivant le contexte.

2.3/ LES RÉSEAUX DE NEURONES CONVOLUTIFS

2.3.1/ PRÉSENTATION GÉNÉRALE

Un réseau de neurones convolutif ou CNN (*Convolutional Neural Network*) est un type de réseaux reposant sur des opérations particulières, dites de convolutions/déconvolutions, très utilisées en traitement d'images. Son organisation est identique à celle d'un perceptron multicouche. Les neurones sont organisés en couches successives permettant d'extraire de l'information de plus en plus abstraite à partir des données brutes. La différence se trouve dans la manière d'extraire cette information. Les CNN sont basés sur une analogie avec la vision humaine. Chaque neurone n'est connecté qu'à une partie des données via ce que l'on nomme « *un champ de réception* ». Dans le contexte de l'analyse d'images, cela signifie qu'un neurone ne traitera qu'une partie de l'image en entrée et que c'est l'ensemble des neurones présents au sein d'une couche (appelée « *couche de convolution* ») qui permettra de traiter l'image dans sa globalité. Chaque couche de convolution produit une ou plusieurs cartes de caractéristiques (le terme *feature map* est largement utilisé pour les désigner) à partir du résultat de la couche précédente (ou de l'image d'origine pour ce qui est de la première couche) jusqu'à la sortie finale du réseau. Suivant le filtrage utilisé lors de la convolution (taille et valeurs du filtre), des caractéristiques différentes de l'image en entrée peuvent être mises en évidence (par exemple des contours) d'où le terme de « *carte de caractéristiques* ». Il est strictement équivalent à celui de « *carte d'activation* » car correspondant aussi à l'activation (sortie) des différents neurones composant la couche de convolution.

D'autres types de couches sont également présentes dans les CNN, pouvant être optionnelles ou non et jouant chacune un rôle particulier :

- **La couche de correction** a un rôle complémentaire à celle de convolution. Un certain nombre de bibliothèques de *Deep learning* implémentent les couches de convolution comme une opération linéaire, sans fonction d'activation. La couche de correction permet d'ajouter cette fonction d'activation indépendamment de l'opération de convolution en amont. La couche ReLU, permettant d'appliquer une fonction ReLU, est la plus courante pour son efficacité [glorot2011rectifier].
- **La couche de pooling** est une couche dont le rôle est de réaliser un sous-échantillonnage en divisant la définition d'une *feature map* par 2. Il existe différents types de *pooling* suivant la méthode utilisée pour effectuer le sous-échantillonnage (*mean pooling* ou *max pooling*). En plus d'alléger les calculs tout en conservant les informations les plus importantes (car diminuant la quantité de données à traiter), cette couche permet également d'introduire une invariance à la translation (le

réseau devient moins sensible à un changement de position des caractéristiques dans l'image).

- **La couche de déconvolution** peut être vue comme une étape de sur-échantillonnage. D'ailleurs, il est courant d'initialiser ces couches avec un noyau réalisant par exemple une interpolation bilinéaire. Le terme « *déconvolution* » est en fait un abus de langage du domaine et le processus ne correspond pas à une opération de déconvolution au sens mathématique. En réalité, il est plus juste de parler de « *convolution transposée* ». Contrairement à une déconvolution qui consiste à récupérer un signal identique à celui qui a été convolué (mêmes dimensions et mêmes valeurs), la convolution transposée se contente de reconstruire un signal ayant les mêmes dimensions. Les couches de déconvolution sont donc utilisées pour compenser l'effet des convolutions (si elle sont réalisées sans remplissage, *padding* en anglais) et obtenir en sortie du réseau une carte de segmentation correspondant à l'image en entrée.
- **La couche de *unpooling*** est également une étape de sur-échantillonnage mais existe pour inverser la perte de résolution due cette fois aux couches de *pooling*. Chaque couche de *unpooling* est appariée à une couche de *pooling* particulière. Cet appariement permet de sauvegarder la configuration des pixels lors d'une opération de *pooling* (par exemple : résolution de l'image et localisation des pixels de valeurs maximales dans le cas d'un *max pooling*) pour effectuer ensuite l'opération inverse. Évidemment, le sous-échantillonnage entraîne une perte d'information. Les valeurs de pixels ne pouvant pas être récupérées sont mise à zéro.
- **La couche de perte** est la dernière couche d'un réseau dans sa configuration d'entraînement et a pour rôle de guider l'apprentissage du réseau en calculant son erreur selon une fonction de coût particulière. Les fonctions de coût utilisées sont assez diverses : log-vraisemblance négative, *cross entropy*, erreur quadratique moyenne (*square error*) ou encore la perte logistique moyenne (*Multinomial logistic loss*). L'une des plus courantes est la fonction *cross entropy* pour son efficacité dans la majeure partie des situations.
- Enfin, **la couche SoftMax** est une couche de normalisation permettant de rendre plus intelligible les scores calculés par un réseau de neurones en les transformant en probabilités (dont les valeurs sont donc comprises entre 0 et 1). Elle est couramment couplée à la couche de perte pour faciliter l'apprentissage du réseau et est conservée pour l'étape d'inférence (calcul des prédictions sur de nouvelles données).

Les dernières couches de l'architecture dépendent de son application. Initialement dédiés à la classification d'images, les premiers CNN possèdent des couches classiques que l'on retrouve dans un perceptron, désormais appelées couches *fully connected* pour les distinguer, dans lesquelles les neurones sont liés aux sorties de tous les neurones de la couche précédente. Elles sont depuis couramment remplacées par des convolutions 1x1 pour créer des réseaux totalement convolutifs (*Fully Convolutional Network*). La prédiction d'un CNN est, dans ce cas, un vecteur de scores pour une série de labels caractérisant l'image en entrée (voir Figure 2.6). Les réseaux dédiés à la segmentation disposent généralement, quant à eux, d'une ou plusieurs couches de déconvolution aboutissant non pas à un vecteur de scores mais à une carte de scores. Cette carte est

souvent désignée sous le terme de « arte de probabilités » après la normalisation des scores par une couche SoftMax.

Les réseaux sont généralement basés sur l'un des trois grands types d'architectures pour la classification d'objets :

- AlexNet [krizh2012alexNet] est composé de 5 couches de convolution, suivies par 3 couches *fully connected* pour un total de 8 couches. Une opération de *Max pooling* est réalisée après les couches de convolution c1, c2 et c5 pour diminuer la résolution des cartes de caractéristiques (*features maps*).
- VGG16 [simonyan2014very] est une architecture plus profonde composée de 13 couches de convolution réparties en 4 groupes de convolutions consécutives (formés par 2 convolutions consécutives pour le premier groupe, 2 pour le deuxième, puis 3 pour chacun des groupes restants). Une couche de *Max pooling* réduit la résolution après chaque groupe. Enfin, 3 couches *fully connected* puis une couche SoftMax terminent le réseau. Pour la segmentation d'images, VGG16 est l'architecture la plus utilisée car elle représente un bon compromis entre performances et nombre de paramètres à entraîner.
- GoogleNet [szegedy2015conv] est l'architecture la plus profonde avec 22 couches consécutives. Sa particularité est la présence de modules, appelés par les auteurs *inception layer*, permettant d'extraire des informations à différentes échelles (avec un contexte plus au moins important). Concrètement, cela consiste à réaliser en parallèle des convolutions avec différentes tailles de filtres sur une même image, puis à concaténer leurs résultats afin de passer l'ensemble à la couche suivante.

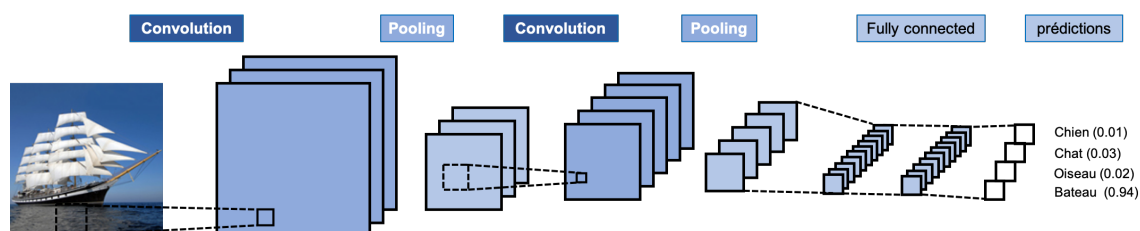


FIGURE 2.6 – Exemple d'architecture d'un CNN (*Convolution Neural Network*)

La Figure 2.6 présente un exemple d'architecture pour classifier des images en fonction de la scène qu'elles représentent. Dans cet exemple, le réseau est conçu pour reconnaître quatre types de scènes : images représentant un chat, un chien, un oiseau ou un bateau. Le réseau produit en sortie un vecteur de scores pour chacun de ces quatre labels. Ces scores peuvent être interprétés comme un pourcentage de caractéristiques de tel ou tel objet retrouvé dans l'image. Dans l'exemple, le réseau a déterminé que les caractéristiques de l'image en entrée correspondaient à 94% à celles d'un bateau, à 0,04% à celles d'un chat, . . . La conclusion du réseau est généralement le label ayant le score le plus élevé (dans le cas où l'on cherche à attribuer un label unique).

Le prédécesseur des CNN est le Neocognitron qui a été introduit dans les années 1980 par Kuniyuki Fukushima *et al.* [fukushima1982neocognitron], mais il est couramment admis que le premier CNN a été mis au point par Yann LeCun *et al.* sous le nom LeNet [lecun1989generalization]. En plus des couches de convolution, on y distingue des couches particulières appelées « *couche de pooling* » diminuant la résolution des *features*

map par sous-échantillonnage (généralement divisant la résolution par deux en conservant la valeur maximale, on parle alors de *Max pooling*). Les auteurs perfectionnent le réseau pour l'appliquer à la reconnaissance des écritures manuscrites sur des chèques (nombres écrits de manière manuscrite) en 1998 [lecun1998gradient]. Les CNN récents sont néanmoins bien plus profonds (possédant plus de couches) mais il restent basés sur la même opération mathématique particulière : la convolution.

2.3.2/ LA CONVOLUTION DANS LES RÉSEAUX DE NEURONES

La convolution, ou produit de convolution, est une opération mathématique entre deux fonctions notée « * » et renvoyant une nouvelle fonction dites « convoluée ». La convolution généralise la notion de moyenne glissante consistant à calculer une série de moyennes locales sur un signal.

Définition

La forme discontinue du produit de convolution entre un signal f et un signal g non bornés est exprimée telle que :

$$(f * g)(n) = \sum_{m=-\infty}^{+\infty} f(n-m)g(m) = \sum_{m=-\infty}^{+\infty} f(m)g(n-m)$$

Dans le domaine du traitement d'images, la convolution consiste à superposer un filtre (que l'on appelle également *noyau de convolution*) à la partie de l'image que l'on souhaite convoluer. La sortie du filtre est calculée en multipliant respectivement les valeurs des pixels de l'image avec les valeurs associées du filtre, puis en les additionnant comme illustré par la figure 2.7. Le filtre est ensuite décalé sur l'image d'un certain pas pour produire une nouvelle sortie. Lorsque le filtre a parcouru la totalité de l'image, l'ensemble des valeurs de sortie forme *l'image convoluée*. On peut remarquer que la convolution décrite est quelque peu différente de sa définition mathématique, qui nécessiterait une transposition du noyau, et correspond en réalité à une opération de *corrélacion croisée* (*cross correlation*). Cependant, par abus de langage et parce que l'usage de l'un ou l'autre ne modifie pas les performances dans un réseau convolutif, le terme de « convolution » est communément employé.

La résultat d'une convolution peut être modifié suivant plusieurs paramètres :

- La taille du filtre utilisé : au sein des CNN, les filtres sont généralement de dimensions 2x2, 3x3 voire 5x5 car une taille trop importante implique un grand nombre de paramètres à entraîner.
- Le pas (*stride*) qui représente la valeur de décalage du filtre sur l'image entre chaque itération.
- Et enfin la valeur du *remplissage*, communément appelé le *padding*, qui consiste à ajouter un certain nombre de lignes et de colonnes de valeurs nulles autour de l'image. Ce paramètre est particulièrement utile pour pallier la réduction des dimensions de l'image induite par l'opération de convolution.

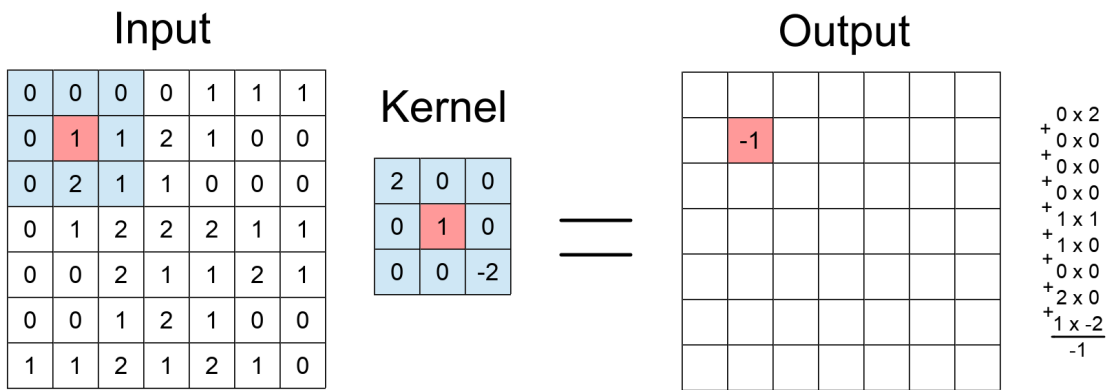


FIGURE 2.7 – Le principe de la convolution

Au sein d'un CNN, la notion de convolution est intimement liée à celle de neurone. Contrairement aux couches *fully connected*, les neurones d'une couche de convolution ne sont pas connectés à la totalité de leurs prédécesseurs mais seulement à un nombre limité d'entre eux. Ce principe est à l'origine de l'idée du « champ de perception » (*Receptive field*) d'un neurone qui peut être directement relié à une opération de convolution. La Figure 2.8 en donne une illustration. Un neurone réalise une somme pondérée entre toutes ses entrées suivant les poids affectés à chacune d'elle. Ceci correspond donc à une convolution. Chaque neurone d'une couche de convolution réalise l'opération sur une partie dédiée de l'image, les sorties de l'intégralité des neurones formant l'image convoluée. Les paramètres (poids et biais) étant partagés par les neurones réalisant la même opération de convolution au sein d'une couche, leur nombre peut être adapté à la volée suivant les dimensions de l'entrée.

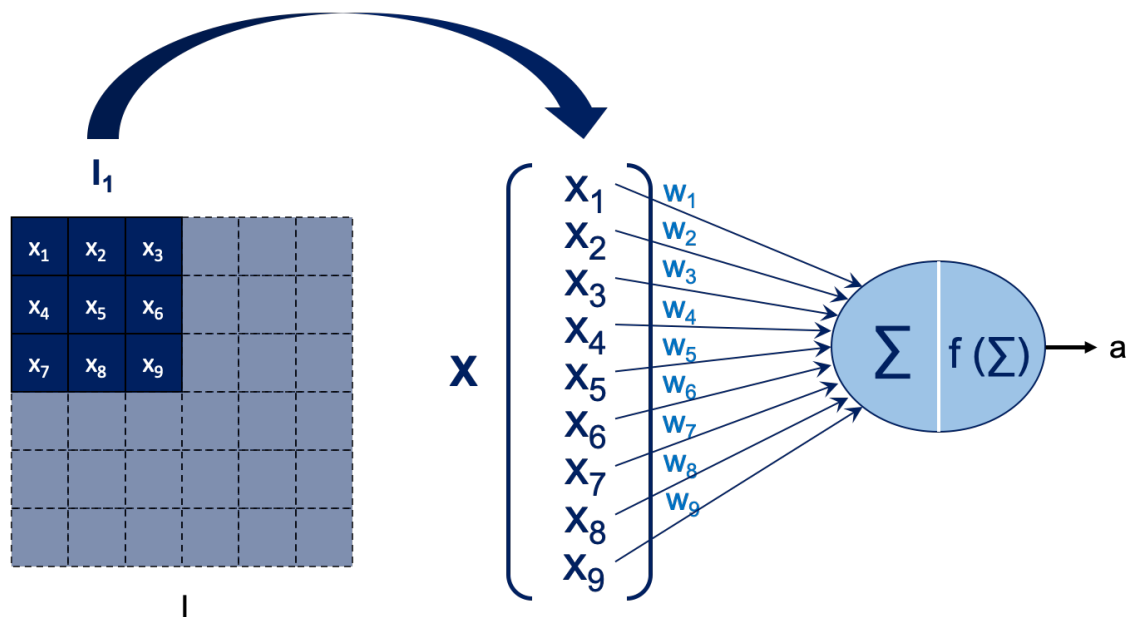


FIGURE 2.8 – La convolution et les neurones artificielles

2.4/ LES RÉSEAUX DE NEURONES CONVOLUTIFS POUR LA SEGMENTATION D'IMAGES

Nous nous intéressons dans nos travaux en particulier aux différents réseaux existants dédiés à la segmentation d'images. Ces réseaux sont généralement des CNN modifiés en incorporant une partie déconvolutive à l'architecture afin de pouvoir retrouver les dimensions initiales de l'image traitée. L'objectif est de faire produire au réseau une carte de segmentation qui corresponde à l'attribution d'un score propre à chaque classe non pas pour la totalité de l'image mais pour chaque pixel composant cette image. Dans ce contexte, la base d'apprentissage est constituée des images d'origine et des images segmentées correspondantes en tant que vérités terrains pour que le réseau puisse apprendre des résultats attendus.

L'utilisation du *deep learning* pour la segmentation représente un domaine de recherche très actif, notamment depuis les années 2012, et plusieurs réseaux ont émergés. La plupart d'entre eux ont été conçus pour résoudre un problème de segmentations sur des images grand public pour des applications variées (reconnaissance d'objets, guidage d'une voiture autonome, ...) comme *Fully Convolutional Network* [long2015fcn] ou encore *DeepLab* [chen2018deeplab]. D'autres ont directement trouvé une application dans le domaine biomédicale à l'image de *U-Net* [ronneberger2015unet]. Les principaux réseaux issus de la littérature sont détaillés dans les sections suivantes.

2.4.1/ FULLY CONVOLUTIONAL NETWORK

Les premiers réseaux de neurones spécifiques à la segmentation d'image apparaissent en 2015 avec la méthode FCN ou *Fully Convolutional Network* [long2015fcn]. Ce nouveau réseau de neurones profond est un CNN classique dans lequel les couches *fully connected* ont été remplacées par des couches de convolution. La dernière couche est une couche de sur-échantillonnage (déconvolution) dans laquelle l'image produite est de la même taille que l'image en entrée. La Figure 2.9 donne une illustration de l'architecture où, pour chaque couche, le nombre de *feature maps* est indiqué. La suppression des couches *fully connected* au profit de convolutions offre l'avantage de pouvoir utiliser le réseau sur des images de tailles arbitraires, en plus d'accélérer l'apprentissage en diminuant le nombre de paramètres. FCN désigne désormais les réseaux reprenant ce principe, soit une absence de couche *fully connected* dans leur architecture.

La déconvolution est effectuée avec un pas (*stride*) de 1/32 (on parle ici du pas effectif, c'est-à-dire du pas simulé par l'ajout de zéros entre les pixels avant la déconvolution) et un noyau de 32×32 , d'où le nom FCN-32s. Il est possible de rendre les segmentations plus précises grâce à des variantes de ce réseau. Pour ces réseaux modifiés, de l'information supplémentaire provenant des couches de *pooling* en amont est combinée avec le résultat du réseau juste avant l'étape de déconvolution. Suivant la(les) couche(s) de *pooling* mise(nt) à contribution, on parle de FCN-16s ou de FCN-8s qui gagnent encore davantage en précision (cf Figure 2.10). La réutilisation et la combinaison des sorties des différentes couches de *pooling* du réseau permettent d'exploiter une information multi-échelle pour le calcul de la segmentation, renforçant ainsi la précision du résultat.

La Figure 2.10 résume le fonctionnement et les différences entre les trois réseaux FCN. Pour davantage de clarté, seuls les résultats des différentes couches de *pooling* y fi-

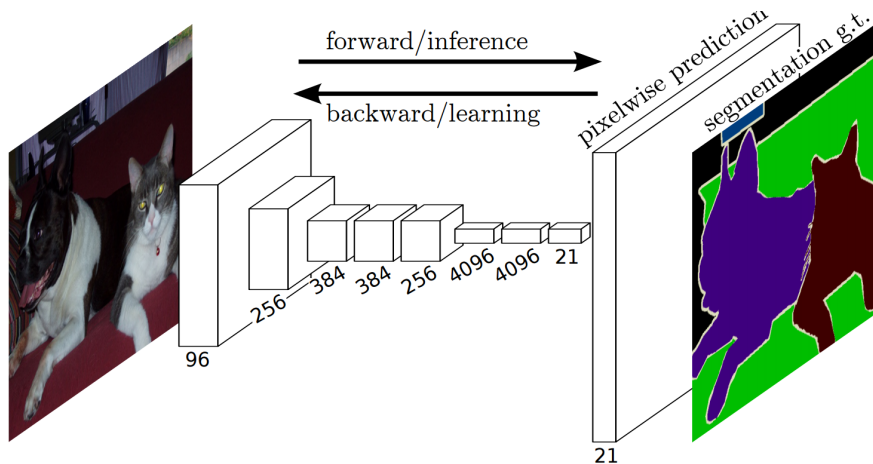


FIGURE 2.9 – Fully Convolutional Network [long2015fcn]

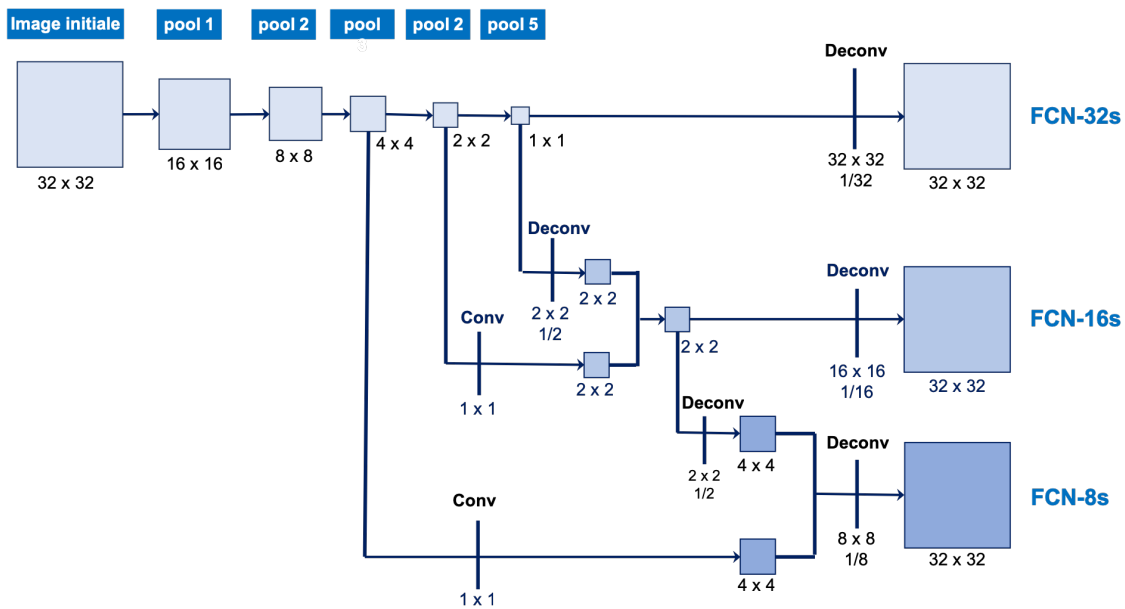


FIGURE 2.10 – Architecture des réseaux FCN-32s, FCN-16s et FCN-8s

gurent. Pour obtenir FCN-16s, le réseau FCN-32s est modifié en reprenant la sortie de *pool5*, en la déconvoluant pour la faire correspondre à la sortie de *pool4* et en additionnant ces deux sorties pour former un nouveau résultat. Enfin, une déconvolution avec un pas de 1/16 permet d'obtenir une nouvelle segmentation. Notons qu'avant d'être combinée à la sortie de *pool5*, la sortie de *pool4* subit une nouvelle convolution 1×1 . FCN-8s fonctionne exactement selon le même principe. Tout comme FCN-16s reprend le résultat de FCN-32s, FCN-8S reprend le résultat de FCN-16s avant déconvolution pour le combiner à la sortie de *pool3*. Une fois les images additionnées, une déconvolution avec un pas de 1/8 produit la segmentation de FCN-8s.

L'illustration des résultats présentée par les auteurs est donnée dans la Figure 2.11. FCN a été testé avec différentes architectures. Il s'est avéré que l'implémentation avec l'architecture VGG16 a rendu les meilleurs résultats sur la base de cas PASCAL VOC ¹ en 2011

1. PASCAL VOC (Visual Object Classes Challenge) est un concours de traitement d'images dans plu-

et en 2012 [everingham2015pvc] malgré un temps d'entraînement plus conséquent.

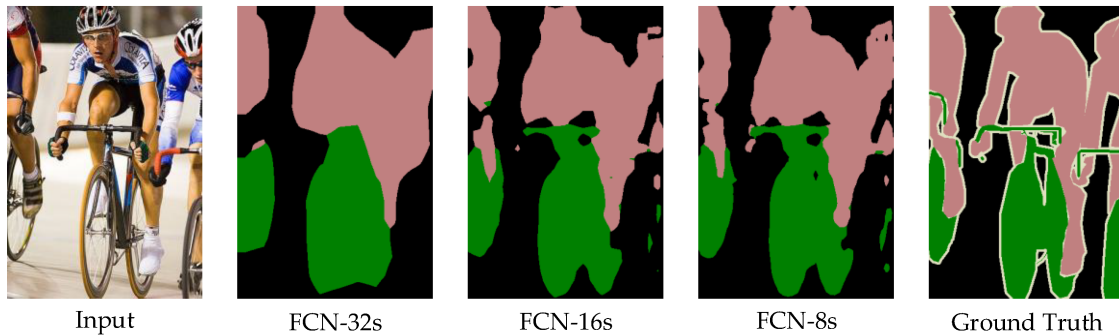


FIGURE 2.11 – Résultats des réseaux FCN-32s, FCN-16s et FCN-8s [long2015fcn]

FCN dans le milieu médical

Austin Ray, de l'Université de Stanford, a utilisé ce réseau pour la segmentation de tumeurs pulmonaires [ray2016fcnMed]. Possédant les vérités terrains de 107 patients, l'auteur a diminué la lourdeur du réseau en enlevant les dernières couches de convolution et a réussi à obtenir de bons résultats avec 7 heures d'entraînement. Pour le coefficient Dice, Ray a obtenu une moyenne de 0.86. Toutefois, les tumeurs rénales étant moins prévisibles et plus diverses que celles du poumon, ce réseau ne semble pas tout à fait adapté pour cette application.

2.4.2/ DECONVOLUTION NETWORK

Proposé par Noh *et al.* en 2015 [noh2015deconvNet], l'architecture de DeconvNet est décomposable en deux parties (cf Figure 2.12) :

- La première est un CNN classique, basé sur l'architecture VGG16, avec une succession de couches de convolution et de couches de *pooling* jusqu'à obtenir un vecteur. Cette partie peut être vue comme une compression de l'information avec une extraction de caractéristiques de plus en plus abstraites. Il ne s'agit pas d'un réseau *fully convolutional* dans la mesure où il conserve l'usage de couches *fully connected*.
- La seconde partie décompresse ensuite l'information en miroir afin de retrouver la taille d'origine des images par une succession de couches de *unpooling* et de déconvolution.

Pour obtenir de meilleurs résultats, Noh *et al.* ont proposé également une variante de DeconvNet : EDeconvNet. Il combine les résultats de DeconvNet et FCN-8s en effectuant la moyenne de leurs images de sortie pour obtenir la segmentation finale. Il peut être réa-

sieurs domaines (classification, détection, segmentation) fournissant une base de données d'images *open source* à utiliser pour le concours.

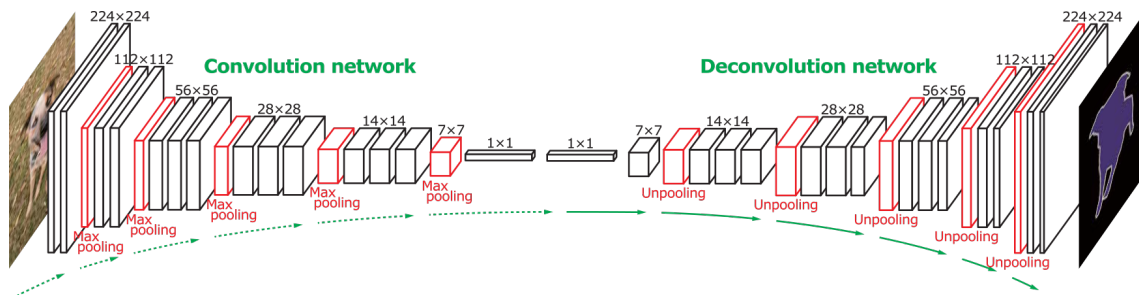


FIGURE 2.12 – Architecture du réseau DeconvNet [noh2015deconvNet]

lisé à la sortie du réseau un *Conditional Random Field*¹ [koltun2012fullyCrf] sur l'image calculée afin d'obtenir une segmentation encore plus précise.

L'entraînement de DeconvNet diffère d'un entraînement classique et il s'effectue en deux étapes. La première est l'entraînement du réseau sur des exemples simples, pour entraîner grossièrement le réseau, et la deuxième est l'affinement de celui-ci en envoyant les images dont les segmentations sont plus complexes. Testé sur la base de données PASCAL VOC 2012, le réseau a obtenu de bons résultats, dépassant ceux de FCN sur les mêmes données.

2.4.3/ DECOUPLED NETWORK

Suite aux travaux sur DeconvNet, les mêmes auteurs ont conçu le réseau *Decoupled Network* (DecoupledNet) [hong2015decoupledNet] en 2015. Il est découpé en trois parties : un réseau de classification (convolutif), un réseau de segmentation (déconvolutif) et des couches intermédiaires, appelées *bridges* ou ponts, permettant la communication entre les deux (cf Figure 2.13).

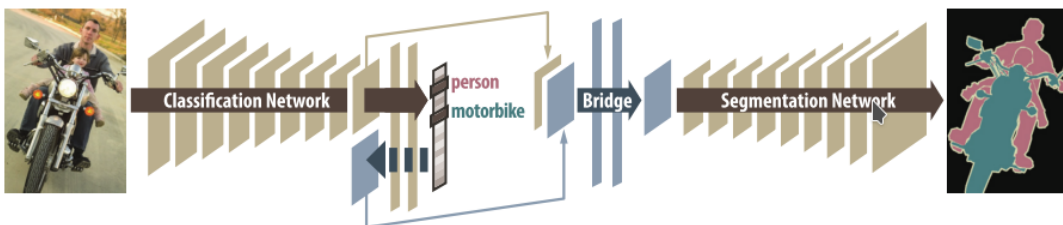


FIGURE 2.13 – Architecture du réseau DecoupledNet [hong2015decoupledNet]

La particularité de ce réseau est de détecter dans l'image le type des objets présents avant d'effectuer une segmentation binaire (*background/foreground*) pour chacun d'entre eux. Les différentes cartes de probabilités en sortie sont ensuite fusionnées pour produire la segmentation finale. La fusion se fait en attribuant à chaque pixel le label ayant le score (\approx probabilité) le plus élevé.

1. Les *Conditional Random Field* (CRF) sont des modèles statistiques utilisés pour la reconnaissance de forme

2.4.4/ SEGNET

SegNet [badri2015segnet] est un autre réseau proposé en 2016 pour la segmentation multiple de scènes en temps réel. Il est plus ou moins semblable à DeconvNet mais les couches *fully connected* ont été supprimées de l'architecture VGG16 (cf Figure 2.14) ce qui rend le réseau moins profond et plus rapide à entraîner (passant de 134 *Mo* de paramètres pour DeconvNet à 14.7 *Mo*). Il diffère également en n'utilisant pas de déconvolutions, les remplaçant par des convolutions simples.

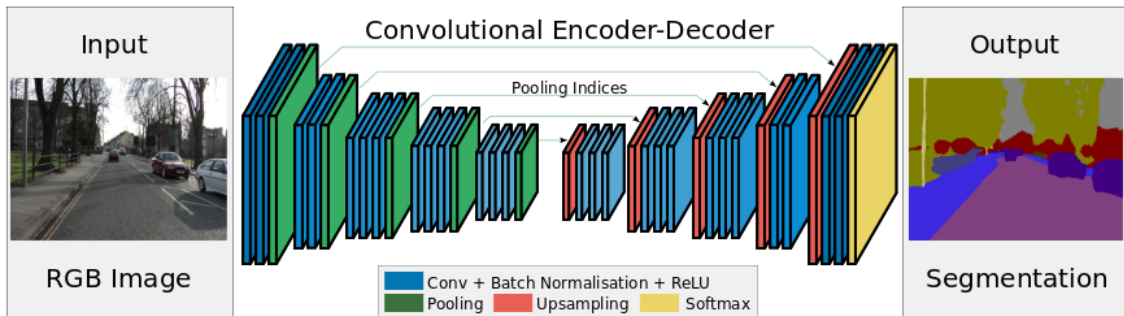


FIGURE 2.14 – Architecture du réseau SegNet [badri2015segnet]

La base d'images de SegNet, CamVid¹ [brostow2009camVid], n'est pas très volumineuse (367 images pour l'entraînement et 233 pour le test), mais elle est suffisante pour obtenir des résultats corrects sur la détection de 11 classes différentes. Pour l'entraînement, l'ensemble de la base d'images est mélangé et des mini-lots de 12 images sont choisis pour entraîner le réseau graduellement sur quelques images à la fois. Les résultats sont concluants pour les objets de tailles importantes en premier plan, mais le réseau rencontre des difficultés pour les objets plus petits et complexes en arrière-plan. L'usage en temps-réel a contraint les auteurs à concevoir un réseau moins profond afin de parvenir à une exécution plus rapide. Cette concession impacte la précision des segmentations.

SegNet dans le milieu médical

Alexander Kalinovsky *et al.* ont appliqué SegNet pour la segmentation d'images de radiologies pulmonaires [kalinovsky2016segnetMed]. Les tests ont été effectués avec une base de données comprenant 354 images, dont 107 avec des nodules tuberculeux, provenant de différents pays et appareils pour permettre un résultat plus généralisable. Les coefficients Dice obtenus se situent entre 92.6% et 97.4% ce qui laisse entrevoir de très bons résultats même si la segmentation pulmonaire est moins complexe que la segmentation de reins (pathologiques de surcroît). En effet, les poumons sont facilement visualisables sur les clichés radiologiques, avec des intensités significatives.

2.4.5/ U-NET

U-Net est un réseau créé en 2015 par Ronneberger *et al.* qui est dédié à la segmentation de cellules dans le domaine de la biologie [ronneberger2015unet]. Le système consiste

1. CamVid est une base de vidéos de voies de circulation du point de vue d'un automobiliste.

en la capture du contexte général des premières couches et en leur concaténation avec des couches plus lointaines qui permettent une localisation plus précise, comme pourrait le réaliser FCN-16s et FCN-8s, mais d'une manière différente. Il s'agit également du premier réseau n'utilisant pas l'architecture VGG16 pour son réseau convolutif, en supprimant certaines convolutions.

Le système est composé d'un premier réseau de neurones convolutif et d'un second déconvolutif dans lequel chaque convolution et sous-échantillonnage de la partie convolutive trouve son inverse dans la partie déconvolutive. Ainsi, le réseau entier est formé de deux parties miroirs l'une de l'autre comme l'illustre la Figure 2.15. De plus, les sur-échantillonnages du réseau déconvolutif sont effectués grâce à des déconvolutions et non des *unpoolings*. Enfin, sa dernière particularité est qu'après chaque déconvolution, le réseau en concatène le résultat avec celui de la convolution correspondante et réalise un *rognage*. L'image finale est cependant plus petite que l'image d'entrée ce qui est dû aux séries de convolutions appliquées sans *padding*, les déconvolutions ne compensant que les étapes de sous-échantillonnages.

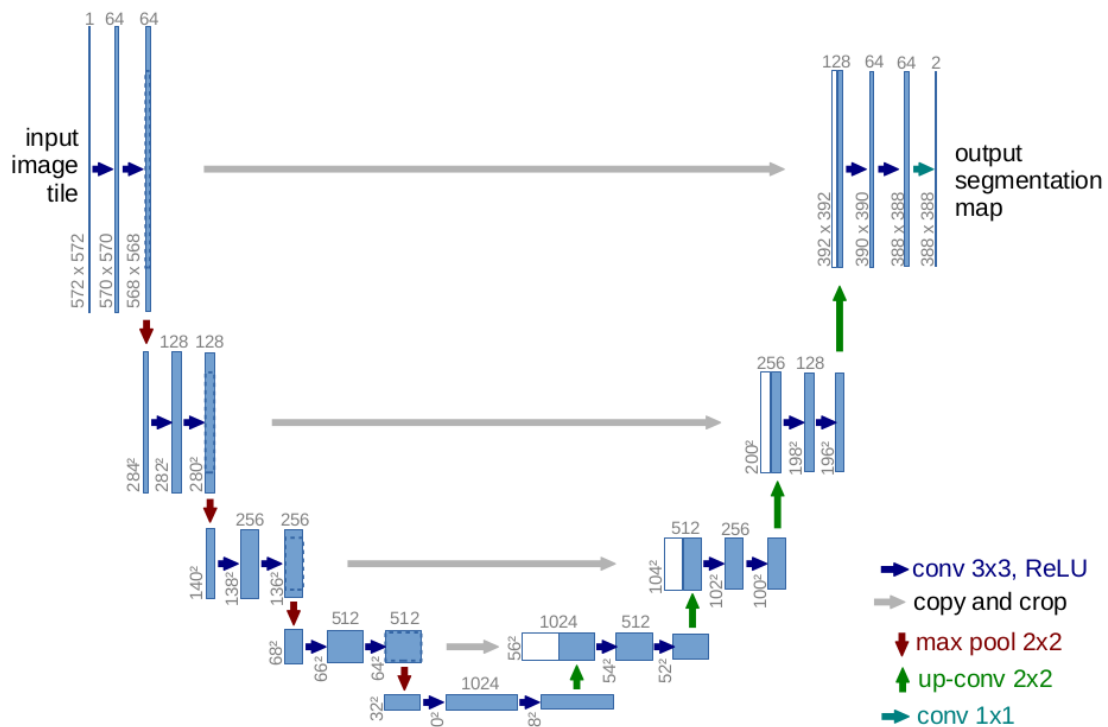


FIGURE 2.15 – Architecture du réseau U-Net [ronneberger2015unet]

Le réseau, entraîné sur 30 images de microscopie électronique, a remporté le *cell tracking challenge 2015* organisé au cours de *ISBI* (International Symposium on Biomedical Imaging) qui est un concours sur la segmentation et le suivi des cellules.

U-Net dans la segmentation du foie

Ce réseau a inspiré Patrick Ferdinand Christ *et al.* dans la création d'un nouveau sys-

tème permettant de segmenter le foie et ses lésions en 2016 [christ2016unetMed]. La segmentation s'effectue en plusieurs étapes. Tout d'abord, le réseau U-Net est appliqué aux images médicales pour la segmentation du foie, puis un autre réseau U-Net est appliqué pour segmenter ces lésions. Il est également ajouté en fin de traitement un CRF qui permet la construction en 3D de la segmentation (*3D CRF*). Le réseau a été entraîné avec la méthode de validation croisée¹ sur 15 volumes hépatiques, provenant de la base 3D-IRCADb². Les résultats sont encourageants avec un coefficient Dice moyen de 94%.

U-Net dans la segmentation d'images médicales en 3D

Depuis l'apparition de U-Net, ce dernier a été utilisé dans la segmentation en 3D d'images médicales, permettant d'avoir un aperçu concret de la zone segmentée en remplaçant toutes les opérations 2D du réseau (Convolution, Pooling, ...) en opérations 3D. C'est le cas de Ciçek *et al.* qui utilisent U-Net pour la segmentation des reins de *Xenopus* (sorte d'amphibien) [ciccek20163dUnet]. Leur réseau peut être utilisé de deux manières :

- La première consiste à effectuer un apprentissage semi-automatique au cours duquel l'utilisateur annoté certaines images 2D dans le volume à segmenter. Le réseau apprend alors à partir de ces annotations pour fournir la segmentation de toutes les coupes du volume.
- La seconde solution consiste à réaliser une configuration entièrement automatisée dans laquelle le réseau s'entraîne sur un jeu de données annoté pour réaliser la segmentation automatique du volume. Les résultats obtenus sont concluants, surtout avec l'utilisation de la méthode de segmentation semi-automatique avec un coefficient IoU moyen de 86.3% contre 70.4% pour la méthode de segmentation automatique. Milletari *et al.* ont également utilisé U-Net pour la segmentation 3D de la prostate [milletari2016unet] et obtenu un Dice moyen de 86.9%. Enfin, H-DenseUNet est une autre version du réseau de Ronneberger *et al.* [li2018hdenseunet]. Il consiste en un système hybride de segmentations en 2D et en 3D afin d'exploiter la topologie du volume tout en limitant la complexité du système impactant les calculs (temps d'apprentissage).

2.4.6/ W-NET

Les réseaux de neurones requièrent la plupart du temps un entraînement se basant sur l'utilisation massive de données étiquetées (apprentissage supervisé). L'étiquetage de ces données est une tâche très chronophage, qui peut nécessiter de surcroît des compétences particulières suivant le domaine d'application (compétences en radiologie pour une application médicale par exemple). Cette étape représente aujourd'hui la principale contrainte des réseaux de neurones et limite leur utilisation ainsi que leur efficacité.

Xia *et al.* se basent sur le réseau U-Net pour proposer une variante non supervisée de ce CNN en 2017 [xia2017wnet] afin d'éliminer la nécessité d'étiqueter en amont les données

1. La validation croisée est un processus qui permet d'entraîner et de tester un réseau en décomposant le jeu de données en sous-parties, ce qui permet de tester une approche avec un jeu de données limité.

2. Base de données contenant 20 volumes scanner (CT-scans) en temps vasculaire provenant de différents hôpitaux européens et réalisés avec différents scanners.

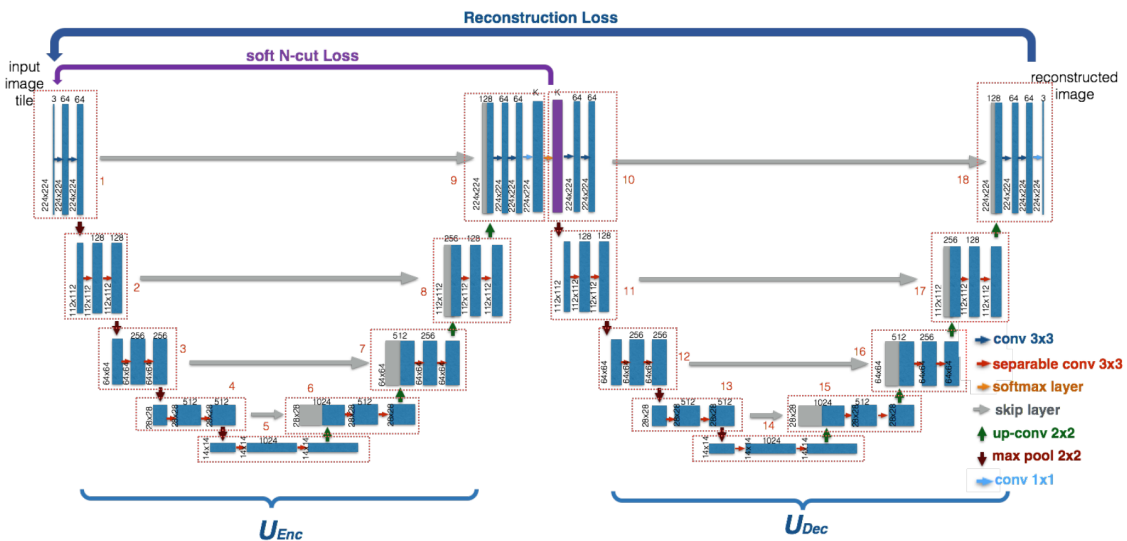


FIGURE 2.16 – Architecture du réseau W-Net [xia2017wnet]

d'entraînement. L'architecture, illustrée dans la Figure 2.16, se présente comme un *auto-encodeur*.

Elle est composée de deux parties successives :

- La première est similaire en tous points au réseau U-Net et possède les mêmes caractéristiques pour encoder l'image.
- La seconde consiste en un décodage en réutilisant U-Net pour la reconstruire. W-Net possède 46 couches de convolution réparties en 18 modules. Chaque module est composée de 2 couches de convolution, suivi par une couche ReLu. L'entraînement consiste à calculer et à minimiser simultanément deux fonctions pour mettre à jour les paramètres respectivement de l'encodeur (U_{Enc}) et de W-Net dans son ensemble. La première fonction *soft N-cut Loss* est le résultat d'un *normalized cut*, une technique de segmentation classique en imagerie. La seconde fonction, *Reconstruction Loss*, est l'erreur entre l'entrée du réseau (image initiale) et sa sortie (image reconstruite). À chaque itération de l'entraînement, les paramètres de U_{Enc} sont tout d'abord mis à jour en minimisant *soft N-cut Loss*, puis les paramètres de W-Net le sont ensuite en minimisant *Reconstruction Loss*. À la fin de l'entraînement, U_{Enc} forme le réseau de segmentation. Les auteurs ont recours à un double post-traitement en utilisant des CRF et une segmentation hiérarchique.

Le *normalized cut* est une technique de segmentation sensible au bruit et nécessitant des images fortement contrastées pour fonctionner correctement. W-Net n'est donc pas très approprié pour une application médicale dans laquelle les images sont souvent bruitées (et peu contrastées dans le cas de clichés tomodensitométriques).

2.4.7/ DEEPMEDIC

Si beaucoup de réseaux de neurones effectuent des segmentations dans des espaces en deux dimensions, cette approche n'est pas nécessairement la plus efficace dans le

domaine de l'imagerie médicale dans lequel les images représentent généralement des volumes (sous la forme d'une série d'images composant un examen par exemple). Dans ce cas de figure, il peut être intéressant d'exploiter l'information donnée par la topologie en trois dimensions pour calculer la segmentation, plutôt que de traiter chaque image indépendamment les unes des autres. Des CNN effectuant des convolutions directement en 3D sont tout à fait réalisables mais cela se heurte à une forte limitation autant en termes de temps de calcul (les paramètres sont beaucoup plus nombreux) qu'en termes de mémoire (l'information image prenant beaucoup plus d'espace).

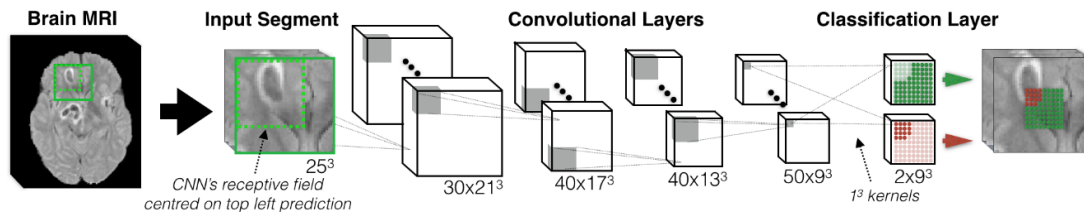


FIGURE 2.17 – Architecture basique du réseau DeepMedic [kamnitsas2017deepMedic]

Pour pallier ce problème, Kamnitsas *et al.* [kamnitsas2017deepMedic] ont proposé en 2017 leur architecture *DeepMedic* dédiée à la segmentation de tumeurs cérébrales sur des clichés IRM dont la Figure 2.17 en illustre le principe de base. Ce réseau possède deux particularités :

- La première est qu'il ne propose pas de couche de sur-échantillonnage pour retrouver les dimensions initiales. La conséquence est que la carte de segmentation en sortie du réseau ne représente que la partie centrale de l'image en entrée.
- La seconde particularité est qu'il ne prend pas en entrée la totalité du volume image, mais seulement un fragment, que les auteurs nomment « *segment* » ou encore « *patch* ». Le calcul de la segmentation complète se fait par la segmentation de multiples sous-images afin d'éviter tout problème de mémoire. Le nombre limité de couches de convolution permet quant à lui de faciliter l'entraînement du réseau en limitant le nombre de paramètres du modèle.

L'information contextuelle joue un rôle très important pour la classification de voxels, qui autrement seraient très similaires. Exploiter efficacement cette information est donc un point central à traiter pour la segmentation par CNN.

Plus le champ de réception du réseau est important, et plus cette quantité d'informations sera importante en contrepartie d'une augmentation du temps de calcul et de la mémoire requise. Kamnitsas *et al.* ont amélioré l'idée de base de leur architecture pour permettre d'exploiter l'information contextuelle à deux niveaux différents, à la fois une information locale et une information plus large, tout en limitant ces inconvénients. Ceci reprend l'idée derrière l'architecture FCN [long2015fcn] pour produire les segmentations FCN16s et FCN8s ou encore celle de U-Net [ronneberger2015unet]. *DeepMedic* prend ainsi la forme de deux séries de convolution en parallèle comme illustré sur la Figure 2.18. Le rôle de la première est de capturer l'information locale pour permettre une segmentation détaillée et précise. À l'inverse, le second chemin permet de capturer une information plus générale comme la localisation de la tumeur au sein du cerveau. Une étape de sur-échantillonnage a été ajoutée pour que la sortie de ce second chemin corresponde à celle du premier.

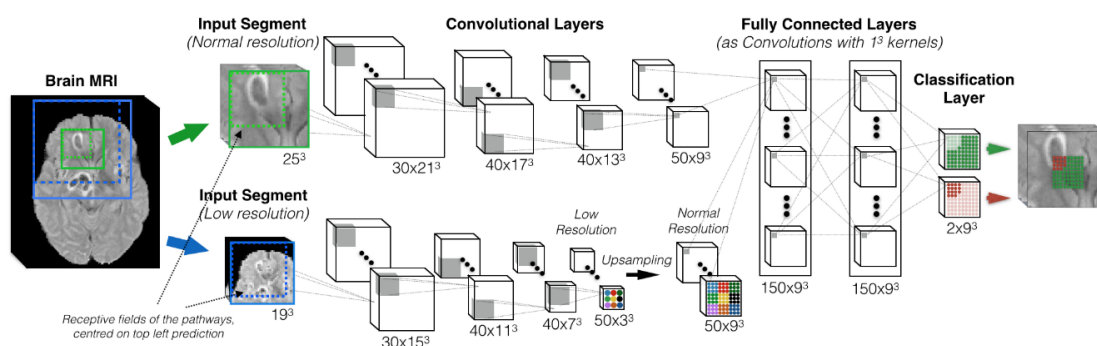


FIGURE 2.18 – Architecture du réseau DeepMedic [kamnitsas2017deepMedic]

Un post-traitement en 3D avec des CRF (*Conditionnal Random Field*) *fully connected* est réalisé sur la segmentation en sortie du réseau pour en améliorer la précision. En effet, les réseaux de neurones donnent généralement de très bons résultats pour localiser les objets d'intérêt dans l'image mais ils ont quelques difficultés à en délimiter clairement les bordures. Les CRF sont classiquement utilisés afin de lisser des images bruitées ([kohli2009crf]). Ici, l'objectif est au contraire d'accentuer les détails de la segmentation, ce que permet de faire les CRF *fully connected* ([krahenbuhl2011fullyConnectedCrff]).

2.4.8/ DEEPLAB

Lors du passage d'une image dans un réseau de convolution, la résolution spatiale des *feature maps* décroît par l'action combinée des sous-échantillonnages (*pooling*) et des convolutions. L'un des moyens d'éviter cet écueil, pour produire une carte de segmentation à haute résolution en sortie du réseau, est d'avoir recours à des couches de déconvolution ([long2015fcn], [ronneberger2015unet]) au prix d'un coup en calcul et en mémoire pour apprendre les paramètres supplémentaires.

Chen *et al.* [chen2018deeplab] proposent dans leur réseau *DeepLab* une alternative utilisant des convolutions à trous. Ce type de convolutions, aussi nommées convolutions dilatées, se base sur un algorithme (appelé « *l'algorithme à trous* ») utilisés en traitement du signal dans le domaine des ondelettes. La démarche consiste à insérer des valeurs nulles dans le noyau de convolution afin d'augmenter artificiellement la taille du champ de réception. Le principal avantage est que la taille du champ de réception est augmentée avec un coût en calcul constant (il n'y a pas plus de paramètres à déterminer durant la phase d'entraînement dans la mesure où seules des valeurs nulles sont ajoutées). Ceci permet au réseau d'être plus optimisé et plus facile à entraîner. De plus, les convolutions à trous peuvent théoriquement éviter une décimation du signal en remplaçant une ou plusieurs étapes de sous-échantillonnages. Cependant, en raison du coût important en mémoire que cela entraîne, les auteurs de *DeepLab* ont choisi une approche hybride dans laquelle les convolutions à trous permettent de limiter la perte de résolution et donc le facteur de sur-échantillonnage à la fin du réseau sans s'affranchir de cet étape.

La Figure 2.19 illustre le principe générale du réseau *DeepLab*. L'image initiale est tout d'abord injectée en entrée du réseau de neurones appliquant en partie des convolutions à trous. Puis, la carte de segmentation résultante subit un sur-échantillonnage bilinéaire pour retrouver les dimensions de l'image en entrée. Enfin, une dernière étape consiste à appliquer un post-traitement avec des CRF *fully connected* pour en améliorer la précision.

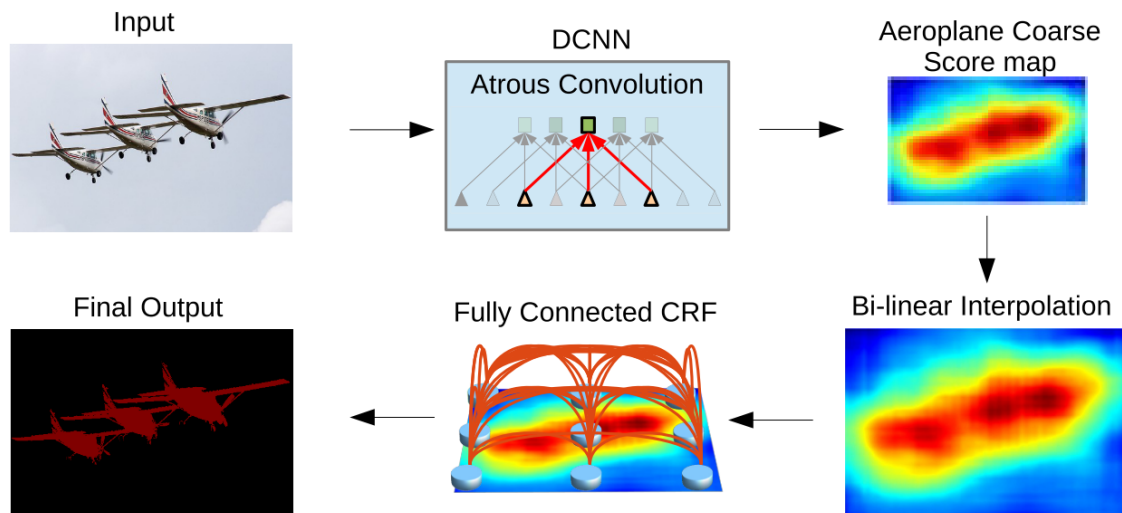


FIGURE 2.19 – Illustration du fonctionnement du réseau DeepLab [chen2018deeplab]

Les auteurs de DeepLab ont également travaillé sur une autre problématique qui est l'extraction d'informations à des échelles différentes dans des réseaux de neurones. Plusieurs approches sont possibles pour capturer et fusionner cette information à de multiples résolutions :

- La première est celle utilisée par les auteurs de FCN-8s ([long2015fcn]) et consiste à sur-échantillonner et à fusionner la sortie de plusieurs couches de *pooling* afin de calculer une segmentation plus précise.
- La deuxième est d'utiliser en parallèle plusieurs réseaux de neurones sur la même image à différentes résolutions ([chen2016multiScale]).
- La troisième est le recours à la stratégie d'une *pyramid pooling* ([he2014pyramidPooling]). Cette stratégie consiste à capturer le contexte à de multiples résolutions en concaténant le résultat de différents *pooling* (ayant chacun un facteur de sous-échantillonnage différent).

Chen *et al.* ont implémenté et testé les deux dernières. D'une part, ils ont extrait les cartes de scores provenant de différentes versions redimensionnées d'une même image injectée en parallèle dans différentes branches d'un même CNN partageant les mêmes paramètres. Ces cartes de scores sont ensuite sur-échantillonnées par interpolation bilinéaire et elles sont fusionnées en gardant uniquement le score le plus élevé pour chaque pixel. D'autre part, les auteurs ont adapté la stratégie de la *pyramid pooling* en remplaçant les *poolings* par des convolutions à trous avec différents facteurs de dilatations pour donner une *Atrous Spatial Pyramid Pooling* (ASPP) comme illustré par la Figure 2.20. Les sorties de chacune des convolutions sont ensuite traitées dans des branches différentes du réseau puis fusionnées de la même manière que dans l'approche précédente.

Si la première version de DeepLab était basée sur l'architecture *VGG16*, les auteurs ont choisi d'utiliser par la suite *ResNet-101* ([he2016resNet]). ResNet est un réseau très connu pour permettre une grande profondeur (un grand nombre de couches) sans perte du gradient lors de l'apprentissage grâce à son système de saut. Son principe est de copier le gradient d'une couche n plus loin dans le réseau en l'additionnant au gradient

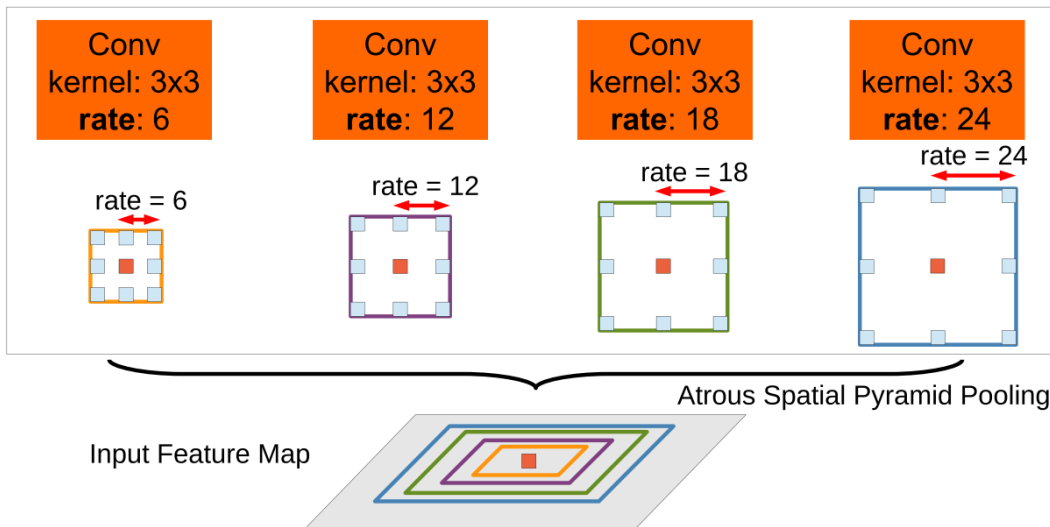


FIGURE 2.20 – Principe de l’Atrous Spatial Pyramid Pooling (ASPP) [chen2018deeplab]

d’une couche située en aval. Ceci permet d’éviter une diminution trop importante de celui-ci, dommageable lors de l’apprentissage. Au final, l’architecture du réseau a pris la forme illustrée par la Figure 2.21.

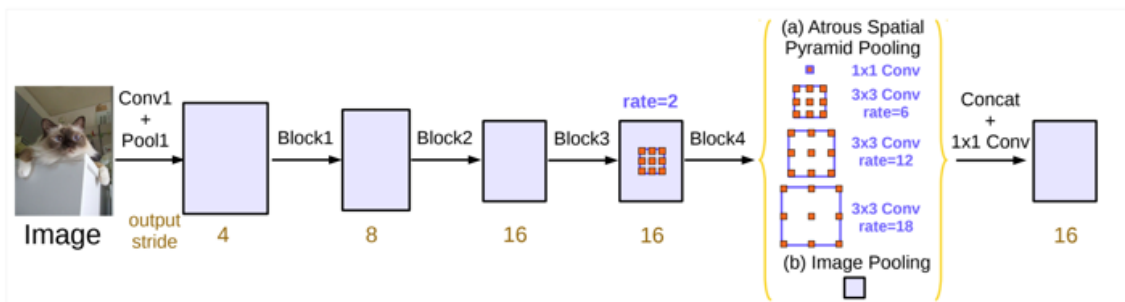


FIGURE 2.21 – Architecture du réseau DeepLab [chen2018deeplab]

L’architecture est constituée de 4 *Blocks*. Chaque *Block* contient un nombre différent de *Residual Units*. Un *Residual Unit* consiste en deux convolutions avec un noyau de taille 3x3, chacune suivie par une couche de *Batch Normalization* (BN) et une couche ReLU. *Block 4* est modifié par rapport à l’architecture *ResNet* initiale avec des convolutions remplacées par des convolutions à trous.

2.4.9/ COMPARATIF DES RÉSEAUX

Le Tableau 2.1 synthétise les principales caractéristiques des différents réseaux de neurones convolutifs (CNN) mentionnés dans les sections précédentes. La seconde colonne indique le contexte d’application pour chacun des réseaux, tandis que la troisième renseigne des informations à propos du type de données utilisées pour l’apprentissage et les tests (nom de la base de données, type et taille, nombre de classes).

Nous souhaitons trouver une méthode qui permette d’obtenir des segmentations les plus

	Contexte	Ensemble d'apprentissage
FCN-8s	Objets communs	Photos RGB Pascal VOC 2012 : 1464 images, 20 classes
DeconvNet	Objets communs	Photos RGB Pascal VOC 2012 : 1464 images, 20 classes
DecoupledNet	Objets communs	Photos RGB Pascal VOC 2012 : 1464 images, 20 classes
U-Net	Images de cellules	Images de microscopie PhC-U373 : 35 images, 2 classes DIC-HeLa : 20 images, 2 classes
SegNet	Images de voix de circulation	Images RGB Cam Vid : 367 images, 11 classes
E-Net	Images de voix de circulation	Images RGB Cam Vid : 367 images, 11 classes
3D U-Net	Images de reins (xenopus)	Images de microscopie confocale 77 images, 4 classes
DeepMedic	Tumeurs cérébrales	Images IRM cérébrales BraTS : 274 cases (240 x 240 x 155), 4 classes
DeepLabV3	Objets communs	Photos RGB Pascal VOC 2012 : 1464 images, 20 classes
W-Net	Objets communs	Photos RGB Pascal VOC 2012 : 1464 images, 20 classes
H-DenseUNet	Tumeurs hépatiques	CT-Scans LiTS : 131 cas, 3 classes

TABLE 2.1 – Caractéristiques des CNN issus de la littérature

proches possible de celles réalisées par les médecins sur des CT scan de reins tumoraux. L'état de l'art nous a permis de comparer les travaux existants dans notre domaine d'application.

FCN-8s et U-Net semblent être les réseaux les plus fréquemment utilisés dans la littérature avec leurs multiples déclinaisons. DeepMedic est en soit un bon candidat : conçu pour une application médicale, il a été évalué sur la segmentation de tumeurs cérébrales avec succès. Ses bons résultats sont toutefois à relativiser par rapport au type d'images utilisé. Les images IRM sont des images offrant naturellement une bonne résolution en contraste et les tumeurs cérébrales sont parmi les plus simples à visualiser et à délimiter. Le seul réseau testé sur des images scanner est H-DenseUnet, qui est lui-même basé sur le réseau U-Net. Les auteurs l'ont utilisé pour segmenter des tumeurs hépatiques avec un large ensemble de données (131 patients), qui dépasse donc très largement les données à notre disposition, ce qui nous empêche de préjuger du résultat pour notre application.

De plus, les tumeurs hépatiques restent plus faciles à segmenter, compte tenu de leur forme plus prévisible et distincte que pour les néphroblastomes. Enfin, tout comme DeepMedic, H-DenseUnet a une approche 3D pour réaliser les segmentations, ce qui est généralement plus efficace qu'une simple approche 2D pour segmenter un volume [li2018hdenseunet]. L'inconvénient de cette méthode est que l'un des axes de notre travail, et une contrainte due au contexte de nos recherches, est la proposition d'une méthodologie d'entraînement avec un très faible jeu de données indépendamment du réseau utilisé et basée sur un ensemble de coupes 2D à segmenter. Cette méthodologie est

donc incompatible avec une segmentation 3D.

Synthèse du chapitre

Les neurones artificielles sont basés sur une analogie avec le neurone biologique. Le neurone, également appelé perceptron, est une fonction mathématique réalisant une somme pondérée d'une série d'entrées suivant les poids associés à chacune d'elle. Le résultat de cette somme est ensuite additionné à un biais et passé en paramètre d'une fonction d'activation qui en détermine la sortie. En associant entre elles ces unités fonctionnelles que sont les neurones au sein d'un réseau, il est possible de réaliser des processus complexes. Cependant, l'absence d'un moyen automatique pour ajuster leurs paramètres limitait fortement l'utilisation de ces réseaux. À la fin des années 1980, l'élaboration d'un algorithme d'apprentissage automatique (la *rétropropagation*) a ouvert la voie à des traitements plus complexes grâce à des réseaux de grande profondeur. Le traitement d'images fait appel à des réseaux particuliers, dits « convolutifs », capables d'extraire des caractéristiques à partir des valeurs de pixels. Il existe dans la littérature un grand nombre d'architectures différentes dédiées à la segmentation d'images dans des contextes variés. Les plus représentés sont les réseaux FCN (*Fully Convolutional Network*) [long2015fcn] et U-Net [ronneberger2015unet] sur lesquels se basent un grand nombre de variantes.

Dans le chapitre suivant, nous présenterons le fonctionnement d'une autre approche d'intelligence artificielle qui est le Raisonnement à Partir de Cas (RàPC) ainsi que l'état de l'art concernant son utilisation pour la segmentation d'images. Les différentes parties d'un système RàPC y seront décrites ainsi que les problématiques associées. Les différents travaux utilisant un système RàPC pour la segmentation d'images y seront présentés.

LE RAISONNEMENT À PARTIR DE CAS : UNE APPROCHE CONNAISSANCE

Le raisonnement à partir de cas (RàPC) représente une approche orientée connaissance ne se basant pas uniquement sur l'expérience brute (base d'exemples). Il s'agit d'une technique d'Intelligence Artificielle permettant à un système de résoudre un problème nouveau par analogie avec les connaissances théoriques et vécues par le système. Dit autrement, il s'agit de proposer une solution utilisée pour un cas similaire et déjà résolu. Grâce à l'apport d'une connaissance métier, le RàPC est une méthode intéressante en particulier lorsque l'on souhaite concevoir un système automatisé sans parvenir à réunir des données en nombre suffisant pour respecter des conditions de performances en terme de résultats. L'apport d'une connaissance métier et l'existence d'un processus d'adaptation de la solution ancienne pour l'optimiser face à un problème nouveau permet de construire des solutions de plus en plus précises sans nécessiter une base de données initialement conséquente.

Les réseaux de neurones forment une méthode d'intelligence largement plébiscitée dans la littérature actuelle grâce à ses performances, et ce dans des domaines très variés. La principale limite découle du fait que les réseaux de neurones représentent une approche IA orientée en totalité sur l'expérience vécue par le système (le réseau lui-même). Celle-ci doit donc être à la fois suffisamment vaste et représentative des situations potentiellement rencontrées pour garantir des performances adéquates. Il y a donc le recours nécessaire à une base d'apprentissage importante et correctement construite pour couvrir la majeure partie des configurations possibles, afin de réaliser les entraînements dans de bonnes conditions. Ainsi, l'apport de la connaissance par le Raisonnement à Partir de Cas permet de contourner cette limitation, et c'est ce que nous étudions dans ce chapitre.

3.1/ GÉNÉRALITÉS

3.1.1/ HISTORIQUE

Les précurseurs des systèmes de Raisonnement à Partir de Cas (RàPC) sont les systèmes experts basés sur des règles (également appelés « moteurs de règles ») qui furent très utilisés dans les années 70 et 80. Ces systèmes consistent généralement en des déductions ou des choix réalisés à partir d'un ensemble de règles. Dans le milieu médical, des systèmes experts ont, par exemple, été conçus pour apporter une aide au diagnostic

[tsumura1986expertSystem].

Les systèmes experts possèdent cependant plusieurs limites mises en évidence par Roger Schank [schank1987memoryExpertSystem] qui limitent leurs performances en situation réelle. Tout d'abord, ces systèmes ne peuvent traiter que des situations explicitement prévues par les développeurs dans leur moteur de règles. De plus, ils n'ont aucune faculté d'apprentissage. Leurs performances initiales sont définitives sans une intervention humaine. Enfin, une troisième limite, qui découle des deux premières, est la complexité et le caractère fortement chronophage de la construction d'une base de connaissances (ensemble de règles) car elle doit envisager toutes les situations potentielles et les formaliser sous forme de règles.

Les systèmes de Raisonement à Partir de Cas sont apparus comme une alternative pour tenter de pallier ces défauts, sans parvenir à les éliminer. Le RàPC trouve son origine dans les travaux sur les sciences humaines d'une part de Marvin Minsky [minsky1975framework], et d'autre part, dans ceux de Roger Shank [schank1977scripts]. M. Minsky définit en 1974 que la pensée est organisée en schémas ou cadres (*frames* en anglais) dans lesquels la connaissance serait subdivisée en structures de données. La liaison entre différents cadres permet de donner naissance à une idée. R. Schank reprend cette hypothèse en 1977 et propose une organisation hiérarchique des cadres qu'il nomme *Memory Organisation Packets* (MOP) pour imiter le raisonnement humain. Si un MOP décrit une situation similaire à celle nouvellement rencontrée, l'expérience est remémorée et l'individu peut essayer d'appliquer la même solution. On retrouve ici l'une des bases du processus de RàPC qui sera développé par la suite. Ceci se différencie des systèmes experts en s'affranchissant d'un ensemble de règles et en ré-appliquant des schémas (R. Shank parle de *scripts*) ayant fonctionné par le passé. En 1987, R. Shank conceptualise ainsi un nouveau modèle de raisonnement basé sur la mémoire [schank1987memory] qui évoluera vers le raisonnement à partir de cas.

La première implémentation d'un système de RàPC est réalisée par Janet Kolodner en 1983 avec le système CYRUS [kolodner1983cyrus]. Ce dernier stocke les différents événements vécus par deux anciens secrétaires d'états, Cyrus Vance et Edmund Muskie, et il peut répondre à des questions posées en anglais les concernant. Les travaux de J. Kolodner se concentrent en particulier avec CYRUS sur l'organisation de la mémoire épisodique. Cependant, le terme de Raisonement à Partir de Cas (*Case-Based Reasoning*) n'apparaît que plus tard dans l'ouvrage de C. Riesbeck et R. Shank *Inside Case-Based Reasoning* en 1989 [riesbeck1989cbr].

3.1.2/ LE RAISONNEMENT PAR ANALOGIE ET LES BASES DU RÀPC

Le Raisonement à Partir de Cas repose sur une analogie entre une expérience passée et une expérience présente pour parvenir à la résolution d'une situation et repose sur un certain nombre de postulats (sur lesquels nous reviendrons en détails dans la section 3.1.2.2).

3.1.2.1/ LE RAISONNEMENT PAR ANALOGIE

L'analogie est un processus de pensée consistant à associer deux concepts ou idées possédant des propriétés communes (par exemple : « *une route qui serpente* » est une

analogie entre « une route sinueuse » et « un serpent » ayant une propriété de forme commune). L'analogie proportionnelle est une analogie particulière entre 4 concepts A, B, C et D dans laquelle les rapports entre A et B, d'une part, et C et D, d'autre part, sont comparés (présentée par E. Chouraqui en 1986 [chouraqui1986analogie]). Cette analogie est notée $A : B :: C : D$ et se lit « A est à B ce que C est à D ». Elle forme la base du carré d'analogie illustré dans la Figure 3.1.

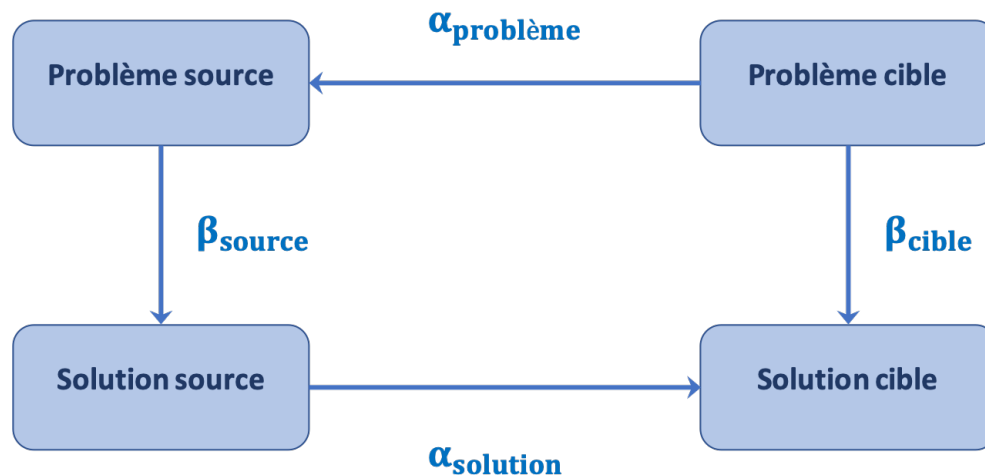


FIGURE 3.1 – Le carré d'analogie appliqué au RàPC [lieber2008hdr]

Transposé à la résolution d'un problème, le raisonnement par analogie permet de résoudre un problème *cible* à partir d'un problème *source* analogue. Il se base sur l'hypothèse qu'une faible différence $\alpha_{probleme}$ entre les deux problèmes correspond à une faible différence $\alpha_{solution}$ entre les deux solutions associées telle qu'illustrée dans le carré de l'analogie. La conséquence est que, pour deux problèmes suffisamment proches, la relation β_{source} connue (permettant de lier les descripteurs du problème *source* et les descripteurs de la solution *source*) peut être réutilisée comme approximation de la relation β_{cible} inconnue, ce qui revient à réutiliser la solution *source* pour le problème *cible*. En pratique, la différence $\alpha_{probleme}$ peut rester significative et il est alors nécessaire de passer par une étape d'adaptation de la solution *source* pour compenser la différence $\alpha_{solution}$.

La différence entre un raisonnement par analogie et un raisonnement à partir de cas (RàPC) concerne uniquement la nature des entités comparées. Une analogie consiste à remarquer une similitude entre deux problèmes de deux natures différentes alors qu'un RàPC remarque cette même similitude entre des problèmes d'un même domaine. Les deux termes sont d'ailleurs parfois utilisés en tant que synonymes. L'analogie représente une problématique de recherche importante et en constante évolution [lieber2008hdr, devignes2018analogy].

3.1.2.2/ DES POSTULATS POUR UN SYSTÈME DE RÀPC

Le principe du raisonnement à partir de cas s'appuie sur remémoration. Des solutions ayant permis de résoudre un problème ancien sont réutilisées afin de résoudre un problème nouveau similaire. Ce principe repose sur une série de quatre hypothèses fortes à propos du monde telles qu'elles sont listées par J. Kolodner en 1996 [kolodner1996principlesCBR].

L'hypothèse de régularité

Les mêmes actions réalisées sous des conditions similaires entraînent des effets similaires. Cette hypothèse est importante pour rendre la démarche du RàPC, en elle-même, valide. La remémoration d'une solution consiste à évaluer le contexte comme similaire puis à appliquer la solution. Si les résultats d'une solution changent du tout au tout malgré des conditions proches, le RàPC ne peut pas parvenir à résoudre des problèmes nouveaux.

L'hypothèse de typicité

Les expériences tendent à se répéter de manières similaires. Sans cette hypothèse, le système de RàPC devrait faire face en permanence à des situations nouvelles atypiques nécessitant une approche originale qui ne serait pas stockée dans sa base de connaissance. Ainsi, la remémoration d'une solution ancienne perdrait tout sens.

L'hypothèse de cohérence

De faibles changements dans une situation entraînent la nécessité de faibles changements dans la solution. Lors du processus de remémoration, il n'est pas possible (ou très peu probable) de retrouver une situation passée identique à la situation présente. Au mieux, les deux seront fortement similaires. Pour que la solution remémorée soit valide, il est nécessaire que l'écart entre celle-ci et la solution idéale soit comparable à la différence entre les deux problèmes.

L'hypothèse d'adaptabilité

Lorsqu'il y a une répétition de situations, les différences tendent à être faibles et le système est capable de les compenser. À partir du moment où les situations ne sont pas identiques, il peut être nécessaire que le système modifie la solution avant de l'appliquer pour obtenir de bons résultats. Cependant, il ne pourra probablement pas compenser des différences trop importantes.

3.2/ LE FONCTIONNEMENT DU RÀPC

3.2.1/ LES DIFFÉRENTS TYPES DE CONNAISSANCES

Tandis qu'un système basé sur la connaissance est généralement organisé en différents modules (cf section 3.2.2 sur la description d'un système de RàPC), la connaissance elle-même peut être organisée de manière similaire, dans ce que M. Richter nomme « *des conteneurs* » [richter2003fourKnowledge]. La représentation de cette connaissance se fait en utilisant différents éléments de descriptions, les « *conteneurs* » qui permettent de décrire le processus de connaissance. Là où ces éléments pris séparément ne permettent pas une restitution complète de cette connaissance, leur somme permet au système de la représenter efficacement et de résoudre un problème. Ces éléments de description peuvent être de natures différentes. Pour l'illustrer, M. Richter cite dans ses travaux l'exemple des programmes logiques dans lequel certains éléments permettent une description de faits et d'autres éléments une description de règles à appliquer. Cette

section a pour objectif de présenter le formalisme associé à la connaissance dans un système de RàPC à travers les différents *conteneurs* que l'on peut y trouver.

Ainsi, M. Richter identifie 4 conteneurs majeurs, pour le RàPC, permettant de représenter le processus de connaissance : le vocabulaire, la mesure de similarité, la base de cas et enfin le processus d'adaptation de la solution (voir Figure 3.2) [richter2003fourKnowledge].

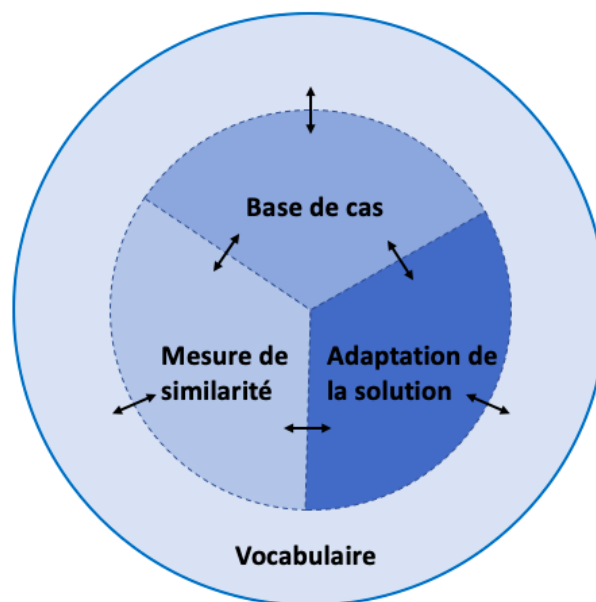


FIGURE 3.2 – Illustrations de l'interdépendance des 4 conteneurs majeurs dans un système de RàPC, issue de [richter2003fourKnowledge]

Si la base de cas est correctement conçue et suffisamment conséquente mais que la mesure de la similarité est mauvaise, alors le cas remémoré (interprété par le système comme étant le plus proche du problème à résoudre) ne permettra pas de résultats probants. À l'inverse, une excellente mesure de similarité associée à une base de cas déficiente n'aboutira pas à la résolution du problème. En parallèle, l'adaptation de la solution dépend beaucoup de la remémoration de la solution du cas source. Si ce cas est mal modélisé dans la base ou bien si la mesure de similarité ne donne pas accès au cas le plus pertinent, l'adaptation de la connaissance ne pourra pas se faire de manière efficace. Le vocabulaire, utilisé par tous les conteneurs, a donc une influence évidente sur ces derniers. Il y a donc une interdépendance entre les différents conteneurs qui est illustrée par une double flèche dans la Figure 3.2 issue des travaux de M. Richter.

3.2.1.1/ LE VOCABULAIRE

Le point central d'un système basé sur la connaissance est la manière dont la connaissance va être représentée et quels éléments vont être utilisés pour permettre cette représentation (des attributs, ou des fonctions, ou des relations logiques, ...). Le vocabulaire correspond à ces éléments et à la représentation choisie. La plupart des systèmes RàPC utilise une représentation attributs-valeurs dans laquelle chacun des objets contenus dans la base de données possède une valeur par d'attribut choisie dans la modélisation suivant sa sémantique. La Table 3.1 donne un exemple possible d'une telle

Patients \ Attributs	Âge (an)	Masse (Kg)	Taille (cm)	Sexe
Patient 1	23	62	182	1
Patient 2	28	65	171	1
Patient 3	14	55	165	0
Patient 4	46	76	174	1

TABLE 3.1 – Exemple d'une représentation attributs-valeurs

représentation dans un contexte médical. La base y est constituée de différents patients modélisés selon 4 caractéristiques/attributs que sont l'âge, la masse, la taille et le sexe. Chaque attribut est associé à une valeur dépendant d'un choix sur la manière de quantifier ces données (le sexe est, par exemple, représenté par une valeur binaire : 0 pour un homme et 1 pour une femme).

Selon M. Richter [richter2003fourKnowledge], la pertinence d'un ensemble d'attributs peut être évaluée suivant deux critères :

- **La complétude :**

Le choix des attributs doit rendre compte de l'ensemble des propriétés pertinentes pour la résolution du problème cible. Si certaines propriétés importantes ne sont pas représentées, il y a alors un manque de paramètres dans la modélisation.

- **L'efficacité :**

Pour un ensemble d'attributs complet donné, il se peut qu'une relation entre deux attributs soit aussi importante que les attributs eux-mêmes pour une prise de décision (par exemple : l'indice IMC, l'indice de masse corporelle, est calculé suivant le sexe à partir du poids et de la taille). Ces relations sont appelées « attributs virtuels ». Leur adjonction peut améliorer significativement l'efficacité du système et même entraîner la suppression d'autres attributs.

3.2.1.2/ LA MESURE DE SIMILARITÉ

Une mesure de similarité consiste à calculer une valeur rendant compte de la ressemblance entre deux entités (entre deux problèmes dans le contexte du RàPC) dans le but de permettre la remémoration d'une solution utile. L'exigence derrière le principe de similarité est que le plus proche voisin p d'un problème q à résoudre fournit la meilleure solution disponible pour la résolution de q . Dit autrement, la mesure de similarité doit quantifier l'utilité de chaque problème p contenu dans une base relativement à q .

Il s'agit d'une opération mathématique permettant d'associer à un couple de problèmes (p, q) une valeur réelle comprise entre 0 et 1 telle que :

$$sim : p, q \mapsto x \in [0, 1] \quad (1)$$

Plus la valeur de $sim(p, q)$ est grande, plus la similitude entre p et q est importante. Une valeur de 1 indique que les deux problèmes sont identiques. La similarité globale est calculée à partir des similarités locales sur les domaines des différents attributs i , généralement selon une distance avec pondération :

$$sim(p, q) = \sum_{i=1}^n (g_i \times sim_i(p_i, q_i)) \quad (2)$$

où :

- $sim_i(p_i, q_i)$ est la similitude locale entre p_i et q_i ,
- g_i est la i^{ieme} valeur, du vecteur poids g , associée à l'attribut i (souvent telle que $g_i \in [0, 1]$ et $\sum_{i=1}^n g_i = 1$).

Le conteneur « *mesure de similarité* » peut être lui-même divisé en deux sous-conteneurs. Le premier est la similitude locale (relative à un attribut) et le second est, ce que Richter nomme, la fonction d'agrégation permettant de calculer la similitude globale à partir des similitudes locales et correspond en pratique au vecteur poids contrôlant la pondération entre les différents attributs suivant l'importance à leur donner.

3.2.1.3/ LA BASE DE CAS

La base de cas est le conteneur, renfermant les expériences disponibles du système utilisées pour la résolution des nouveaux problèmes rencontrés. Idéalement, elle ne doit contenir que les couples (p, s) où la pertinence de la solution s est maximale pour la résolution de p (uniquement de bonnes solutions).

La base de cas doit également répondre à une double exigence contradictoire :

- Elle doit contenir autant d'expériences pertinentes que possible afin de maximiser la couverture de la connaissance,
- Elle doit être aussi petite que possible afin de minimiser le temps de recherche.

Une base de cas idéale représente donc un compromis entre ces deux objectifs pour maximiser l'efficacité du système, à la fois du point de vue du résultat et du point de vue du temps d'exécution pour la production de ce résultat. Une bonne base de cas est primordiale pour obtenir un système avec une bonne compétence que P. Perner définit comme suit [perner2001cbrForImages] :

Définition

La compétence d'un système de Raisonnement à Partir de Cas (RàPC) est définie comme l'étendue des problèmes cibles que ce système peut résoudre. Elle dépend de propriétés statistiques comme la taille de la base de cas, sa densité ou sa distribution.

L'organisation de la base de cas a une influence importante sur le temps de recherche lors des calculs de similarité [negny2010effective]. Une approche naïve consiste à construire la base sans organisation particulière. On parle alors de *flat base* et une valeur de similarité est calculée entre le problème q à résoudre et tous les problèmes p contenus dans la base afin de sélectionner celui qui est perçu comme étant le plus proche. Cette approche a pour principal défaut d'augmenter de manière linéaire le temps de recherche lorsque la taille de la base augmente. Ce défaut est amplifié lorsque le nombre d'attributs est important. Une organisation hiérarchique lui est généralement préférée car elle offre l'avantage de limiter l'impact du nombre de cas stockés sur la complexité en temps. Un exemple est l'indexation des cas en différentes catégories, qui crée ainsi un niveau hiérarchique supplémentaire afin d'éviter une recherche exhaustive. Ceci permet de déterminer la catégorie la plus similaire avant de s'intéresser plus spécifiquement aux cas qu'elle contient.

Dans la littérature, différentes méthodes pour catégoriser une base ont été définies dont : un modèle bayésien [pal2004cbr], un réseau de neurones [malek2001hybridCbr], un algorithme génétique [bueno2007geneticSim] ou encore un algorithme des *k*-médoides [barioni2008metricAccess] (une variante plus robuste de l'algorithme des *k*-moyennes). Une autre approche communément utilisée en RàPC, et très similaire, est l'usage d'arbre d'indexation comme les B-Trees, les B+-Trees ou encore les R-Trees. Ceci consiste à diviser de manière arborescente l'ensemble de données en sous-ensembles. Les nœuds situés aux extrémités de l'arbre forment les feuilles associées à la classification, tandis que les nœuds intermédiaires possèdent des valeurs (ou des intervalles de valeurs) pour séparer les différents cas.

La figure 3.3 illustre l'exemple particulier d'un *slim-Tree*. Son principe est de répartir les cas en différentes catégories de taille fixe. Le nœud supérieur est associé à un cas considéré comme représentatif de chacun de ses nœuds fils. En partant des feuilles, le cas représentatif du premier ensemble (à gauche) est le cas C1. Le deuxième nœud fils est représenté par le cas C3. Ainsi, le premier nœud intermédiaire a pour valeurs C1 et C3, et son représentant au niveau du nœud racine est de nouveau C1. Il est donc possible de calculer un indice de similarité de manière itérative jusqu'à arriver aux feuilles de l'arbre pour trouver le cas le plus proche.

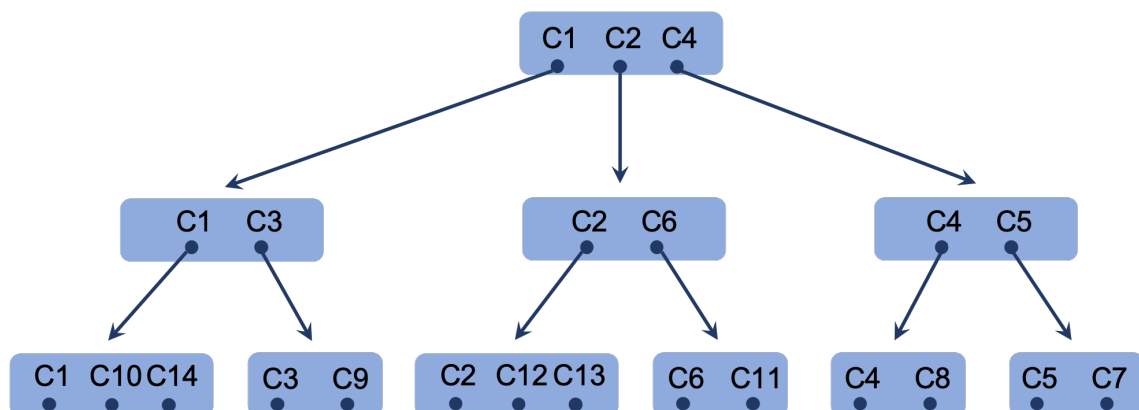


FIGURE 3.3 – Exemple d'utilisation d'un slim-Tree issue de [negny2010effective]

Cependant, une organisation hiérarchique rend plus complexe la maintenance de la base et est réellement intéressante uniquement pour de *grandes* bases de données. La notion de grande base reste *vague* mais notons tout de même, comme nous l'aborderons dans

la suite de ce document, qu'un système de RàPC complet a pour principal avantage de voir sa base de cas croître au fur et à mesure de son utilisation pour augmenter ses performances (apprentissage). Il peut donc être pertinent d'utiliser une organisation hiérarchique même si la base initiale est de taille raisonnable.

3.2.1.4/ L'ADAPTATION DE LA SOLUTION

Lors du calcul de similarité, le cas sélectionné est par définition « *le cas le plus proche* » du problème à résoudre. Ceci implique que non seulement ce cas peut ne pas être identique mais encore qu'il peut également être assez éloigné du problème : par exemple si jamais la base de cas n'a pas une taille suffisante pour permettre de couvrir de manière satisfaisante l'ensemble des possibilités.

Ainsi, la solution réutilisée peut ne pas être optimale mais être seulement la meilleure solution à disposition du système. La robustesse d'un système RàPC est donc fortement liée à l'existence d'une adaptation de la solution pour pallier la différence entre les deux problèmes. Le fait de trouver comment une solution peut être adaptée de manière efficace représente à la fois un verrou technique important et un point critique de ce genre de système.

En reprenant le carré d'analogie (Figure 3.1), l'adaptation consiste à déterminer la solution cible s_q à partir d'un triplet constitué du problème cible q , du problème source p et de la solution source s_p . Il s'agit donc d'une transformation mathématique notée A telle que :

$$A(q, p, s_p) = s_q \quad (3)$$

La conception de la transformation A dépend de la stratégie adoptée et la littérature en comporte différentes. Par exemple, l'adaptation par règles consiste en l'application de règles remplissant le conteneur [melis1998reformulation]. Le système de RàPC EquiVox est un exemple d'utilisation de règles, déterminées empiriquement et par réseaux de neurones artificiels, pour conduire l'adaptation via l'usage d'un vecteur de combinaison [henriet2014equivox].

Il existe cependant d'autres stratégies :

- *L'adaptation différentielle* est basée sur l'évaluation de la différence α_{probleme} entre le problème source et le problème cible afin d'approximer la différence α_{solution} entre les deux solutions source et cible [fuchs2000adaptation]. Dans ce but, Cordier *et al.* utilisent des fonctions d'influence reliant les variations des descripteurs des problèmes à ceux des descripteurs des solutions [cordier2006adaptation]. Cette stratégie se rapproche de *l'adaptation par transformation* est décrite par J. Lieber [lieber2008hdr]. L'auteur y distingue également l'adaptation par dérivation, s'appuyant sur une adaptation par transformation, mais qui diffère de la précédente en s'intéressant à la relation de dépendance β liant un problème à sa solution.
- *L'adaptation par généralisation/spécialisation* consiste quant à elle à utiliser une solution $Sol(gen)$ d'un cas plus générale gen que le cas *cible* p à résoudre et à l'adapter en la spécialisant (par exemple une fonction mathématique $f(x)$ est une solution générale quand $f(2)$ est une solution spécialisée). Cette stratégie d'adaptation par spécialisation rejoint la notion de « *chemin de similarité* » décrite dans

[lieber2008hdr] et permettant d'aboutir à un « *chemin d'adaptation* ». Un chemin de similarité est une séquence de relations r_i reliant un problème *source* pb_0 à un problème *cible* pb_n telle que :

$$source = pb_0 r_1 pb_1 r_2 pb_2 \dots pb_{q-1} r_{q-1} pb_q = cible \quad (4)$$

La construction de l'adaptation se base sur une association de chaque relation r_i à une fonction d'adaptation Ar_i pour former un couple (r_i, Ar_i) appelé « *reformulation* ». Le chemin d'adaptation est alors la composition des différentes fonctions Ar_i de spécialisation de la solution s_0 à pb_0 telle que :

$$Ar_{r_1 \circ r_2 \circ \dots \circ r_{q-1} \circ r_q}(s_0) = (Ar_{r_q} \circ Ar_{r_{q-1}} \circ \dots \circ Ar_{r_2} \circ Ar_{r_1})(s_0) = Sol(cible) \quad (5)$$

Où $r_1 \circ r_2 \circ \dots \circ r_{q-1} \circ r_q$ est la composition des relations de similarités. Les concepts de « *généralisation* » et « *d'abstraction* » sont généralement utilisés de manière équivalente. Cependant, J. Lieber différencie dans [lieber2008hdr] une adaptation par *abstraction/raffinement* dans laquelle la solution $Sol(abst)$ d'un problème abstrait $abst$ ne représente qu'une solution partielle (et donc insuffisante) de p .

- Enfin, *l'adaptation conservative* [lieber2007conservativeAdapt] est fondée sur la théorie de la révision qui consiste à appliquer des changements minimums (aussi faibles que possible) à la solution source tout en s'assurant de sa cohérence avec la définition du problème cible [alchourron1985revisionTheory]. Des applications de cette stratégie peuvent être trouvées notamment pour la résolution de problèmes spatiaux et temporels [dufour2012adaptation] ou encore pour la détermination de thérapies en oncologie [daquin2006adaptation].

Il est intéressant de noter que l'utilité d'un cas peut ne pas dépendre directement de la mesure de similarité. L'objectif étant de maximiser la pertinence du résultat, la sélection d'un cas moins similaire mais dont la solution est plus facilement adaptable peut être en théorie préférable. Dans cet optique, Smyth *et al.* proposent par exemple d'inclure un critère d'adaptation pour que la capacité de la solution d'un cas à être adaptée participe à sa sélection [smyth1998adaptationCriterion].

3.2.2/ LE CYCLE DU RÀPC

En 1994, A. Aamodt et E. Plaza proposent un cycle composé de 4 étapes servant de base à la majeure partie des systèmes de RàPC qui ont suivi [aamodt1994cbr] :

1. **Retrouver/Remémorer** le cas stocké dans la base le plus similaire au problème cible,
2. **Réutiliser** les informations et la connaissance contenues dans le cas remémoré pour résoudre le problème,
3. **Réviser** la solution proposée, si nécessaire,
4. **Retenir** le résultat comme une nouvelle expérience qui sera probablement utile pour résoudre de futurs problèmes.

Un nouveau problème est résolu par la remémoration d'une ou plusieurs expériences passées sur lesquelles s'appuyera le système pour le résoudre. La sélection de ces expériences (appelées cas) se fait par une mesure de leur similarité avec le problème cible. Une fois le cas le plus proche récupéré durant la phase de remémoration, la solution qui lui est associée est réutilisée. Une fois le problème résolu, le résultat est vérifié et la solution est adaptée, si besoin. De la dernière étape découle la capacité d'apprentissage du système de RàPC. Le problème nouvellement résolu, associé à sa solution, est intégré à la base de connaissances pour l'enrichir. Ainsi, un tel système voit théoriquement ses performances évoluer à la hausse avec le temps. L'ensemble du cycle est illustré par la Figure 3.4.

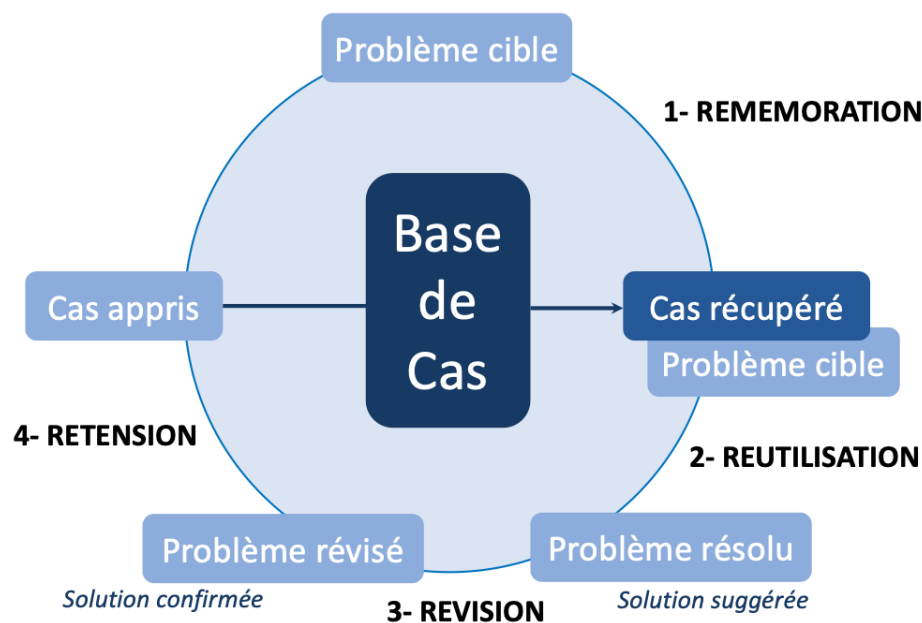


FIGURE 3.4 – Cycle RàPC issu de [aamodt1994cbr]

Bien que cette étape soit implicitement contenue dans l'étape plus générale de la réutilisation de la solution remémorée, un cycle exhaustif fait généralement distinctement apparaître l'adaptation de cette solution. De plus, ce cycle se termine généralement par une étape de maintenance. La Figure 3.5 donne une seconde version plus complète du cycle. L'adaptation formant l'un des 4 conteneurs majeurs d'un RàPC ayant été abordée dans la section 3.2.1.4, elle ne sera pas détaillée ici. Le rôle de l'étape de maintenance est de gérer la phase d'apprentissage du système et de modifier en conséquences le contenu de la base de cas. Ceci peut simplement consister à ajouter le nouveau cas dans la base dans le cadre d'une *flat base* mais peut également être une opération plus complexe si une organisation hiérarchique a été adoptée (Comment incorporer le nouveau cas ? Dans une catégorie existante ? Dans une nouvelle catégorie ?). Ainsi, le terme de « *capitalisation* » est plus couramment utilisé pour la dernière étape du cycle afin de refléter cette complexité. Ainsi, un autre point clé de cette maintenance est la pertinence d'une modification. Il n'est en effet pas toujours utile de faire grandir la base avec une nouvelle expérience si l'utilité de celle-ci pour le système est nulle (c'est à dire si son ajout ne permet pas d'améliorer les performances du RàPC). Par exemple, lorsque le système parvient à résoudre un problème sans intervention humaine, ceci signifie que sa base

de connaissance est suffisante pour y parvenir. Ainsi, l'ajout de ce problème en tant que nouvelle expérience ne permettra pas nécessairement d'augmenter sa compétence et ne contribuera qu'à encombrer inutilement la base de cas. À l'inverse, si une révision par un utilisateur est nécessaire, ceci met en évidence une lacune dans le domaine de connaissances couvert par cette base. Il devient donc tout à fait pertinent de la compléter. Il est également possible, dans l'intention de garder une base de cas optimale (voir section 3.2.1.3 pour la définition d'une base optimale), que l'ajout d'une expérience entraîne une certaine redondance avec une expérience plus ancienne, et donc une diminution de son utilité, rendant sa suppression pertinente. L'étape de maintenance ne consiste donc pas simplement à ajouter un nouveau cas ou non à la base, il s'agit avant tout de procéder à une mise à jour de la connaissance dans son ensemble (mise à jour des expériences, mise à jour des pondérations des attributs pour le calcul de similarité, mise à jour des règles d'adaptation, ...).

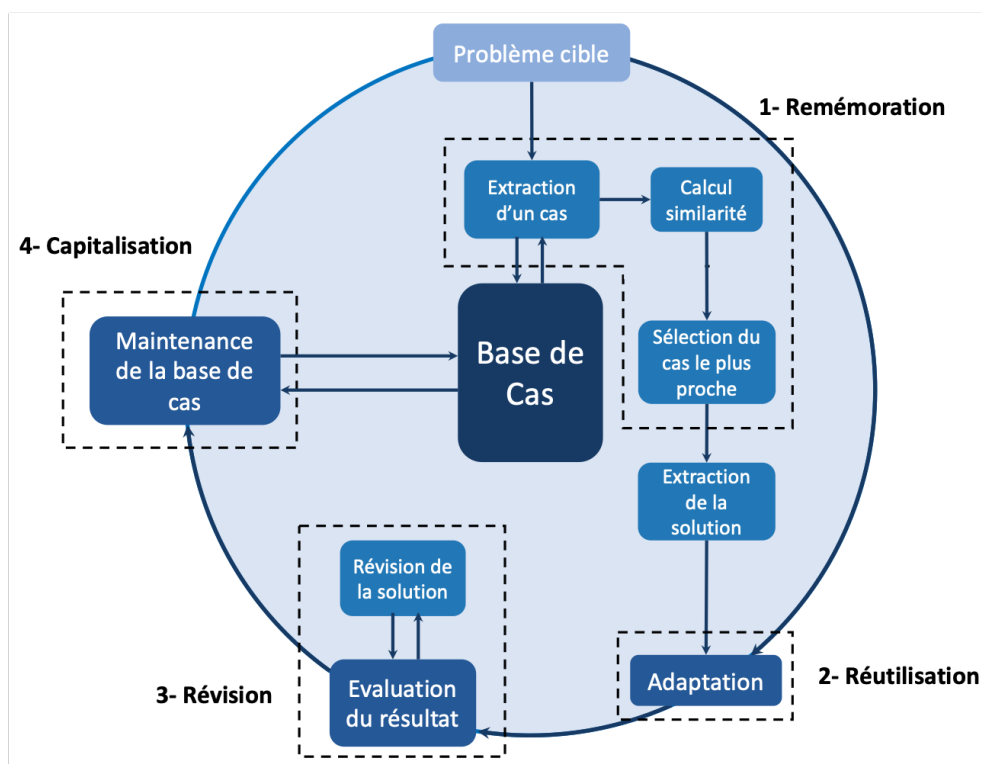


FIGURE 3.5 – Cycle d'un système de RàPC complet

3.3/ LES RÀPC POUR LA SEGMENTATION D'IMAGES

Un grand nombre de systèmes RàPC existent, notamment dans le domaine de la santé dans lequel cette approche reste populaire comme [daquin2006adaptation], [bichindaritz2010cbrHs], [henriet2014equivox], [nasiri2015medicalCbr] ou encore [gu2017cbrCancerDiagnosis], mais il existe également un certain nombre d'applications pour la vision par ordinateur. M. Grimnes et A. Aamodt décrivent en 1996 une des premières architectures pour la compréhension d'images scanner abdominales [grimnes1996imageUnderstanding], tandis que deux ans plus tard J. Jarmulak propose un système pour la classification d'images ultrasonores [jarmulak1998cbrClassif] et

qu'en 2000, Micarelli *et al.* s'intéressent à une application pour la reconnaissance d'images via l'utilisation d'une transformée en ondelettes [micarelli2000cbrRecognition]. Une autre application intéressante a été réalisée en 1999 par Ficet-Cauchard *et al.* [ficet1999cbr]. L'architecture d'un système interactif offre la possibilité à l'utilisateur de recourir à un ensemble de modules, librement sélectionnables et paramétrables, afin de réaliser une tâche particulière en traitement d'images (ex : segmentation d'une image). Ces modules représentent autant de sous-tâches à réaliser, séquentiellement ou non, pour parvenir au but recherché. Le RàPC est utilisé ici en tant qu'aide pour permettre à l'utilisateur de construire son application suivant cet objectif. La remémoration d'applications déjà construites, suffisamment similaires en terme de tâche à réaliser et d'images à traiter, permet de proposer à l'utilisateur un ensemble de plans (composés de modules organisées en un arbre selon un modèle *Task-Method-Tools* et de leurs paramètres respectifs). Il peut ensuite sélectionner le plan qui lui convient le mieux, selon le contexte et le résultat souhaité, et le modifier à loisir. L'adaptation de la solution proposée peut se faire à plusieurs niveaux, que ce soit au niveau des modules (substitution d'un ou plusieurs modules et/ou modification de leurs paramètres) ou des sous-tâches (substitution d'une sous-tâche par une autre). Cette adaptation a pour particularité de pouvoir recourir à la solution de plusieurs cas pour construire une solution originale mieux adaptée au problème à traiter mais n'est pas automatique. L'étape de mémorisation d'un nouveau plan en tant que nouveau cas de la base est aussi originale dans le sens où elle permet de construire et mémoriser plusieurs cas à partir d'un unique plan complet. Les sous-tâches de ce plan peuvent en effet être eux mêmes mémorisés suivant différents critères représentant la pertinence de cette incorporation. Plus spécifiquement dans le domaine médical, l'état de l'art de D. Wilson et D. O'Sullivan sur l'utilisation des systèmes de RàPC pour l'imagerie donne un bon aperçu de la richesse de la littérature sur le sujet [wilson2008medicalImageCbr].

P. Perner décrit les problématiques liées au traitement d'images ainsi que les réponses qui peuvent y être apportées grâce aux stratégies du RàPC [perner2001cbrForImages] [perner2016cbrForImages]. Notamment, les systèmes d'imagerie font souvent appel à des algorithmes fortement dépendant des données utilisées et il est courant que leur réutilisation sur une entrée différente mène à un effondrement des performances, empêchant leur généralisation. Ce problème peut venir des algorithmes en eux-mêmes (inadaptés pour certaines images) ou de leur paramétrage (mauvaises valeurs numériques utilisées en tant que paramètres). De plus, il est courant qu'un système doive faire face à un cas inédit non prévu par le programmeur. D'une part, le RàPC offre la possibilité de stocker la connaissance et de réutiliser la plus appropriée durant la phase de remémoration (type d'algorithmes à utiliser suivant l'image, valeurs numériques des paramètres à adopter). Et d'autre part, il permet une mise à jour constante de cette connaissance, augmentant la robustesse du système durant la phase de *maintenance/capitalisation* (ajout d'expériences initialement non couvertes par la base de connaissances).

Si l'analyse d'images a largement profité de l'application du RàPC, ceci est beaucoup moins vrai dans le domaine plus spécifique de la segmentation. À notre connaissance, deux travaux principaux, sur lesquels nous nous sommes appuyés, font références. Ils seront présentés en détails dans la suite du manuscrit : [perner1999segCBR] et [frucci2008watershedCBR]. Ces travaux ont d'ailleurs été repris et améliorés par P. Perner et A. Attig [attig2012watershedSeg].

Nous pouvons citer d'autres travaux dans ce domaine. Hudelot *et al.* utilisent un système, sans que ce soit réellement un RàPC, dans lequel la connaissance est stockée sous la

forme d'une ontologie associée à des relations floues pour segmenter des structures cérébrales de la matière grise dans des images IRM [hudelot2008fuzzySeg]. Une structure de référence est préalablement segmentée et les relations floues (relations spatiales entre cette structure de référence et la/les structure(s) à segmenter : proche de, à droite de, au-dessus de, ...) permettent de contraindre l'espace de recherche d'un modèle déformable.

Burgos-Artizzu *et al.* reprennent l'architecture RàPC de P. Perner [perner1999segCBR] pour une application agricole [burgos2009cbrSeg]. L'objectif est de segmenter automatiquement des épines malades tombées au sol sur des photographies couleurs afin d'estimer la quantité et de déterminer le meilleur dosage du traitement. Le système de RàPC est utilisé pour se remémorer un problème de segmentation proche et réutiliser sa solution (les méthodes et les valeurs de paramètres de ces méthodes). Il n'y a ici aucune phase d'adaptation et les paramètres sont ré-appliqués sans modification.

3.3.1/ UNE PREMIÈRE ARCHITECTURE DE P. PERNER POUR LA SEGMENTATION PAR SEUILLAGE

En 1999, P. Perner propose une architecture pour un système dédié à la segmentation d'images médicales et basée sur un RàPC [perner1999segCBR]. La plupart des algorithmes de segmentation d'images nécessitant l'utilisation de paramètres numériques doivent être modifiés suivant les données [perner2001cbrForImages]. Ainsi, dans l'approche de P. Perner, le RàPC est utilisé pour sélectionner les paramètres de segmentation en accord avec les caractéristiques de l'image à segmenter. La méthode a été évaluée sur des images cérébrales scanner (CT-scans) en format DICOM. L'objectif du système est de permettre une segmentation d'une part du cerveau et d'autre part du liquide cérébro-spinal. Le calcul du ratio entre ces deux structures est en effet un bon paramètre pour caractériser la gravité de certaines maladies neuro-dégénératives.

3.3.1.1/ L'ARCHITECTURE DU SYSTÈME DE RÀPC DE P. PERNER

P. Perner décrit son architecture en deux parties : une unité de segmentation et une unité de gestion de la base.

L'unité de segmentation

Cette partie du système est dédiée, comme son nom l'indique, à la réalisation de la segmentation. Elle inclut la phase de remémoration (ainsi que ses sous-étapes : calcul de similarité, sélection du cas le plus proche et extraction de la solution) et la procédure de segmentation. La Figure 5.1.1 résume le fonctionnement de l'unité de segmentation. Le volume de données correspondant à un patient (30 coupes) est fourni en une seule fois au système mais le traitement se fait de manière séquentielle, coupe par coupe, et non pas en 3D. Avant de lancer la procédure de segmentation, les images sont pré-traitées afin d'éliminer les structures peu pertinentes comme le crâne.

Deux types d'informations sont extraites :

- Les informations non images (âge du patient, sexe, ...) sont extraites directement du *header* du fichier DICOM.
- Les informations statistiques de l'images (moyenne, variance, kurtosis, ...) sont calculées directement à partir de la matrice de pixels.

Ces 2 types d'informations permettent le calcul de la similarité. Les détails du calcul sont abordés dans la section 3.3.1.4. Suivant la similarité la plus haute, le cas le plus proche peut ensuite être sélectionné. À partir de celui-ci, les paramètres de la segmentation peuvent être déterminés et utilisés pour guider la segmentation d'une part du cerveau et d'autre part du liquide cérébro-spinal. P. Perner semble indiquer que l'algorithme est ajusté mais sans faire explicitement référence à une phase d'adaptation. Une fois la totalité des coupes segmentées, le volume cerveau/liquide peut être calculé puis affiché à l'utilisateur.

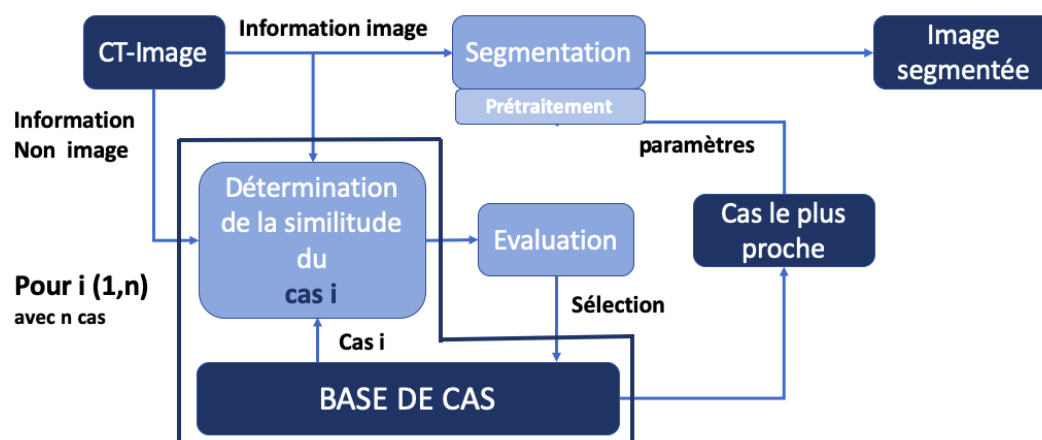


FIGURE 3.6 – Fonctionnement de l'unité de segmentation du système de RàPC issu de [perner1999segCBR]

L'unité de maintenance de la base de cas

Le rôle de cette unité est de vérifier la pertinence du résultat et de gérer la phase d'apprentissage du système en mettant à jour la connaissance contenue dans la base. Une fois la segmentation calculée, une étape d'évaluation est assurée par l'utilisateur. S'il détecte des erreurs dans la segmentation des zones d'intérêt, il a la possibilité d'indiquer au système que le résultat est incorrect, ce qui lance le processus de gestion de la base. La procédure débute de la même façon si aucun cas suffisamment similaire n'est trouvé. Une seconde étape de segmentation permet à l'utilisateur de déterminer les meilleurs paramètres. Ces derniers sont ensuite sauvegardés, avec les informations image (statistiques calculées sur le CT-Scan) et les informations non-image (extraites depuis le *header* du DICOM). Le tout forme alors un nouveau cas qui est stocké dans la base de connaissances. La gestion de cette base passe néanmoins par une étape que P. Perner nomme la « *généralisation du cas* ». Son objectif est de compléter la connaissance sans construire une base qui deviendrait à terme trop importante. La généralisation consiste à grouper les paramètres de segmentation en différents *clusters* : chaque combinaison

formant un cluster. Le nom du cluster est ensuite associé aux autres informations du cas. Cependant, cette partie du système est décrite comme étant manuelle.

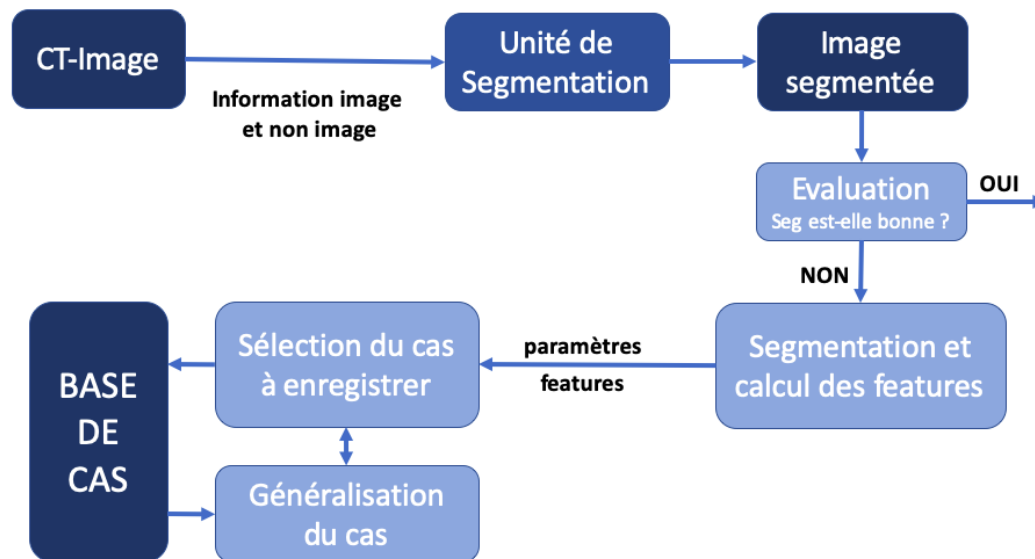


FIGURE 3.7 – Fonctionnement de l'unité de maintenance du système de RàPC issue de [perner1999segCBR]

3.3.1.2/ L'ALGORITHME DE SEGMENTATION D'IMAGES DANS LE RÀPC DE P. PERNER

L'algorithme de segmentation est basé sur le calcul de l'histogramme de l'image (en niveaux de gris). Cet histogramme est lissé par l'utilisation de fonctions numériques et de règles heuristiques afin de trouver les valeurs de seuils à appliquer pour segmenter le cerveau et le liquide. Ces paramètres et ces règles sont à adapter suivant l'image pour permettre un résultat pertinent (et forment donc la solution).

La première étape, le lissage de l'histogramme, se fait par l'application d'une fonction numérique. Cette fonction fait appel à deux paramètres : la complexité de l'interpolation et sa largeur (espace entre deux points servant au calcul de l'interpolation). Puis, chaque pic dans l'histogramme permet de déterminer un intervalle. Un intervalle débute par une vallée, contient un pic et finit par une seconde vallée. Chaque intervalle correspond à une ou plusieurs structures dans l'image.

La seconde étape est de fusionner les intervalles déterminés précédemment afin d'en obtenir un nombre correspondant au nombre de labels recherchés dans l'image (structures à segmenter). Cette fusion se base sur le calcul d'un ratio entre la hauteur d'un pic et la hauteur (valeur) la plus élevée parmi ses deux bornes. Un intervalle est fusionné avec un second partageant sa borne qui a la valeur la plus élevée si son ratio (exprimé en pourcentage) est inférieur à un certain seuil (ceci aboutit donc à l'élimination de l'intervalle en question). Ceci revient à tester la hauteur relative du pic et à vérifier qu'il soit suffisamment significatif pour constituer un label dans l'image. Si le nombre final d'intervalles est supérieur au nombre de labels voulu, alors les deux pics ayant la valeur la plus élevée sont sélectionnés pour calculer les seuils de segmentation.

La solution de l'algorithme de segmentation est donc constituée par les paramètres de la

fonction numérique (complexité et largeur) pour le lissage et par le seuil pour la sélection des intervalles pertinents.

3.3.1.3/ LA MODÉLISATION D'UN CAS DANS LE RÀPC DE P. PERNER

La modélisation d'un cas dans un système de RàPC est souvent divisée en deux parties. La partie problème contient tous les descripteurs nécessaires au calcul de similarité afin de trouver le cas source le plus proche du problème cible. La partie solution contient tous les paramètres nécessaires à l'algorithme de segmentation pour le calcul des résultats. Dans le cas présent, la modélisation a été réalisée de sorte qu'un cas stocké corresponde à une image (coupe) préalablement segmentée. Cette modélisation peut se résumer comme suit :

- **La partie problème :**

Elle est constituée de deux sous-parties : la partie image et la partie non image. Les informations non images sont extraites de l'entête (*header*) du DICOM et sont constituées du **sexe du patient**, de **son âge**, de **l'épaisseur des coupes** et du **nombre de coupes**. Le numéro de la coupe est transformé en valeur qualitative, suivant la position de celle-ci dans l'examen du patient, selon 3 qualificatifs : Haut, Milieu et Bas. La partie image est constituée de différentes mesures statistiques comme **la moyenne**, **la variance**, **le curtosis**, **l'entropie**, **le centroïde** ou encore **l'asymétrie**.

- **La partie solution :**

Ce qui constitue la solution a été abordé et décrit dans la section 3.3.1.2. Il s'agit de tous les paramètres requis par l'algorithme de segmentation : **la complexité et la largeur d'interpolation** pour la fonction de lissage et **le seuil pour la sélection des pics** dans l'histogramme.

3.3.1.4/ LE CALCUL DE LA SIMILARITÉ ENTRE LES DEUX IMAGES DANS LE RÀPC DE P. PERNER

P. Perner a effectué en amont un travail d'évaluation entre deux approches pour le calcul de la similarité : celle de P. Zamperoni et V. Starovoitov, décrivant une mesure quantitative de la diversité entre deux images en niveaux de gris [zamperoni1995similarity] et celle, plus simple, de H. Dreyer et W. Sauer [dreyer1982prozessanalyse]. Malgré certains avantages de la méthode de P. Zamperoni (meilleure discrimination), P. Perner met en évidence certaines de ses limites : sensibilité à la rotation et à la translation due à la méthode de calcul (fenêtre glissante) et alourdissement de la base de cas à cause de la nécessité de stocker la matrice image. Pour ces raisons, le calcul de H. Dreyer se basant exclusivement sur les statistiques globales de l'image a été préféré par P. Perner.

Une première similarité est déterminée pour la partie non image des descripteurs et une seconde pour la partie Image. La similitude globale est calculée par (6) :

$$Sim = \frac{1}{2}(Sim_N + Sim_I) = \frac{1}{2}(S(C_i, b) + 1 - dist_{AB}) \quad (6)$$

Où :

- Sim_N la similarité sur les informations non Image,
- Sim_I la similarité sur les informations Image,
- C_i correspond au cas numéro i dans la base
- et b correspond à un nouveau problème cible.

Dans cette formule, P. Perner a donc choisi d'attribuer une pondération égale aux deux similarités de manière à ce qu'une haute similarité globale corresponde à ce que ces deux similarités le soit également.

Le calcul de la similarité Sim_N est basé sur une mesure de Tversky. La similarité entre un cas C_i de la base et un nouveau cas problème b est donné par (7) :

$$S(C_i, b) = \frac{|A_i|}{\alpha|A_i| + \beta|D_i| + \gamma|E_i|} \quad (7)$$

Où :

- $|A_i|$ est le nombre de caractéristiques (*features*) en commun entre C_i et b ,
- $|D_i|$ est le nombre de *features* appartenant uniquement à C_i ,
- $|E_i|$ est le nombre de *features* appartenant uniquement à b ,
- Les valeurs α , β et γ sont des valeurs de pondérations, choisies empiriquement, telles que $\alpha = 1$ et $\beta = \gamma = 0.5$.

Nous pouvons noter que cette formule permet de calculer une similarité à partir d'attributs qualitatifs. La comparaison est absolue (identique ou différent) et il n'est pas possible avec une mesure de Tversky d'évaluer la proximité entre les deux cas vis à vis d'un attribut donné qui serait quantitatif (par exemple le nombre de coupes et l'épaisseur des coupes).

Le calcul de la similarité Sim_I sur les informations Image entre une image A et une image B est quant à lui la distance définie par (8) :

$$dist_{AB} = \frac{1}{k} \sum_{j=1}^k w_j \left| \frac{C_{jA} - C_{jmin}}{C_{jmax} - C_{jmin}} - \frac{C_{jB} - C_{jmin}}{C_{jmax} - C_{jmin}} \right| \quad (8)$$

Où :

- C_{jA} est le j^{ieme} attribut de l'image A,
- C_{jB} est le j^{ieme} attribut de l'image B,
- C_{jmax} est la valeur maximale dans la base de le j^{ieme} attribut,
- C_{jmin} est la valeur minimale pour cet attribut,
- w_j est la valeur de pondération attribuée au j^{ieme} attribut telle que $\sum_{j=1}^k w_j = 1$.

3.3.1.5/ L'ÉVALUATION DES PERFORMANCES DU RÀPC DE P. PERNER

L'évaluation a été faite sur près de 600 images provenant de 30 patients différents. La base de cas constituée pour les tests contenait 130 cas (130 coupes). Il n'y a pas eu de score calculé comme les indices DICE ou IoU mais une comparaison a été faite entre le ratio cerveau/liquide calculé par le système et celui calculé par un expert. Les résultats montrent une sous-estimation systématique du ratio par le système par rapport à la vérité terrain. L'auteur analyse cette sous-estimation comme une sur-segmentation fréquente du liquide. Toutefois, les résultats montrent une forte corrélation linéaire ($r = 0.85$) entre les valeurs du système et les valeurs de l'expert. Dit autrement, les rapports de surface cerveau/liquide les plus hauts obtenus par le système correspondent aux rapports les plus hauts obtenus par l'expert, ce qui indique une forte cohérence des résultats.

La base de cas a été construite en s'appuyant sur la connaissance de l'expert qui a déterminé manuellement les meilleurs paramètres de segmentations pour chacun des cas (en s'appuyant certainement sur les vérités terrains des chirurgiens).

3.3.2/ UNE NOUVELLE APPLICATION DU SYSTÈME DE RÀPC DE P. PERNER : SEGMENTATION PAR LIGNE DE PARTAGE DES EAUX (*watershed*) DE FRUCCI *et al.*

En 2008, Frucci *et al.* reprennent le système de 1999 pour l'appliquer à la segmentation d'images par ligne de partage des eaux (connue en anglais sous le terme de *watershed*), un classique des algorithmes de segmentation d'images [frucci2008watershedCBR]. Le système de RàPC, en lui-même, n'a subi que peu d'évolutions. Ainsi, son architecture, le calcul de la similarité et la modélisation d'un cas sont identiques (voir section 3.3.1).

Cette section se concentrera sur la présentation de l'algorithme de partage des eaux, sur la manière de le coupler à un système de RàPC et sur l'évaluation des performances.

3.3.2.1/ LA SEGMENTATION PAR LIGNE DE PARTAGE DES EAUX DE FRUCCI *et al.*

La ligne de partage des eaux est initialement un concept géographique qui divise un territoire en un ou plusieurs bassins versants où l'eau se déverse dans des directions différentes de part et d'autre de cette ligne. L'algorithme de partage des eaux (ou plutôt la famille d'algorithmes associés à cette méthode) consiste à considérer une image comme un relief topographique où les intensités de pixels correspondraient à une altitude (voir Figure 3.8). Son objectif est de déterminer les différentes lignes de partage des eaux délimitant des vallées locales dans l'image.

Il s'agit d'un algorithme itératif, semblable à la croissance de régions, qui suit un processus d'inondation et son déroulement est le suivant :

1. Chaque vallée est percée en son minimum (minimum local),
2. L'eau monte et remplit les vallées depuis leur minimum,
3. Lorsque deux bassins se rejoignent, une ligne de partage des eaux est créée (un « barrage »),
4. Les différentes lignes forment la segmentation (le « contours ») de l'image.

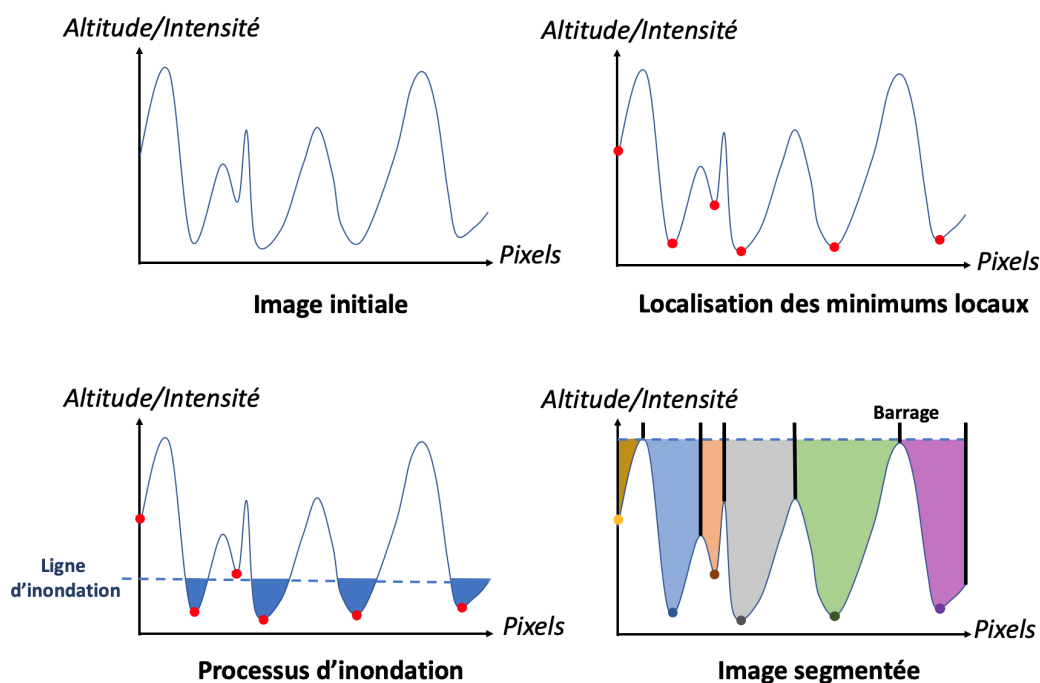


FIGURE 3.8 – Fonctionnement d'un algorithme de ligne de partage des eaux (LPE)

La ligne de partage des eaux (LPE) est souvent réalisée sur l'image gradient et non pas directement sur l'image originale. Le parallèle avec la croissance de région vient du fait que le processus d'inondation revient à placer des germes dans l'image et à faire croître une région à partir de chaque germe. Ainsi, l'initialisation de l'algorithme est également le point le plus critique. Dans une ligne de partage des eaux classique, chaque minimum local dans l'image devient un point d'inondation. La conséquence est que cet algorithme tend à produire une importante sur-segmentation. Ce défaut peut être amélioré en pré-traitant l'image, notamment avec un filtrage moyen, afin de diminuer le nombre de minima locaux et donc la sur-segmentation qui en résulte. Une solution généralement adoptée est de réaliser une segmentation par LPE sous contraintes. Ces contraintes peuvent être des marqueurs spatiaux (pour initialiser l'inondation avec un nombre défini de minima régionaux, l'analogie pouvant être faite avec le placement manuel d'un germe pour la croissance de régions) ou un critère de contraste, mais posent des problèmes en termes d'automatisation de l'algorithme.

En 2006, M. Frucci propose d'introduire un critère dans l'algorithme de ligne de partage des eaux pour réduire significativement la sur-segmentation de l'image [frucci2006watershed] et reprend ce critère pour le coupler au RàPC de P. Perner. Ce critère permet d'évaluer la pertinence de chaque région et de supprimer les germes jugés inutiles. Il se compose de deux mesures correspondant respectivement à la profondeur relative D_{xy} , d'un bassin x suivant un bassin y , et de Δ_{xy} , la différence entre les minimums de x et y . L'algorithme de segmentation est contrôlé par deux paramètres de seuils, d_t et δ_t , associés respectivement à D_{xy} et Δ_{xy} . Ces deux seuils sont automatiquement calculés suivant les statistiques de la partition en ligne de partage des eaux initiale de l'image en niveau de gris. Si $D_{xy} < d_t$ et $\Delta_{xy} < \delta_t$, alors le germe correspondant à cette région est supprimé. La segmentation est réalisée et les régions sont testées jusqu'à convergence de l'algorithme vers un résultat optimal. Afin de pouvoir pondérer l'importance des deux mesures en fonction de l'image à segmenter, Frucci *et al.* les combinent au sein d'un

calcul global :

$$\frac{1}{2} \left(\frac{a\Delta_{xy}}{\delta_t} + \frac{bD_{xy}}{d_t} \right) > T \quad (9)$$

Où :

- a et b sont les pondérations associées respectivement à l'importance de Δ_{xy} et D_{xy} dans l'évaluation de la pertinence d'une région, telles que $a + b = 2$,
- T est une valeur de seuil globale.

Pour chaque image utilisée pour construire la base de cas, la segmentation est lancée avec différentes combinaisons de a , b et T et la meilleure image est retenue par comparaison avec la vérité terrain. La meilleure combinaison forme ainsi la partie solution du cas.

3.3.2.2/ L'ORGANISATION DE LA BASE DE CAS DE FRUCCI *et al.*

La base de cas a été construite à partir d'images de différents domaines (cellules nerveuses, bâtiments, animaux, ...) divisées en groupes d'images similaires (voir Figure 3.9). Chaque groupe forme une classe décrite par un prototype. Ce prototype contient les statistiques moyennes de la classe ainsi que les paramètres de segmentation associés. Ceci se base sur l'hypothèse que ces paramètres fournissent les meilleures segmentations pour toutes les images similaires d'un même groupe. Si ces paramètres sont identiques pour toutes les images d'une classe, ils sont stockés dans le prototype. S'ils diffèrent, des paramètres moyens sont calculés pour la classe. Les images d'une même classe sont décrites par un seul et même prototype. Cette organisation permet de limiter la taille de la base tout en maximisant le domaine de connaissance qu'elle couvre.

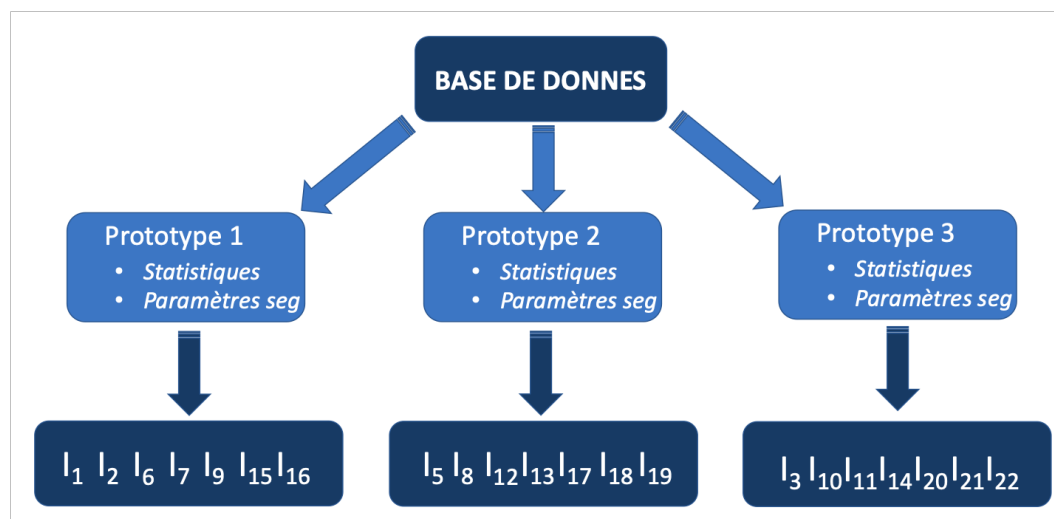


FIGURE 3.9 – Organisation de la base de cas en prototypes

3.3.2.3/ L'ÉVALUATION DU SYSTÈME DE FRUCCI *et al.*

Les auteurs ne donnent pas dans ce travail une évaluation complète de la performance de leur système. Les quelques exemples utilisés tendent à montrer que le RàPC, couplé à cette méthode de segmentation, permet de limiter drastiquement le problème de sur-segmentation, sans parvenir à l'éliminer complètement. Dans l'un de ces exemples, les auteurs sont parvenus à obtenir une segmentation constituée de seulement 382 régions par une recherche expérimentale des meilleurs paramètres, comparée à une ligne de partage des eaux classique initialisée automatiquement qui fournit 6370 régions. Le système de RàPC, après remémoration et réutilisation de la solution source, est parvenu sur cette même image à produire une segmentation en 454 régions. Ceci démontre qu'un RàPC est tout à fait apte à fournir une solution proche de l'optimal. En terme d'évaluation absolue des résultats, ceux-ci se heurtent aux limites d'un algorithme de ligne de partage des eaux, produisant naturellement une situation de sur-segmentation.

Ceci renforce l'idée que coupler un RàPC avec un algorithme de segmentation différent, n'ayant pas ces mêmes limites, pourrait être intéressant.

Synthèse du chapitre

Le Raisonnement à Partir de Cas (RàPC) est une approche incluant plusieurs types de connaissances qui diffère d'autres méthodes d'Intelligence Artificielle comme les Réseaux de Neurones Artificiels (RNA) basés uniquement sur l'expérience.

Ceci permet d'envisager de concevoir un système efficace même en l'absence d'une large base de données, ce manque étant compensé par l'injection de connaissances métier. Il y a dans la littérature un grand nombre de travaux traitant de l'application du RàPC au domaine du traitement d'images. Cependant, peu d'auteurs se sont intéressés au domaine plus spécifique de la segmentation. P. Perner propose, en 1999, un système de RàPC pour la segmentation d'images médicales, mettant au point un calcul de similarité efficace pour retrouver le cas le plus pertinent dans la base et le couplant à un algorithme de segmentation par seuillage d'histogramme. L'auteur imagine pour cela une architecture qui sera largement reprise dans le travail de Frucci *et al.* en 2008. Cette fois-ci, le système de RàPC est couplé à un algorithme de ligne de partage des eaux et les auteurs tendent à montrer que ce système est capable de trouver pour de nouveaux problèmes une solution proche de l'optimale (sans en démontrer formellement l'efficacité sur tout un ensemble de tests). La ligne de partage des eaux possède comme défaut majeur lorsqu'elle est initialisée automatiquement, de mener à une sur-segmentation. La variante proposée par M. Frucci permet de drastiquement diminuer le nombre de régions, sans toutefois parvenir à éliminer complètement le problème de sur-segmentation.

Ceci renforce l'idée que coupler un RàPC à un algorithme de segmentation différent, n'ayant pas ces mêmes limites, est une perspective intéressante. Un système de RàPC serait donc potentiellement capable de guider un processus de segmentation comme le ferait un expert du domaine (chirurgien, radiologue, . . .).

Cette première partie était consacrée à l'explication du contexte propre à ce travail de thèse et à un état de l'art approfondi sur l'usage des approches d'Intelligence Artificielle pour la segmentation d'images. Nous nous sommes en particulier intéressés au Raisonnement à Partir de Cas (RàPC) et aux Réseaux de Neurones Artificiels (RNA). La seconde partie de ce document abordera nos contributions au domaine. Et plus particulièrement, le chapitre suivant présente la plate-forme COLISEUM-3D dans son ensemble avant de détailler plus spécifiquement chacun de ses différents composants.



CONTRIBUTION

Cette partie détaille notre contribution qui s'articule principalement autour de la conception d'une nouvelle plate-forme pour la segmentation d'images médicales. Au sein de cette plate-forme, nous avons, en particulier, conçu un système de Raisonnement à Partir de Cas (RàPC) pour la segmentation du parenchyme rénal ainsi qu'une méthode d'entraînement originale pour un réseau de neurones artificiels profond afin de segmenter le néphroblastome. La plate-forme est donc appliquée au cas des néphroblastomes, et elle est capable de segmenter de manière indépendante les structures abdominales d'intérêt et de fusionner ces informations au sein d'une seule et même image.

Le quatrième chapitre se consacre à une présentation générale de la plate-forme COLISEUM-3D (**COL**laborative platform with artificial **I**ntelligence for **SE**gmentation of **tU**moral kidney in **M**edical images in **3D**), de son organisation et de son fonctionnement depuis l'injection des images du patient à segmenter jusqu'à la production de l'image finale regroupant les segmentations de l'ensemble des structures d'intérêt. Une fois que nous avons défini l'architecture de cette nouvelle plate-forme, notre travail principal dans cette thèse a consisté à se concentrer sur les segmentations de la tumeur et du rein pathologique.

Le cinquième chapitre expose notre contribution dans l'usage du Raisonnement à Partir de Cas (RàPC) pour réaliser la segmentation du rein tumoral. Celle-ci s'articule autour du couplage du RàPC avec un algorithme de croissance de régions et de la conception d'une phase d'adaptation efficace de la solution remémorée pour optimiser le résultat. Il aborde également la conception d'une nouvelle méthode d'entraînement, pour des réseaux convolutifs profonds, permettant d'exploiter un phénomène de sur-apprentissage pour limiter la quantité de données requises pour l'apprentissage du réseau.

Enfin, le sixième et dernier chapitre de cette partie présente l'ensemble des résultats pour les deux approches précédentes, avec une évaluation de leur performance.

COLISEUM-3D : UNE PLATE-FORME INNOVANTE POUR LA SEGMENTATION D'IMAGES MÉDICALES

La segmentation est l'une des étapes clés dans la construction d'une représentation 3D. Le problème réside en la nature incertaine des structures d'intérêt, autant sur leur forme que leur superficie, et ceci est dû à la présence d'une tumeur aux caractéristiques imprévisibles qui peuvent fortement varier d'un patient à l'autre. En conséquence, les radiologues et les chirurgiens doivent réaliser et vérifier les segmentations de plus de 200 images scanner manuellement pour chacun des patients afin d'optimiser au mieux la thérapie : ce qui nécessite entre 7 et 9 heures et est donc difficilement réalisable en pratique. L'Intelligence Artificielle (IA) est un puissant outil représentant une solution viable pour la conception d'un traitement totalement automatique des clichés.

L'intelligence artificielle est un procédé de plus en plus utilisé en vision par ordinateur afin de permettre le traitement automatique d'images par un programme. Il existe différentes approches, dont certaines ont pu être détaillées dans la partie I, ayant chacune leurs performances et leurs contraintes. Les réseaux de neurones convolutifs exigent par exemple un grand nombre de données labellisées pour s'entraîner dans de bonnes conditions alors que le Raisonnement à Partir de Cas (RàPC) est fortement dépendant de son domaine de connaissance. Afin de maximiser leur efficacité respective, il est pertinent de chercher à combiner ces approches d'IA pour des applications différentes dans lesquelles chacune d'elle est la plus performante et la plus intéressante. COLISEUM-3D est une concrétisation de cet objectif, alliant de la segmentation par RàPC, couplée à un algorithme de croissance de régions, suivie d'une segmentation par *Deep Learning* pour parvenir à produire le résultat souhaité tout en minimisant l'intervention humaine.

4.1/ PRÉSENTATION GÉNÉRALE DE LA PLATE-FORME

COLISEUM-3D (**COL**laborative platform with artificial Intelligence for **SE**gmentation of **tU**moral kidney in **M**edical images in **3D**) est une plate-forme dédiée à la segmentation de clichés scanner pour la mise en évidence de différentes structures abdominales chez l'enfant. Les structures d'intérêts sont le parenchyme (somme des tissus constituant les parties fonctionnelles) du rein pathologique, les cavités rénales correspondantes, le néphroblastome (tumeur) et les vaisseaux sanguins (artères et veines). L'entrée de la plate-

forme est constituée des différentes images du patient à segmenter. Ces images peuvent être les clichés en temps vasculaire ou en temps tardif suivant les structures d'intérêt (voir la section 1.3.2 pour plus de détails). Sa sortie est la segmentation de ces structures au sein d'une unique représentation numérique fusionnée.

COLISEUM-3D est organisée en couches, elles-mêmes constituées de modules, comme l'illustre la Figure 4.1. La couche données regroupe l'ensemble des données à disposition du système sur lesquelles s'appuyer pour résoudre les problèmes de segmentation. La couche de segmentation, comme son nom l'indique, produit les différentes segmentations à partir des différentes entrées (images scanner en temps vasculaire et en temps tardif). Enfin, la couche de fusion permet de regrouper les segmentations dans un même espace tout en résolvant les conflits qui peuvent survenir. Les modules implantés dans cette thèse sont présentés en couleur bleu avec bordure et les modules présentés sur un fond blanc feront l'objet de thèses futures au sein de l'équipe de recherche. Les entrées et les sorties du système sont mises en évidence par une coloration grise.

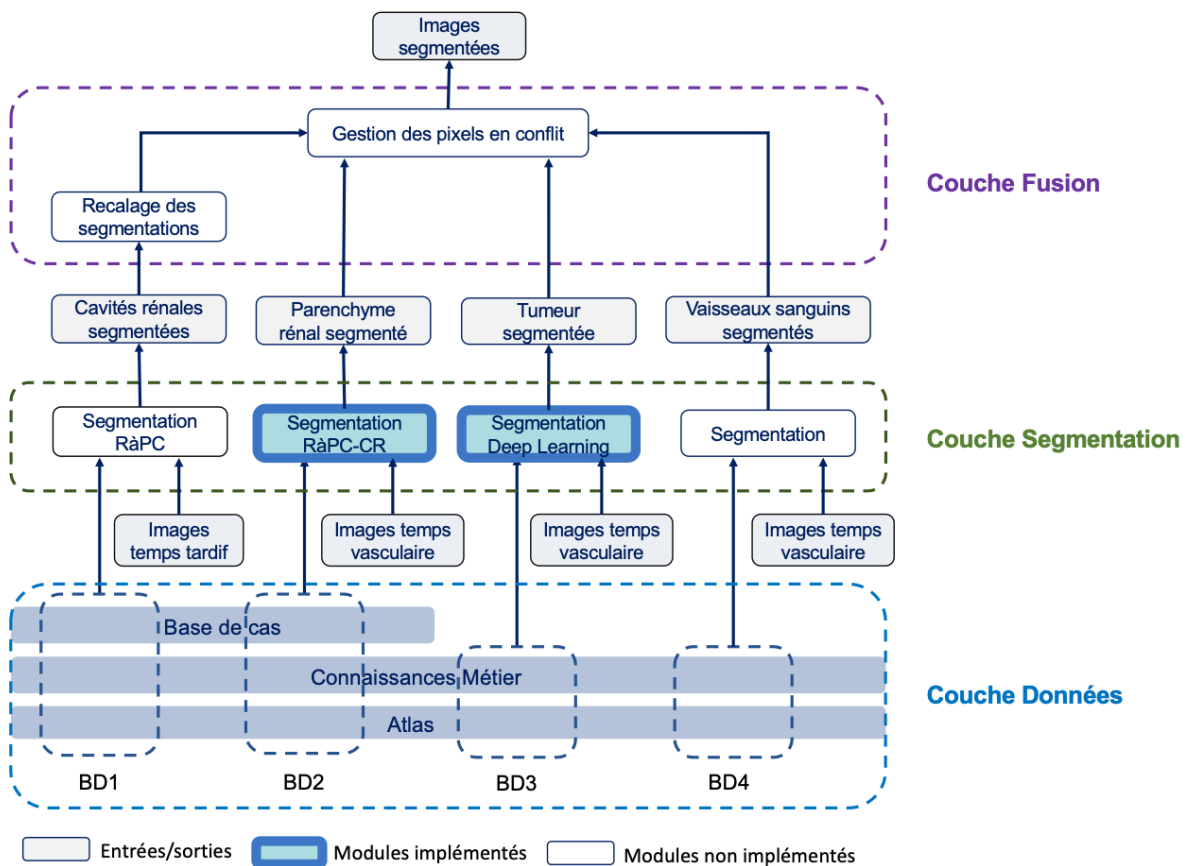


FIGURE 4.1 – Organisation de la plate-forme COLISEUM-3D

4.2/ LES RÔLES ET LE FONCTIONNEMENT DES DIFFÉRENTES COUCHES

COLISEUM-3D est définie en 3 couches : la couche de données, la couche de segmentation et enfin la couche de fusion.

Ces couches sont liées les unes aux autres et ont chacune un rôle précis et essentiel.

4.2.1/ LA COUCHES DE DONNÉES

La couche de données est fortement liée à la couche de segmentation dont les modules dépendent pour fonctionner correctement. Son organisation peut être plus facilement visualisée dans la Figure 4.2. Cette couche est décomposée en 3 sous-couches :

- La première est l'atlas qui correspond à une banque d'images exploitables (clichés scanner). Les images sont initialement en format DICOM mais ont été converties en format JPEG pour faciliter leur traitement.

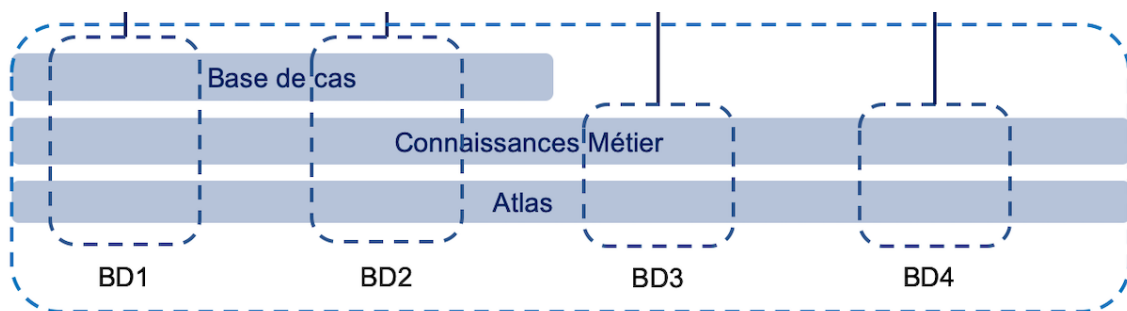


FIGURE 4.2 – Organisation de la couche données de COLISEUM

- La seconde représente les connaissances métier. Elle regroupe toute l'expertise médicale et technique essentielle au bon fonctionnement des modules. Ces connaissances correspondent tout d'abord à l'expertise médicale des chirurgiens au travers des vérités terrain réalisées par leurs soins (segmentations manuelles des images de l'atlas servant de référence et de résultat idéal). La seconde partie de ces connaissances est le paramétrage optimal des algorithmes de croissance de régions par un expert afin de produire les meilleures segmentations possibles à partir de l'atlas. Nous verrons par la suite qu'un algorithme de croissance de régions nécessite l'entrée de trois paramètres pour fonctionner : deux valeurs de seuils (local et global) et la localisation des germes (dont le nombre peut varier suivant l'image) pour l'initialisation.
- Enfin, la base de cas s'appuie sur les deux premières sous-couches pour rendre les connaissances et les données exploitables par les processus constituant le RàPC. La construction et le contenu de cette base de cas seront abordés dans la suite du manuscrit.

L'ensemble de ces 3 sous-couches permet ainsi de produire différentes bases de données suivant les besoins des modules de segmentation.

4.2.2/ LA COUCHE DE SEGMENTATION

La conception de cette couche de segmentation concentre la majorité des contributions présentées dans ce manuscrit et le fonctionnement des modules qui la compose sera abordé de manière détaillée dans les chapitres suivants. Elle permet de calculer les segmentations des différentes structures à partir des images en entrée, comme indiqué dans la Figure 4.3. La méthode de segmentation utilisée, représentée par un module, varie suivant le contexte c'est à dire suivant la structure qui doit être segmentée. Chaque segmentation est faite indépendamment des autres afin qu'une erreur éventuelle faite par l'un des modules de segmentation ne soit pas reportée sur les segmentations des autres structures. Chaque module de segmentation a donc été conçu et optimisé spécifiquement pour un type de structure anatomique.

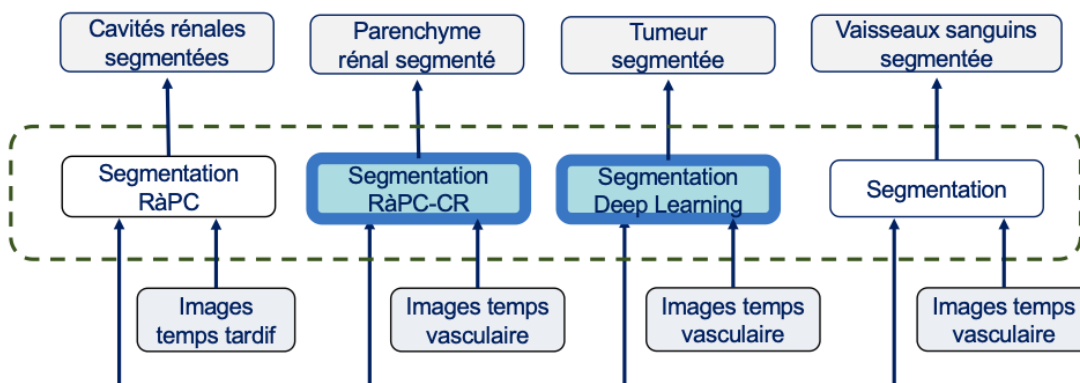


FIGURE 4.3 – Organisation de la couche segmentation de COLISEUM

Les parties anatomiques du rein, parenchyme et cavités, sont segmentées par un procédé de Croissance de Régions (CR) associé à un RàPC. La croissance de régions est un algorithme de segmentation itératif classique en traitement d'images. Il permet de produire une segmentation par incorporations successives des pixels (processus de croissance) à partir d'un ou plusieurs pixels germes correctement choisis au préalable dans l'image. Sur le principe, il se rapproche fortement de l'algorithme de partage des eaux précédemment défini dans la section 3.3.2.1, à la différence près que l'initialisation du processus se fait de manière un peu différente. Au-delà de la conception du système de RàPC pour ce module, la contribution de ce travail de thèse s'est concentrée sur l'implémentation d'une phase d'adaptation capable de modifier la solution remémorée (les paramètres de l'algorithme) afin de proposer une solution optimale maximisant la pertinence du résultat.

Les Réseaux de Neurones Convolutifs (*Convolutional Neural Networks*) interviennent lors de la segmentation du néphroblastome car il est difficilement segmentable par une croissance de régions à cause d'un faible contraste avec les tissus environnants. Cette faiblesse en contraste rend toute tentative de segmentation par cette méthode itérative très vulnérable à un phénomène de « fuite ». Bien qu'il existe dans la littérature des approches permettant de limiter ce phénomène (comme par exemple la croissance de régions avec forme de référence), nous avons fait le choix, dans un souci de simplicité et d'efficacité, de recourir au *Deep Learning* pour cette tâche. Le *Deep Learning* est utilisé ici avec une nouvelle méthode d'apprentissage permettant de diminuer le recours à une large base de données pour l'entraînement du réseau. Cette méthode a été nommée OV²ASSION

(*Overlearning Vector for Valid Sparse Segmentations*).

Les segmentations des cavités rénales ainsi que des vaisseaux sanguins sont un objectif du cahier des charges du projet ASKid qui seront explorées dans les suites des travaux de cette thèse, comme nous le décrirons dans nos perspectives :

- Concernant la segmentation des vaisseaux sanguins, il restera à concevoir un module de segmentation à cette fin. La plus grande difficulté se trouve à la fois dans un contraste pouvant être très faible avec l'environnement immédiat des vaisseaux et dans l'imprévisibilité de leur localisation due à la présence du néphroblastome. Ceci rend les vaisseaux sanguins difficilement discernables, même par un œil humain entraîné.
- Concernant la segmentation des cavités sur les clichés en temps tardif, un protocole basé sur un second système RàPC paraît tout à fait envisageable. Il est toutefois difficile d'appliquer un algorithme de croissance de régions dans cette situation. Contrairement au parenchyme rénal qui peut être en 1, 2 ou 3 parties maximum (sur les images, comme la tumeur peut s'imbriquer dans le rein), il peut y avoir plusieurs parties distinctes formant le rein, les cavités rénales peuvent former sur les images un grand nombre de petites structures distinctes qui exigeraient tout autant de germes à placer pour parvenir à produire la segmentation. Cette approche serait donc assez peu efficace. Il est toutefois possible d'exploiter une particularité des cavités par rapport au parenchyme : celles-ci concentrent une grande partie du produit de contraste en temps tardif, rendant leur intensité de pixels sur les clichés très élevés et distincts de leur environnement. Un simple seuillage serait donc suffisant pour parvenir à effectuer les segmentations à condition de prendre le problème des os en considération. Les niveaux de gris des cavités étant très proches de celui des structures osseuses, un seuillage généralisé sur l'ensemble de l'image inclurait également ces cavités. Le RàPC permettrait d'apporter une solution originale à ce problème en rendant possible la mémorisation d'une fenêtre délimitant une zone de l'image qui contiendrait les cavités. La segmentation serait ainsi le résultat d'un seuillage limité à cette fenêtre. Des tests supplémentaires sont néanmoins nécessaires pour évaluer la fiabilité de cette approche dans la pratique.

4.2.3/ LA COUCHE DE FUSION

Les sorties de la couche de segmentation étant constituées de segmentations de structures différentes faites indépendamment les unes des autres, il est nécessaire de fusionner ces informations pour produire un unique résultat. C'est le rôle de la couche de fusion (voir Figure 4.4). Cette couche, essentielle au bon fonctionnement de la plate-forme, fait l'objet de la thèse en cours de Lisa Corbat. Les fondements de ces travaux, qui font partie de la plate-forme, sont présentés dans ce chapitre.

4.2.3.1/ ARBITRAGE ET GESTION DE CONFLITS

La principale problématique consiste à définir la manière de réaliser cette fusion. Dans la mesure où chaque segmentation produite en amont concerne une structure distincte, la logique voudrait que les informations soient majoritairement complémentaires et donc

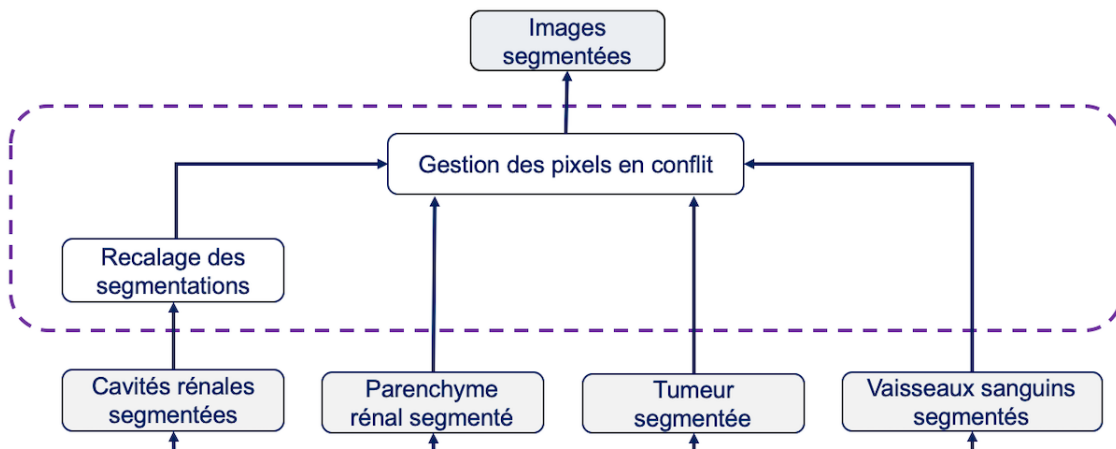


FIGURE 4.4 – Organisation de la couche de fusion de COLISEUM

que leur fusion ne pose aucun problème particulier. Si l'intersection des différentes segmentations est nulle (dit autrement, si un même pixel n'a pas été simultanément associé à des labels de structure différents), une simple union des résultats, est suffisante.

Cependant, dans la pratique, cette intersection sera très rarement vide et il faut donc concevoir un protocole pour la gestion des conflits engendrés : si un même pixel est associé à plusieurs labels différents, comment définir quel label doit être retenu dans la segmentation finale ? Cette gestion des conflits peut se faire suivant différentes stratégies :

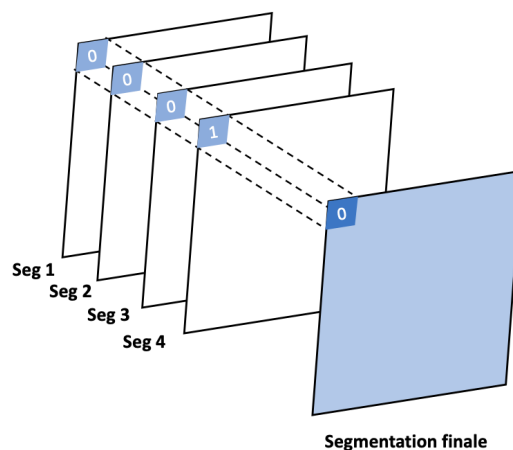


FIGURE 4.5 – Principe d'un vote majoritaire pour arbitrer un conflit entre plusieurs segmentations

- La stratégie la plus simple et la plus intuitive est une approche par vote, que ce soit avec un vote majoritaire (le label le plus présent parmi les différents résultats pour un pixel donné est considéré comme le plus pertinent, voir Figure 4.5) ou encore un vote pondéré (certains résultats sont alors considérés comme plus fiables que les autres et un poids plus important leur est attribué pour la prise de décision). Le principal défaut de cette approche est qu'il s'agit d'une prise de décision exclusivement locale (au niveau du pixel) ne tenant pas compte de la cohérence du résultat.

- D'autres stratégies plus complexes existent également comme une approche morphologique, basée sur des éléments de contours et de forme [rohlfing2007shape, mignotte2014criterionModel], ou encore une approche probabiliste calculant la vraisemblance des labels [xu1992classifiers, nguyen2018fusionModel].

4.2.3.2/ LE PRINCIPE DU RECALAGE D'IMAGES POUR LES SEGMENTATIONS

Il s'agit ici de la partie fusion des images segmentées manuellement par les médecins puisque, comme nous l'avons indiqué précédemment, la segmentation automatique des cavités rénales fera l'objet de prochains travaux.

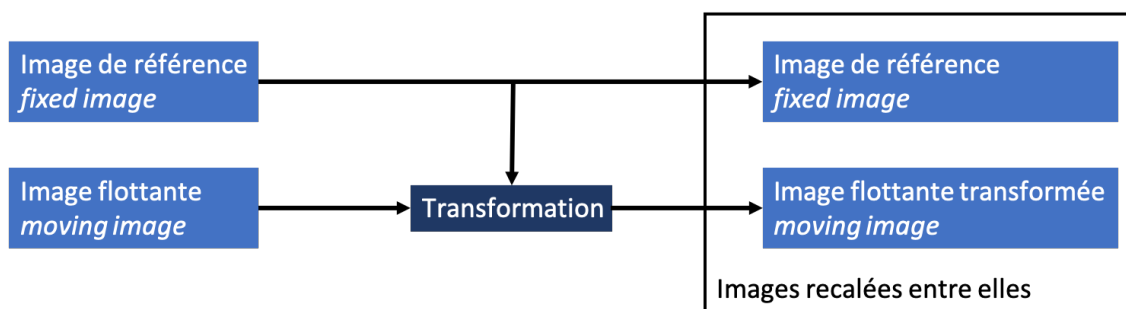


FIGURE 4.6 – Principe général d'un recalage d'images

La segmentation des cavités rénales ne pouvant se faire sur les mêmes images que celles des autres structures d'intérêt, elle est effectuée sur des images en temps tardif pour que le produit de contraste, en cours d'élimination dans les voies excrétrices, permette la mise en évidence de l'appareil urinaire dont les cavités font parties. Ceci implique des différences, pouvant être importantes, dues au mouvement du jeune patient durant l'examen (changement de position, mouvements respiratoires, ...). Afin de pouvoir fusionner les informations issues à la fois des images en temps vasculaire et de celles en temps tardif, il est nécessaire de les faire correspondre dans un seul et même référentiel. Cette étape de mise en correspondance se nomme dans le domaine du traitement d'images « *le recalage* ».

Le recalage d'images consiste en la mise en correspondance de deux images différentes afin de pouvoir les comparer (par exemple, un atlas anatomique) ou afin de combiner leurs informations (par exemple la combinaison des informations anatomique et fonctionnelle). Les images recalées peuvent être issues d'une même modalité (comme par exemple le recalage CT-Scan/CT-Scan) ou de plusieurs modalités différentes (comme par exemple le recalage d'une image IRM avec un CT-Scan).

Un recalage entre deux images se fait toujours en prenant l'une d'elles en temps qu'**image de référence** (*fixed image*). Cette image fixe ne subit aucune transformation durant le processus. La seconde image est qualifiée d'**image flottante** (*moving image*) et sera mise en correspondance avec la première par l'application d'une transformation (Figure 4.6). La technique de recalage consiste à déterminer cette transformation spatiale afin de maximiser la mise en correspondance de l'image flottante sur l'image de référence.

Il existe différentes possibilités de classer les méthodes de recalage, de manière plus ou moins complexes, en fonction des critères retenus (approche géométrique ou iconique,

métrique utilisée, transformations, stratégie d'optimisation, ...). Dans la mesure où l'objectif de cette partie est d'introduire les notions de recalage d'images, nous présentons ici une classification limitée au type de transformations utilisée, ce qui a pour avantage de permettre une distinction claire et facilement compréhensible.

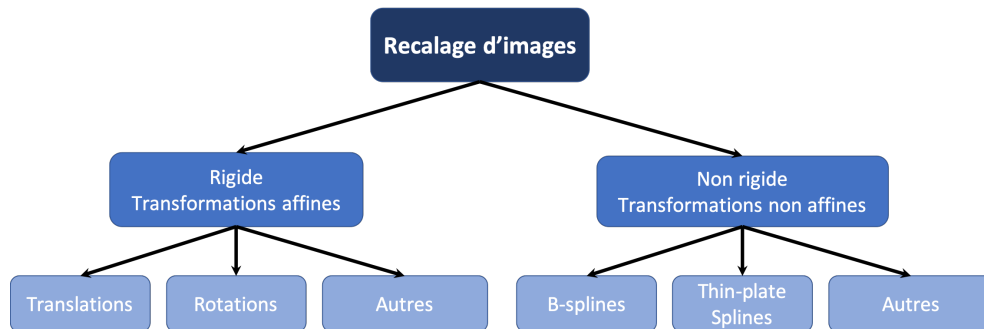


FIGURE 4.7 – Présentation simplifiée des différents types de recalages existants

Comme illustré par la Figure 4.7, deux grands types de recalages d'images, suivant le modèle de transformations utilisé pour mettre en correspondance l'image flottante et l'image fixe, peuvent être distingués :

- **Les recalages rigides** consistent à mettre en correspondance deux images dans l'espace sans induire de déformations de l'image flottante (deux lignes parallèles dans l'image resteront parallèles après la transformation, on parle de transformations linéaires). Il s'agit donc de transformations se limitant à des translations et des rotations. L'intérêt des recalages rigides est d'effectuer une mise en correspondance dans l'espace de deux objets, tout en conservant leur variabilité (différences) de contours. Ils sont donc utilisés lorsque l'on souhaite étudier cette variabilité (comme par exemple l'étude de la variabilité anatomique inter-patient).
- **Les recalages non rigides/élastiques** regroupent toutes les transformations non linéaires qui induisent une déformation de l'image flottante pour maximiser la mise en correspondance avec l'image fixe. En imagerie médicale, ce type de recalage peut être, par exemple, utilisé pour éliminer une variabilité anatomique entre plusieurs patients et ainsi conserver uniquement une variabilité fonctionnelle.

Les différences entre les images en temps vasculaire et celles en temps tardif ne résultant pas uniquement de simples transformations linéaires (translation, rotation), il est nécessaire de recourir à une transformation élastique. Le recalage des segmentations peut se faire de façon simple en appliquant la transformation calculée pour le recalage des images correspondantes. Il est nécessaire de passer par cette étape au préalable afin d'obtenir de bons résultats car les segmentations sont pauvres en informations pour réaliser un recalage correct.

Lors d'un recalage, il est possible de réaliser la mise en correspondance des deux images en utilisant des points de repère (généralement anatomiques). On parle alors d'**une approche géométrique**. Pour un recalage mono-modale (dans lequel les deux images donnent une information comparable), il n'est pas forcément nécessaire de déterminer ces points de repère. L'utilisation d'une métrique sur l'ensemble des pixels des images (calculant une distance en fonction de la valeur des pixels et non pas des coordonnées

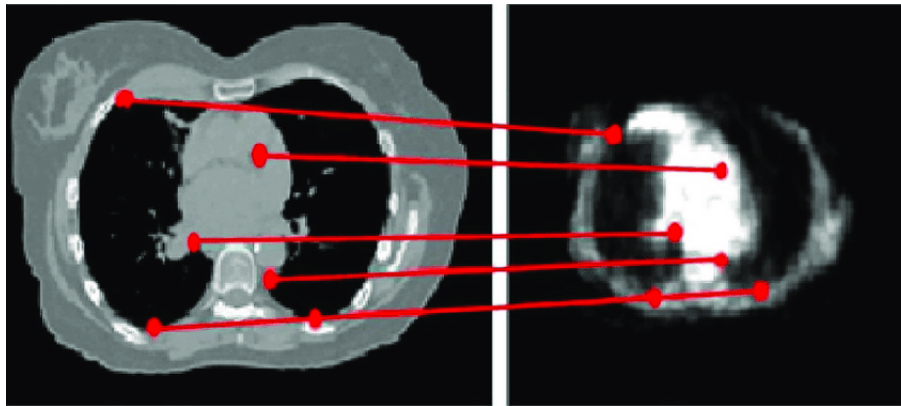


FIGURE 4.8 – Utilisation de points de repère anatomiques pour le recalage d'une image scanner (à gauche), avec une image TEP (Tomographie par Émission de Positions, à droite)

des points de repère) peut s'avérer suffisante. Cette méthode correspond à **une approche iconique**. L'usage de points de repère (*landmarks*) reste toutefois un bon moyen d'améliorer la pertinence du résultat. Pour un recalage multi-modal, ces points de repère deviennent essentiels en l'absence d'autres moyens de guider la mise en correspondance dans la mesure où les intensités de pixels ne sont pas comparables (Figure 4.8).

4.3/ LES FONDEMENTS DE NOS TRAVAUX SUR LE RECALAGE TEMPS VASCULAIRE/TEMPS TARDIF

Dans le cadre du projet ASKid, j'ai encadré les travaux de G. de Oliveira (stage de fin d'étude de l'école d'ingénieurs en génie biomédical ISIFC de Besançon). Nous avons ainsi entamé l'étude du recalage des images prises en temps vasculaire et en temps tardif.

L'exploitation des images pour la caractérisation du rein pathologique par rapport au néphroblastome passe par une étape cruciale de recalage pour mettre en correspondance les données et en extraire l'information voulue. Dans cette optique, le travail préliminaire a permis de réaliser un *framework* qui effectue de manière automatique le recalage entre images en temps vasculaire et images en temps tardif.

Dans les clichés en temps vasculaire et en temps tardif, l'information recherchée, compte tenu de l'objectif du projet ASKid, concerne avant tout le rein (parenchyme et cavités). Cependant, pour faciliter la mise en correspondance des deux types d'images, une référence intéressante se trouve être les structures osseuses, et en particulier le pelvis, pour fournir les différents points de repère requis. Dans un second temps, à partir de ces repères anatomiques, la transformation peut être calculée puis appliquée aux images en temps tardif et aux segmentations associées afin de les faire correspondre aux images en temps vasculaire.

Nous reviendrons dans nos perspectives sur ce framework et sur les voies d'améliorations auxquelles nous avons réfléchi. En particulier, dans cette première version, le recalage est rigide, et comme nous l'avons précédemment indiqué, pour augmenter la précision du recalage « *Les différences entre les images en temps vasculaire et celles*

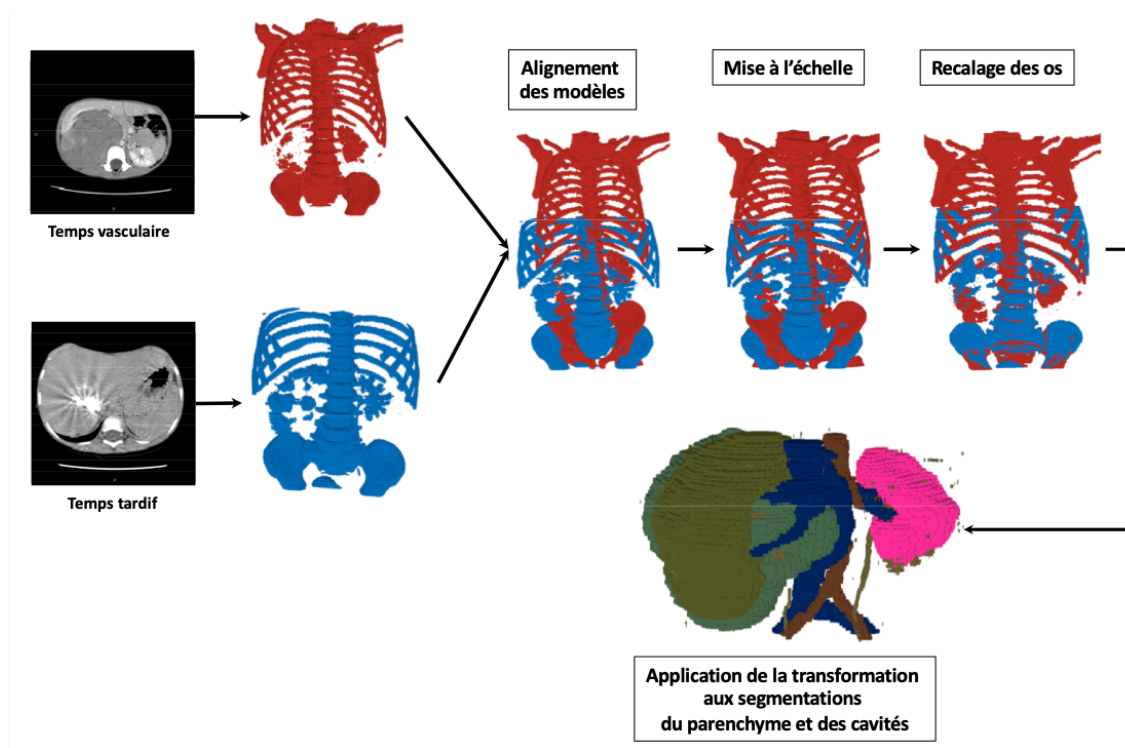


FIGURE 4.9 – *Framework* du recalage automatique des images scanners en temps vasculaire/tardif

en temps tardif ne résultant pas uniquement de simples transformations linéaires (translation, rotation), il est nécessaire de recourir à une transformation élastique ».

4.3.1/ DÉTECTION AUTOMATIQUE DES POINTS DE REPÈRE ANATOMIQUES

Le recalage se base sur un ensemble de 8 points de référence anatomiques situés au niveau du pelvis du patient. Pour chaque hémipelvis les 4 points suivants sont définis :

- 1 point délimitant l'extrémité supérieure,
- 1 point délimitant l'extrémité inférieure,
- 1 point délimitant l'extrémité avant,
- 1 point délimitant l'extrémité arrière.

La première étape pour parvenir à détecter ces points de repère est la segmentation des structures osseuses. Celle-ci se fait par un seuillage avec une valeur de 230 (intensité de pixel déterminée de manière empirique pour cette première version du framework : cette valeur est très dépendante des modalités médicales utilisées lors de l'acquisition, et de l'usage propre à chaque praticien, d'où sa détermination empirique) sur chacune des coupes. Le pelvis est ensuite sélectionné sur un critère de largeur et de volume en fonction de l'âge et de la taille du patient, permettant la détermination des marqueurs anatomiques.

4.3.2/ RECALAGE DES IMAGES UTILISÉ DANS CETTE PREMIÈRE VERSION

Dans le cadre d'un recalage rigide, seules des rotations et des translations sont acceptées. Ainsi le recalage entre un ensemble A et un ensemble B peut être vu comme une transformation linéaire comme suit :

$$B = R.A + Tr \quad (10)$$

où R et Tr désignent les transformations à appliquer pour recaler les points de l'ensemble A dans l'espace de l'ensemble B .

4.3.2.1/ CORRECTION DE LA DIFFÉRENCE D'ÉCHELLE ENTRE LES IMAGES

La première étape du *framework* est l'alignement des deux modèles et la correction d'une éventuelle différence d'échelle entre les deux examens. Le facteur de zoom est déterminé de façon automatique en calculant le ratio de 4 lignes virtuelles formées par les 8 repères anatomiques du pelvis. Ce facteur permet ensuite de calculer les nouvelles coordonnées des points.

4.3.2.2/ DÉTERMINATION DE LA ROTATION ET DE LA TRANSLATION

Dans le *framework* actuel, le recalage osseux s'effectue en suivant la méthode SVD (*Singular Values Decomposition*). Il s'agit d'une méthode mathématique permettant de calculer une transformation à partir d'une correspondance entre des points.

Tout d'abord, les centroïdes des deux ensembles sont calculés, puis les jeux de données sont recentrés sur leur centroïde respectif. Puis, le calcul de la transformation à appliquer pour le recalage est effectué en deux étapes :

1. La détermination de la matrice de rotation optimale,
2. La détermination de la matrice de translation.

La méthode SVD intervient dans le calcul de la matrice de rotation. Dans un premier temps, la matrice de covariance H peut être déterminée selon l'expression suivante :

$$H = \sum_{i=1}^N (p_A^i - c_A)(p_B^i - c_B)^T \quad (11)$$

Où :

- c_A et c_B sont respectivement les centroïdes des images A et B ,
- N est le nombre de pixels à recaler dans les deux images,
- les p^i sont les coordonnées du $i^{\text{ème}}$ pixel (il s'agit donc d'un vecteur)
- et la matrice de covariance H , par l'intermédiaire d'une décomposition en valeurs singulières, donne directement accès à la matrice de rotation.

Définition

Soit une matrice A , sa décomposition en valeurs singulières (SVD) consiste à la décomposer en 3 matrices distinctes telles que :

$$A = U.D.V^T$$

Où :

- U et V sont des matrices orthogonales, respectivement nommées « *vecteurs singuliers gauches* » et « *vecteurs singuliers droits* »,
- et D est une matrice diagonale.
 D est composée des valeurs dites singulières de A .

La matrice A peut être vue comme une application linéaire pouvant être décomposée en 3 sous-transformations que sont une rotation, une déformation et une rotation. Ces 3 étapes correspondent aux matrices U , D et V .

En utilisant la méthode *SVD* afin de décomposer la matrice de covariance H en trois matrices U , D et V , il est possible de calculer :

- la matrice de rotation à appliquer pour effectuer le recalage telle que :

$$R = V.U^T \quad (12)$$

- ainsi que la matrice de translation Tr telle que :

$$Tr = -R \times c_A + c_B \quad (13)$$

Les résultats préliminaires ont été fortement encourageants mais demandent de plus amples recherches pour améliorer la précision du recalage : nous y reviendrons dans les perspectives de ce travail. Une amélioration consisterait notamment à tenir compte de la respiration du patient dans le processus, ce qui n'a pour le moment pas été traité de manière efficace. Ceci passe par l'usage d'un recalage élastique pour compenser la déformation des structures abdominales due à l'abaissement du diaphragme. Plutôt que de recourir à une méthode de décomposition en valeurs singulières, l'utilisation d'un programme pour calculer de manière itérative (en relation avec une métrique comme décrit dans la section 4.2.3.2) la meilleure transformation à appliquer serait plus performante car elle permettrait de tenir compte des intensités de pixels de l'image en plus des repères anatomiques.

Synthèse du chapitre

La plate-forme COLISEUM-3D a pour objectif de faire collaborer plusieurs approches d'Intelligence Artificielle (IA) optimisées pour réaliser une segmentation d'un ensemble de structures abdominales d'intérêt qui soit médicalement exploitable.

Son organisation a été formalisée sous la forme de différentes couches et modules assurant des fonctions complémentaires. La couche de segmentation prend en entrée les images en temps vasculaire et en temps tardif pour produire les segmentations de chaque élément, de manière indépendante. Le module de segmentation du parenchyme rénal utilise une approche par Raisonnement à Partir de Cas, couplé à un algorithme de croissance de régions. Le module dédié à la segmentation du néphroblastome fait appel, quant à lui, à un réseau de neurones convolutionnel entraîné suivant un nouveau protocole permettant de s'affranchir d'une large base de données pour la phase d'apprentissage. Les modules chargés de la segmentation des cavités rénales et des vaisseaux sanguins seront traités lors de prochaines thèses au sein de l'équipe de recherche. La dernière couche du *framework* de la plate-forme est la couche de fusion chargée de mettre en commun les différentes segmentations. Elle se compose d'une étape de recalage, pour mettre en correspondance les segmentations issues des images dans les deux temps vasculaire et tardif, et d'une étape finale de gestion des conflits, pour produire la segmentation finale unique. L'implémentation de cette couche de fusion dépasse le cadre de ce manuscrit et ne sera pas abordée par la suite.

Dans le chapitre suivant, nous aborderons nos deux contributions principales qui sont les conceptions d'un système de Raisonnement à Partir de Cas (RàPC) pour la segmentation du parenchyme rénal ainsi que d'une méthode d'entraînement originale pour un réseau de neurones artificiels profond afin de segmenter le néphroblastome.

CONTRIBUTIONS MAJEURES SUR LA SEGMENTATION EN UTILISANT L'INTELLIGENCE ARTIFICIELLE D'UNE PART AVEC LE RÀPC ET D'AUTRE PART AVEC LE DEEP LEARNING

Ce chapitre présente plus précisément les 2 grandes contributions de ce travail de thèse. La première a consisté à concevoir un système de RàPC dédié à la segmentation du parenchyme rénal (en tenant compte des déformations dues à la pathologie) par un algorithme de croissance de régions. Cette contribution, en plus de l'implémentation du système de segmentation lui-même, s'est en particulier concentré sur la conception d'une phase d'adaptation efficace pour maximiser la qualité des segmentations produites tout en minimisant la quantité de données nécessaires.

La seconde contribution s'est orientée vers une approche de type *Deep Learning* pour réaliser la segmentation du néphroblastome. Pour pallier le manque de données inhérent au domaine du néphroblastome, formant la principale limite des réseaux de neurones convolutifs (CNN), nous suivons une méthode d'entraînement particulière permettant d'obtenir un réseau spécifique au patient à traiter. À partir d'un échantillon représentatif de coupes segmentées manuellement, nous entraînons notre CNN pour calculer les segmentations des coupes restantes. Nous avons nommé cette approche *Overlearning Vector for Valid Sparse Segmentations (OV²ASSION)*.

5.1/ RAISONNEMENT À PARTIR DE CAS : APPLICATION À UN PROBLÈME DE SEGMENTATION D'IMAGES PAR CROISSANCE DE RÉGIONS

L'Intelligence Artificielle est un puissant outil à disposition des programmeurs, mais elle a généralement pour principale limite que sa performance est fortement dépendante de la quantité et de la qualité des données à disposition. Une approche connaissance permet, comme nous l'avons détaillé dans le Chapitre 3, de limiter cette dépendance aux données. Notre objectif est d'introduire un processus basé sur les connaissances et les

expériences d'experts afin de réaliser un processus de segmentation du parenchyme rénal. Nous avons donc considéré le Raisonnement à Partir de Cas (RàPC) comme une approche adaptée à ce type de problème particulier. En parallèle, les caractéristiques du parenchyme sur des images scanner (mises en évidence par l'usage d'un produit de contraste durant l'examen médical) rendent l'usage d'un algorithme de croissance de régions efficace pour produire une segmentation. Cette section présente un système de RàPC capable de réaliser la segmentation de parenchyme rénal déformé par la présence d'un néphroblastome.

L'originalité de notre approche RàPC réside dans la manière dont notre système adapte les solutions à partir de situations passées. Le principe d'une segmentation par croissance de régions consiste à placer des germes à différents emplacements dans l'image devant être segmentée, puis à vérifier que les pixels environnants de ces germes ont un niveau de gris suffisamment proche pour être intégrés dans la même région. Cette intégration se fait suivant un seuil local (différence entre le pixel candidat à l'incorporation dans la région et son voisinage) et un seuil global (différence entre le pixel candidat et la moyenne de la région à intégrer). Cet algorithme permet à chaque région de croître jusqu'à ce que chaque pixel dans l'image appartienne à une et une seule région. Le principal problème avec la méthode de segmentation par croissance de régions est qu'un utilisateur doit placer manuellement les germes dans l'image pour parvenir à produire un résultat pertinent. Des méthodes automatiques existent mais ne sont pas assez fiables et elles ont une forte tendance à aboutir à une sur-segmentation (division de l'image en un nombre trop important de régions). Notre idée consiste à utiliser un système de RàPC capable de placer les germes aux bons endroits, ce qui permet ainsi de réaliser la segmentation dans les meilleures conditions. De plus, notre principale contribution consiste en l'élaboration d'une étape d'adaptation pour que le système puisse de lui-même modifier les valeurs des paramètres de l'algorithme pour trouver une combinaison optimale : *la mieux adaptée possible au nouveau problème*.

Dans cette section, le système est présenté dans son ensemble dans la Figure 5.1. Toutes les phases sont expliquées de manière précise. Notre système récupère en entrée des images CT-scans et il recherche l'image déjà segmentée, stockée dans la base de cas, qui soit la plus proche. Pour cette recherche, il calcule une valeur de similarité pour chaque cas stocké et, durant une phase de remémoration, il extrait le cas source ayant la plus haute similarité avec le problème cible. Puis, les paramètres de segmentation extraits sont adaptés au cas courant grâce à une phase d'adaptation. Ces paramètres adaptés sont utilisés pour réaliser une nouvelle segmentation. Finalement, le résultat est évalué par un expert lors d'une phase de révision puis il est stocké dans la base de cas en tant que nouveau cas source, mais uniquement si cet ajout est pertinent. Cette pertinence est déterminée par une phase de capitalisation qui gère la maintenance de la base de cas et constitue la dernière étape du cycle.

5.1.1/ L'UNITÉ DE SEGMENTATION

L'unité de segmentation se base sur un algorithme très classique en traitement d'images : la croissance de régions. Cet algorithme consiste à placer dans l'image des germes (ou *seeds*). Ces germes correspondent à des pixels choisis pour être à l'intérieur d'objets d'intérêt que l'on souhaiterait segmenter et qui forment les régions initiales. Puis, une croissance de régions suit un processus itératif visant à faire progressivement croître

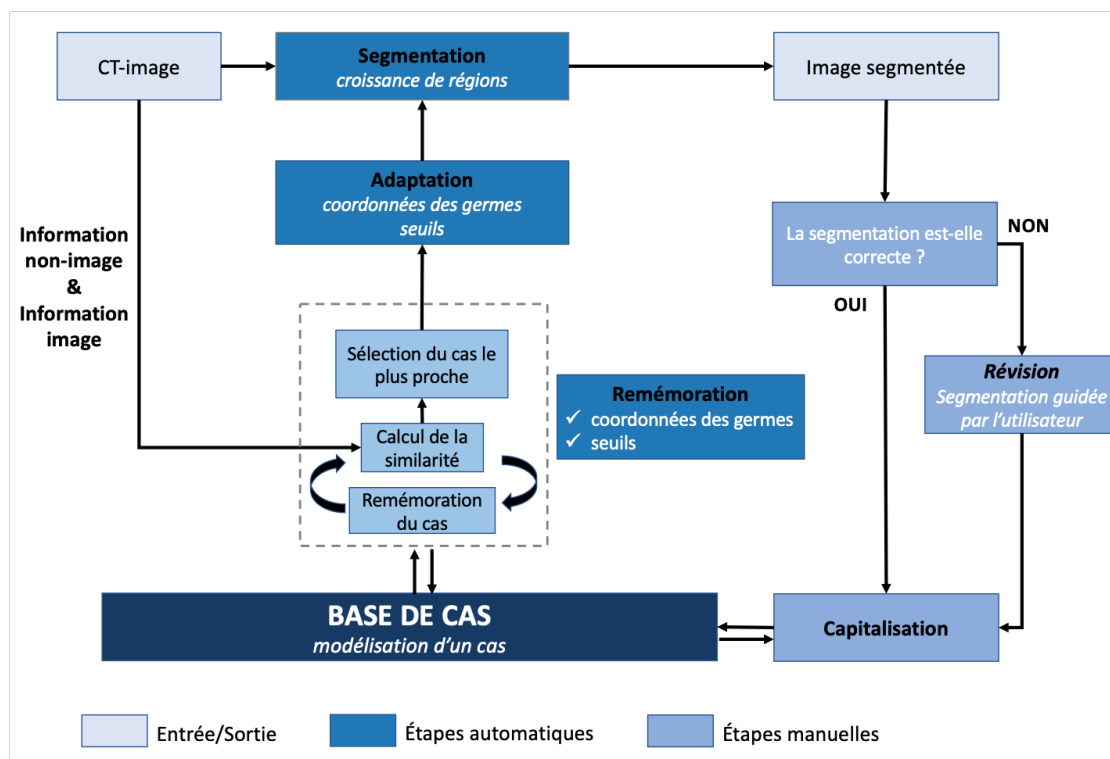


FIGURE 5.1 – Vue d'ensemble de notre système de RàPC pour la segmentation

les régions jusqu'à convergence (dit autrement, jusqu'à ce que de nouvelles itérations ne changent plus le résultat). À chaque itération, les pixels voisins d'une région donnée (en connexité 4 ou 8) sont évalués suivant un ou plusieurs critère(s) calculé(s) à partir de leur valeur d'intensité et comparé(s) à une ou plusieurs valeur(s) seuil(s). La Figure 5.2 en illustre le principe avec une forme simple. Dans cet exemple, l'algorithme parvient à produire une segmentation en 4 itérations à partir d'un germe placé au centre de l'objet d'intérêt.

Dans notre implémentation, le guidage de l'algorithme se fait suivant deux critères, ϕ_{local} et ϕ_{global} , associés à deux seuils distincts, γ_{local} et γ_{global} . Le critère ϕ_{local} correspond à une différence entre la valeur du pixel candidat et la valeur du pixel voisin appartenant à la région. Ce critère est ensuite comparé à un γ_{local} . Et le critère ϕ_{global} est un critère global correspondant à la différence entre la valeur du pixel candidat et la moyenne de la région et il est quant à lui comparé au seuil global γ_{global} . Si les 2 tests sont validés, alors le pixel candidat est incorporé dans la région et le prochain pixel est évalué. Un test est validé si le critère est inférieur ou égal au seuil correspondant.

L'utilisation d'une double contrainte (locale et globale) permet de limiter l'un des défauts majeurs d'une croissance de régions : sa sensibilité au phénomène de fuite. Dans la pratique, les segmentations se font sur des images bien plus complexes que celle présentée dans la Figure 5.2, dans lesquelles les contrastes entre l'objet d'intérêt et le reste de l'image ne sont pas aussi nets. Dans ces cas, une simple contrainte sur la différence entre le pixel candidat et le pixel de la région devient insuffisante et peut mener à une fuite du processus vers des pixels non pertinents (dans le cas d'un dégradé par exemple). Une deuxième contrainte sur la différence entre la valeur du pixel candidat et la moyenne de la région déjà calculée permet une amélioration sensible des résultats.

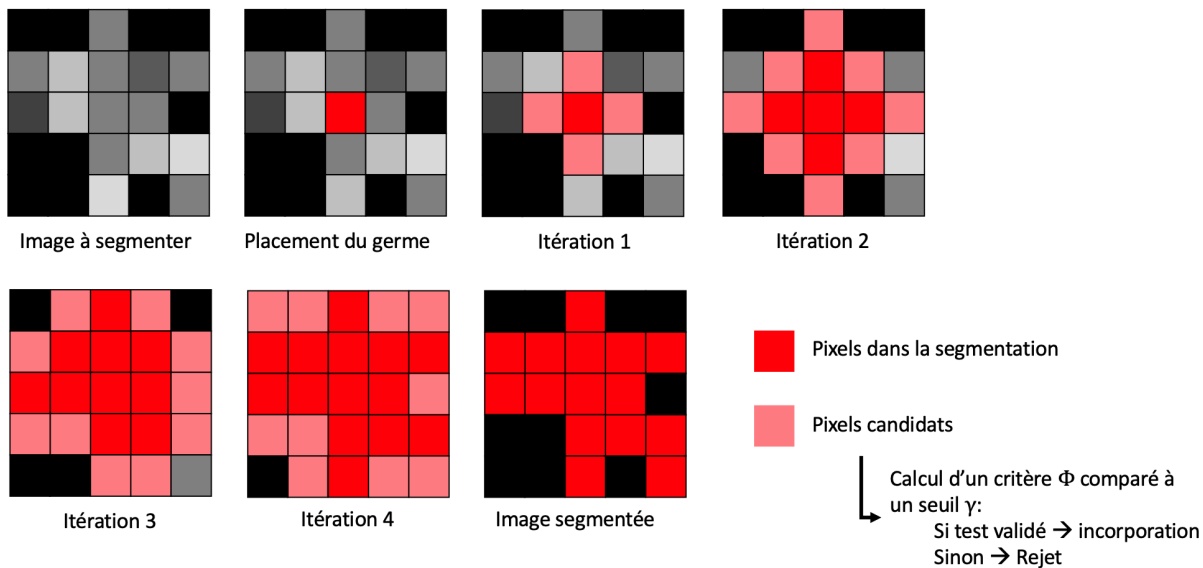


FIGURE 5.2 – Fonctionnement d'un algorithme de croissance de régions

5.1.2/ LA MODÉLISATION D'UN CAS

Un cas est composé de deux parties distinctes : un problème et une solution associée. La partie problème décrit les caractéristiques du problème à résoudre et la partie solution fournit le moyen de parvenir à sa résolution. La modélisation d'un cas pour le système de RàPC est décrite dans la Figure 5.3, avec à gauche la description du problème et à droite la description de la solution.

Dans notre implémentation, la partie problème doit décrire les images scanner (CT-images), c'est à dire qu'elle doit fournir les données informant sur la structure et les statistiques des images. Quant à la partie solution, elle doit donner l'ensemble des paramètres nécessaires à la réalisation d'une segmentation optimale par croissance de régions.

En se basant sur l'approche de A. Attig et P. Perner ([attig2009study], [perner2011knowledge], [attig2012watershedSeg]), la partie problème de nos cas est composée :

- Des méta-données représentant les informations à propos du patient : le sexe, l'âge, la taille et le poids ;
- Des informations statistiques de l'image : la moyenne, le kurtosis, l'asymétrie, la variance et l'écart type.

La partie solution de nos cas contient la localisation des germes pour initialiser l'algorithme, c'est à dire leurs coordonnées, les valeurs de seuils (local et global) pour chacun des germes placés dans l'image à segmenter, ainsi qu'une liste des prétraitements à appliquer sur l'image avant la segmentation pour en maximiser la pertinence (donnant l'ordre d'application des filtres et leurs paramètres). Chaque image ayant ses propres propriétés et particularités, il est nécessaire d'ajouter le prétraitement (qui est dépendant de l'image traitée) à la solution afin de s'adapter au mieux à chaque cas. En outre, nous avons également ajouté 3 autres éléments supplémentaires relatifs à la segmentation du

cas stocké (décrivant la segmentation réalisée lors de la résolution ce problème) : la superficie, le centre de masse et l'orientation de la forme segmentée. Ceci est nécessaire pour la phase d'adaptation des seuils, présentée dans la section 5.1.4.2.

$$\text{Cas} = \begin{bmatrix} \text{Sexe du patient} \\ \text{Age du patient} \\ \text{Taille du patient} \\ \text{Poids du patient} \\ \text{Moyenne de l'image} \\ \text{Kurtosis de l'image} \\ \text{Asymetrie de l'image} \\ \text{Variance de l'image} \end{bmatrix} + \begin{bmatrix} \text{Coordonnees 2D des germes} \\ \text{Valeurs de seuils pour chaque germe} \\ \text{Liste des pretraitements à appliquer} \\ \text{Superficie de la segmentation} \\ \text{Centre de masse de la segmentation} \\ \text{Orientation de la segmentation} \end{bmatrix}$$

Description de la partie problème **Description de la partie solution**

FIGURE 5.3 – La modélisation d'un cas dans notre système de RàPC : partie problème et partie solution

Nous utilisons deux prétraitements différents afin de faciliter la segmentation et de limiter au mieux le phénomène de fuite : un filtrage médian et un masque flou (plus connu sous le terme de *unsharp mask*). Un filtre médian est un filtre dit « passe-bas », bloquant les hautes fréquences. Son effet est d'éliminer les détails fins de l'image pour ne conserver que les informations globales (homogénéisation), mais au prix d'une diminution des contrastes. La technique du masque flou a un cheminement inverse et vise à accentuer artificiellement les contrastes de l'image dans des cas pour lesquels ils doivent être améliorés pour le bon déroulement de la segmentation.

L'augmentation des contrastes se fait en appliquant sur l'image à traiter un filtre passe-bas (tel qu'un filtre médian), en soustrayant l'image floutée à l'image originale et en ajoutant en dernière étape le résultat de cette soustraction à l'image originale. La Figure 5.4 illustre l'application d'un masque flou sur un exemple afin d'augmenter les contrastes de l'image.

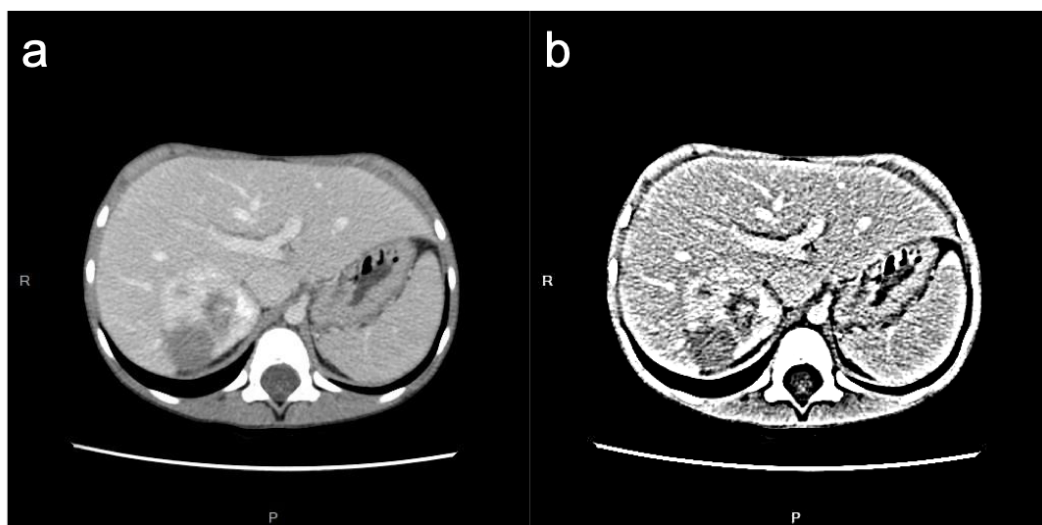


FIGURE 5.4 – Illustration de l'application d'un masque flou : (a) CT-scan (b) image après application d'un masque flou

5.1.3/ LA PHASE DE REMÉMORATION

Cette section décrit la démarche utilisée par le système pour effectuer l'étape de remémoration en utilisant un calcul de similarité entre les cas stockés dans la base et le cas à résoudre.

Comme nous l'avons expliqué dans la section précédente, la base de cas du système est composée de deux types de données : les méta-données (concernant le patient) et les caractéristiques de l'image (cf 5.1.2). Deux variantes de la formule de similarité ont été testées entre 2 images x et y . Pour la première, nous avons utilisé la formule de P. Perner décrite dans [perner1999segCBR], appelée ici $s_1(x, y)$ dans l'équation 14 avec une composante $SM_d(x, y)$ pour les méta-données (une mesure de la similarité de Tversky [tversky1977similarity]) et une autre $SI(x, y)$ dédiée à l'information de type image.

La seconde formule est une formule hybride nommée $s_2(x, y)$ qui est présentée dans l'équation 15. Elle reprend la composante de P. Perner pour les méta-données mais remplace la composante des caractéristiques de l'image par le critère MSSIM (*Mean Structural SIMilarity* : $MSSIM(x, y)$) provenant des travaux de Wang *et al.* [wang2004similarity].

Le critère MSSIM est une amélioration du critère SSIM (*Structural Similarity*) qui est couramment utilisé en compression d'images. MSSIM utilise un fenêtrage itératif (fenêtre glissante) pour augmenter la pertinence de la comparaison structurale. Chaque fenêtre décrit une RoI (*Region of Interest*) dans chacune des 2 images. Ces dernières sont comparées indépendamment pour produire une mesure de similarité pour chaque fenêtre. L'ensemble des similarités permet ensuite de calculer une similarité moyenne pour toute l'image.

Les formules suivantes montrent la construction des 2 calculs de similarité entre les 2 images x et y :

$$s_1(x, y) = \frac{1}{2}(SM_d(x, y) + SI(x, y)) \quad (14)$$

$$s_2(x, y) = \frac{1}{2}(SM_d(x, y) + MSSIM(x, y)) \quad (15)$$

La composante pour comparer les méta-données, commune aux 2 formules, est calculée comme suit :

$$SM_d(x, y) = \frac{|A_i|}{\alpha|A_i| + \beta|D_i| + \gamma|E_i|} \quad (16)$$

Où :

- les A_i représentent le nombre de caractéristiques communes entre x et y ,
- les D_i représentent les caractéristiques propres à x ,
- les E_i représentent les caractéristiques propres à y .
- α , β et γ sont des facteurs poids définis ainsi $\alpha = \beta = 1$ et $\gamma = 0.5$. Les valeurs de poids proviennent des travaux de P. Perner dédiés à son architecture RàPC pour la segmentation d'images ([perner1999segCBR]), que nous avons reprises.

La composante dédiée à la comparaison des caractéristiques de l'image $SI(x, y)$ dans la formule $s_1(x, y)$ a pour expression :

$$SI(x, y) = \frac{1}{K} \sum_{i=1}^K w_i \left| \frac{C_{ix} - C_{imin}}{C_{imax} - C_{imin}} - \frac{C_{iy} - C_{imin}}{C_{imax} - C_{imin}} \right| \quad (17)$$

Où :

- C_{ix} et C_{iy} sont les $i^{\text{ème}}$ caractéristiques des 2 images x et y ,
- C_{imin} et C_{imax} sont respectivement le minimum et le maximum de la $i^{\text{ème}}$ caractéristique dans la base de cas,
- w_i est un facteur poids permettant de pondérer l'influence d'une caractéristique par rapport aux autres. Dans notre implémentation, nous avons fait le choix de ne privilégier aucune d'entre eux et de définir tous les poids tels que $w_i = 1$.

La composante image $MSSIM(x, y)$ dans $s_2(x, y)$ est définie ainsi :

$$MSSIM(x, y) = \frac{1}{M} \sum_{i=1}^M SSIM(x_i, y_i) \quad (18)$$

Où :

- M est le nombre de fenêtres parcourues par l'algorithme.

Le critère SSIM se calcule de la manière suivante :

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1) + (2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (19)$$

Où :

- μ est la moyenne,
- σ est l'écart-type.
- $C_1 = K_1 * L$ et $C_2 = K_2 * L$,

avec $L = 255$ correspondant à l'intervalle dynamique des valeurs de pixels (255 pour une image 8 bits) et $K_1 = 0.01$ et $K_2 = 0.03$. Les valeurs K_1 et K_2 sont issues directement des travaux de Wang *et al.* [wang2004similarity].

5.1.4/ LA PHASE D'ADAPTATION

La phase d'adaptation a pour but de modifier automatiquement les paramètres de la croissance de régions afin de maximiser la pertinence du résultat. Conformément à l'algorithme présenté dans la section 5.1.1, il y a donc deux grands types de paramètres à modifier : les coordonnées des germes (servant à l'initialisation de l'algorithme) et les

valeurs de seuils (contrôlant la propagation/croissance des régions). Une modification de ces paramètres, même minimale, peut engendrer une grande différence concernant le résultat. Il est donc primordial de parvenir à créer une phase d'adaptation suffisamment robuste pour assurer l'efficacité du système. Le nombre de germes n'est quant à lui pas modifié et le système utilisera uniquement ceux issus de la remémoration.

5.1.4.1/ L'ADAPTATION DE LA POSITION DES GERMES

Une partie de l'adaptation consiste à correctement placer les germes dans l'image. Dans la mesure où la forme et la position de la tumeur sont imprévisibles, la localisation du rein est vulnérable à une erreur lors du processus de remémoration. De plus, le rein peut représenter sur certaines coupes une région de faible superficie, surtout lorsque la tumeur le comprime. En conséquence, le cas remémoré n'est pas toujours suffisamment proche du cas à résoudre pour que la localisation du germe soit directement correcte, même avec une similarité élevée. Cette position doit donc être adaptée aux besoins, en particulier pour les régions de faible superficie dans lesquelles le germe peut facilement se trouver en dehors de la zone à segmenter.

Nous partons du principe qu'après la phase de remémoration, le germe se trouve soit à l'intérieur du parenchyme rénal (il n'y a dans ce cas, pas besoin d'adaptation) soit proche de celui-ci. L'adaptation des coordonnées se fait en inférant la position du germe suivant un critère d'intensité des pixels. Après l'application d'un prétraitement identique nous partons du principe que l'intensité moyenne du parenchyme pour le cas remémoré et celle pour le cas à résoudre sont similaires. Nous avons défini un « *intervalle de cohérence* » IC pour chaque germe, correspondant à un intervalle de niveau de gris dans lequel le pixel accueillant un germe doit se situer. En effet, le parenchyme rénal n'est pas toujours homogène et il n'est pas systématiquement possible de retrouver une valeur de pixel strictement égale à celle utilisée pour le cas source afin d'accueillir ledit germe, d'où l'utilisation d'un intervalle de recherche plutôt que d'une valeur fixe. De plus, le parenchyme peut également être non connexe et se retrouver sur une coupe en plusieurs parties. Il est donc parfois nécessaire de recourir à l'usage de plusieurs germes, ayant chacun leur propre intervalle IC , afin de produire une segmentation complète.

Avec le test suivant, nous pouvons définir une procédure pour vérifier si la valeur d'un pixel germe appartient ou non à cet intervalle :

$$\forall \text{germe}, \text{bonnePosition}(\text{germe}) = \text{vrai} \text{ Si } I(\text{germe}) \in IC_{\text{germe}} \quad (20)$$

$$\text{avec } IC_{\text{germe}} = [i_{\text{germe}} - z, i_{\text{germe}} + z]$$

Où :

- $I(\text{germe})$ correspond à l'intensité du pixel sur lequel est placé le germe,
- IC_{germe} est l'intervalle de cohérence pour le rein que nous avons fixé autour d'une valeur de référence i_{germe} ,
- z est l'itération et peut être également vue comme un degré de relâchement.

Lors de la remémoration, chaque germe est associé à une valeur de référence i_{germe} afin de tenter de reproduire au mieux la configuration ayant permis de produire un résultat

satisfaisant. Cette valeur de référence correspond à l'intensité du pixel ayant accueilli le germe lors de la résolution du cas stocké. Dans la mesure où les images ne sont pas identiques, il ne peut pas être garanti qu'un pixel ayant exactement cette valeur puisse être trouvé. Pour pallier ce problème, la recherche d'un pixel candidat correspondant au critère se fait de manière itérative. À chaque itération z , si aucun pixel satisfaisant cette condition n'a été trouvé dans la zone de recherche, l'algorithme relâche un peu plus la contrainte sur l'intervalle de cohérence. De manière à ne pas placer le germe n'importe où dans l'image, mais également de manière à limiter le temps de calcul nécessaire, la recherche se fait dans une fenêtre 50×50 autour de la position initiale du germe. L'usage de cette fenêtre est également en accord avec l'une des hypothèses fortes pour cette phase d'adaptation : l'erreur sur la position du germe est supposée comme étant faible et la modification de ses coordonnées doit rester limitée.

À partir de ce test, nous avons défini un algorithme capable de modifier les coordonnées d'un germe, de manière automatique, pour que celui-ci soit placé à une position correcte (c'est à dire au niveau d'un pixel appartenant au parenchyme rénal). La Figure 5.5 illustre l'extension progressive du voisinage dans lequel rechercher un meilleur pixel candidat à partir de la position initiale du germe. Le principe est d'évaluer dans le voisinage du germe (en connectivité 8) l'intensité des pixels en vérifiant son appartenance à l'intervalle de cohérence décrit dans l'équation 20. Si aucun pixel candidat ne satisfait la condition sur l'intensité, le voisinage est étendu jusqu'à trouver un pixel ayant une intensité cohérente. L'extension de la recherche se fait dans les limites données par la fenêtre 50×50 . Dans la Figure 5.5, chaque carré représente un pixel de l'image. Durant la première itération, les voisins du germe sont évalués. Ne trouvant aucun pixel satisfaisant, une deuxième itération permet d'étendre la recherche à une distance de 2 du germe et

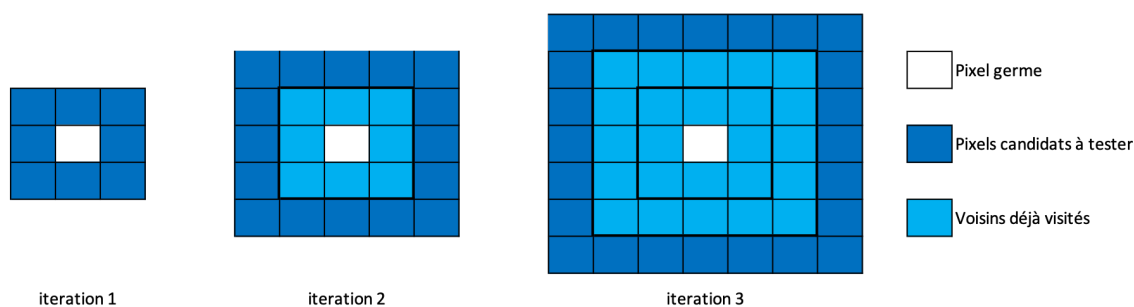


FIGURE 5.5 – Vue d'ensemble de l'évaluation des pixels voisins pour l'adaptation de la position d'un germe

Soit un germe G (avec ses coordonnées 2D dans l'image $G.x$ et $G.y$), sa valeur d'intensité associée $G.int$, et z le degré de relâchement de la contrainte sur l'intensité, notre algorithme retourne les coordonnées 2D adaptées de G . Pour chaque voisin évalué, le test présenté plus tôt est réalisé pour vérifier si l'intensité du pixel est cohérente ou non avec la plage de valeurs recherchées. À chaque itération, la portée de recherche, est augmentée tout en restant au sein d'une fenêtre de recherche 50×50 .

La Figure 5.6 montre un exemple de l'exécution de cet algorithme. Soit une valeur d'intensité de référence associée au germe G de 224 et un intervalle IC se limitant à cette même valeur (première exécution sans relâchement de la contrainte sur l'intensité). À partir de la position initiale du germe (en blanc), le périmètre de recherche est étendu 2

fois jusqu'à trouver un pixel candidat appartenant à l'intervalle de cohérence, de manière à ce que celui-ci soit compris dans une région de l'image appartenant au rein. Lors d'une exécution en conditions réelles, il existe une probabilité significative de ne pas parvenir à trouver un pixel candidat ayant exactement la même intensité que l'intensité de référence associée au germe. Dans ce cas, le processus est répété plusieurs fois, avec un relâchement progressif de la contrainte sur l'intensité (élargissement de IC), jusqu'à trouver une meilleure position pour le germe. Cette méthode permet de placer les germes dans la nouvelle image à segmenter dans une configuration la plus proche possible de celle utilisée pour la segmentation du cas remémoré, bien qu'ayant un coût en temps de calcul pouvant être très supérieur à celui obtenu avec un intervalle IC fixe.

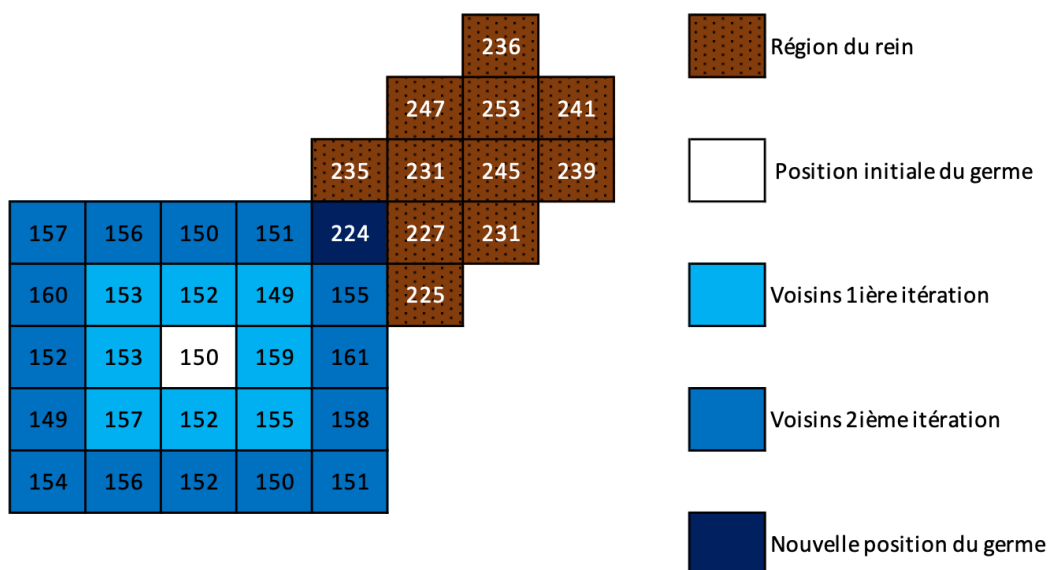


FIGURE 5.6 – Exemple d'une exécution de l'algorithme d'adaptation de la position d'un germe

5.1.4.2/ L'ADAPTATION DES VALEURS DE SEUILS

L'algorithme de croissance de régions étant très sensible à l'initialisation (des conditions initiales différentes ont beaucoup d'impact sur le résultat), il est essentiel d'adapter la position des germes. Mais cette adaptation ne suffit pas à garantir la qualité de la segmentation calculée et il est courant que même avec de bonnes coordonnées, la croissance des germes aboutisse à un résultat aberrant. Le problème est que l'adaptation de la position ne permet pas de prévenir le phénomène de fuite dans des images complexes et parfois peu contrastées. Si la localisation des germes y joue un rôle, la sensibilité de l'algorithme à ce phénomène dépend fortement des valeurs de seuils utilisées. Une mauvaise valeur de seuil, même très proche du seuil optimal, peut entraîner une segmentation très éloignée du résultat attendu. Il est donc nécessaire d'adapter non seulement les seuils associés aux germes mais également leurs coordonnées.

L'adaptation des valeurs des seuils revient à optimiser les valeurs de 2 seuils (local et glo-

bal) par germe. Chaque valeur de seuil a une influence sur le résultat de la segmentation en fonction des autres valeurs, que ce soit la seconde valeur (associée au même germe) ou celles associées aux autres germes. Effectuer l'adaptation consiste donc à déterminer une combinaison optimale de valeurs de seuils pour l'ensemble des germes utilisés pour la segmentation. Ceci implique de concevoir une fonction quantifiant la qualité d'une segmentation (ou l'erreur) afin de pouvoir la maximiser (ou respectivement la minimiser).

Les critères d'évaluation

Cette fonction se base sur 3 critères différents permettant de caractériser la segmentation calculée par rapport à la segmentation du cas remémoré servant de référence. Ces 3 critères correspondent au calcul des 3 premiers moments géométriques. Un moment géométrique d'ordre ij , pour une image dans laquelle chaque pixel a pour coordonnées (x, y) et pour valeur $I(x, y)$, a pour expression :

$$m_{ij} = \sum_x \sum_y x^i y^j I(x, y) \quad (21)$$

Le premier critère m_{00} est le moment d'ordre 0. En reprenant l'expression 21, nous aboutissons à :

$$m_{00} = \sum_x \sum_y I(x, y) \quad (22)$$

Dans une segmentation dans laquelle les valeurs de pixels sont binaires (0 pour le *background* et 1 pour l'objet segmenté), ceci revient à calculer la superficie, en nombre de pixels, de la segmentation.

Le deuxième critère est le centre de masse, ou centre d'inertie, C_m ayant pour coordonnées (\bar{x}, \bar{y}) :

$$\bar{x} = \frac{m_{10}}{m_{00}} \quad \bar{y} = \frac{m_{01}}{m_{00}} \quad (23)$$

Enfin, le troisième critère θ permet de caractériser l'orientation de la segmentation dans l'espace et fait cette fois appel au moment d'ordre 2 :

$$\theta = \frac{1}{2} \arctan \left(\frac{2m_{11}}{m_{20} - m_{02}} \right) \quad (24)$$

La fonction de score

Une fois ces critères définis, nous pouvons construire une fonction *score* permettant d'évaluer la qualité d'une segmentation. Cette fonction est elle-même composée de sous-fonctions calculant un score pour chaque critère :

$$scoreGlobal = (a * scoreSup + b * scoreCdM + c * scoreOrient) / (a + b + c) \quad (25)$$

Où :

- $scoreSup$, $scoreCdM$ et $scoreOrient$ sont les scores sur respectivement le critère de superficie, le critère du centre de masse et le critère sur l'orientation de la forme,
- a , b et c sont des valeurs de poids donnant la possibilité, au besoin, de donner davantage d'importance à un critère par rapport à un autre dans le calcul.

Soit x_{seg} le paramètre x calculé pour la segmentation du cas à résoudre et x_{ref} le paramètre x obtenu pour la segmentation du cas mémorisé. Les scores sont calculés en effectuant la différence $|x_{seg} - x_{ref}|$ et en les normalisant afin de supprimer la différence d'échelle de valeurs entre les 3 critères. La normalisation se fait en prenant la borne supérieure de leur ensemble de définition. De cette façon, pour des poids $a = b = c = 1$, les 3 scores ont une influence identique dans l'évaluation de la segmentation.

Le score sur la superficie est normalisé tel que :

$$scoreSup = 1 \text{ lorsque } |supRein_{seg} - supRein_{ref}| = w * h,$$

Où w est la largeur de l'image et h est sa hauteur, et ce score a pour expression :

$$scoreSup = 1 - \frac{|supRein_{seg} - supRein_{ref}|}{w * h} \quad (26)$$

Le score sur le centre de masse est normalisé tel que $scoreCdM = 1$ lorsque :

$$|CdM_{seg} - CdM_{ref}| = w * \sqrt{2},$$

Où la distance entre les 2 centres de masse équivaut à la diagonale de l'image dans le cas $w = h$, et ce score a pour expression :

$$scoreCdM = 1 - \frac{|CdM_{seg} - CdM_{ref}|}{w * \sqrt{2}} \quad (27)$$

Enfin, le score sur l'orientation est normalisé tel que $scoreOrient = 1$ lorsque :

$$|Orient_{seg} - Orient_{ref}| = \frac{\pi}{2},$$

Où $\frac{\pi}{2}$ est l'amplitude maximum dans l'ensemble de définition $[-\frac{\pi}{4}, \frac{\pi}{4}]$ de la fonction $\frac{1}{2}arctan$, et ce score a pour expression :

$$scoreOrient = 1 - \frac{2 * |Orient_{seg} - Orient_{ref}|}{\pi} \quad (28)$$

L'algorithme d'adaptation des seuils

Dans le but de déterminer une combinaison optimale de paramètres, nous définissons les intervalles de recherche IR_{local} et IR_{global} autour des seuils de référence $seuilLocal_{ref}$ et $seuilGlobal_{ref}$ tels que $IR_{local} = [seuilLocal_{ref} - \alpha, seuilLocal_{ref} + \alpha]$ et $IR_{global} = [seuilGlobal_{ref} - \alpha, seuilGlobal_{ref} + \alpha]$.

α est une valeur permettant de contrôler l'étendue de la recherche : plus celle-ci est élevée, plus le nombre de combinaisons Nb à évaluer est important. Ce nombre peut être déterminé par la formule suivante :

$$Nb = (2\alpha)^{nbre \text{ de germes} * 2} \quad (29)$$

Afin de limiter la complexité du problème, nous avons utilisé un maximum de 3 germes pour initier la segmentation. En plus de réduire le temps de calcul, ceci permet d'améliorer la réutilisation des solutions anciennes (germes à placer avec les seuils associés) pour résoudre un problème nouveau. Ainsi, à l'usage, se limiter à 3 germes nous a paru être un bon compromis entre précision de la segmentation et capacité de cette solution à être utilisée à nouveau pour un autre problème, non identique. D'après l'équation 29, pour une valeur $\alpha = 8$, il y a donc $16^6 = 16777216$ combinaisons possibles, ce qui représente un temps de calcul au-delà de l'acceptable. Nous avons donc privilégié une approche heuristique qui permet d'explorer de manière efficace une faible partie des possibilités sans obtenir *la solution optimale* mais une solution *acceptable* c'est à dire médicalement exploitable (nous utilisons les indices DICE et IU pour valider les segmentations obtenues par rapport aux segmentations manuelles effectuées par les médecins).

Algorithme 1 Adaptation des valeurs de seuils des germes

Entrée: *image, lgermes, infoSeg_{ref}, α , a, b, c*

Sortie: *lgermes*

```

seg ← segmentation(image, lgermes)
scoreSup ← calculScoreSup(seg, infoSegref.superficie)
scoreCdM ← calculScoreCdM(seg, infoSegref.centreDeMasse)
scoreOrient ← calculScoreOrient(seg, infoSegref.orientation)
scoreGlobal ← a * scoreSup + b * scoreCdM + c * scoreOrient
pour chaque germe g de lgermes faire
  seuilLocalref ← g.seuilLocal
  seuilGlobalref ← g.seuilGlobal
  pour i de  $-\alpha$  à  $\alpha$  faire
    pour j de  $-\alpha$  à  $\alpha$  faire
      seuilLocaltemp ← g.seuilLocal
      seuilGlobaltemp ← g.seuilGlobal
      g.seuilLocal ← seuilLocalref + i
      g.seuilGlobal ← seuilGlobalref + j
      seg ← segmentation(image, lgermes)
      scoreSup ← calculScoreSup(seg, infoSegref.superficie)
      scoreCdM ← calculScoreCdM(seg, infoSegref.centreDeMasse)
      scoreOrient ← calculScoreOrient(seg, infoSegref.orientation)
      scoreGlobalnew ← a * scoreSup + b * scoreCdM + c * scoreOrient
      si scoreGlobalnew > scoreGlobal alors
        scoreGlobal ← scoreGlobalnew
      sinon
        g.seuilLocal ← seuilLocaltemp
        g.seuilGlobal ← seuilGlobaltemp
      fin si
    fin pour
  fin pour
retourne lgermes

```

L'algorithme 1, d'adaptation des seuils, procède en plusieurs étapes. Tout d'abord, la segmentation *seg* est calculée une première fois avec la liste des germes *lgermes* directement issue du processus de remémoration. Ceci permet d'obtenir le score initial servant

de base pour la recherche d'une meilleure combinaison de seuils. Le calcul du score fait appel aux poids a , b et c tels qu'ils sont définis dans l'équation 25. Les valeurs de ces poids sont discutées dans le chapitre 6 portant sur les expérimentations. Ils sont à définir empiriquement pour maximiser la qualité des segmentations. Le fait de modifier ces poids impacte en effet les différents critères (superficie, centre de masse et orientation) dans la définition d'une segmentation optimale et modifie donc les résultats du système de RàPC.

Pour chacun des germes de $Igermes$, tous les couples possibles ($seuil_{local}$, $seuil_{global}$) sont explorées, en accord avec les intervalles de recherche suivant α , en fixant les autres valeurs puis la segmentation produite par chacune d'elles est évaluée. Si le score de la nouvelle segmentation est supérieur à celui de la précédente, les nouvelles valeurs de seuils sont conservées, sinon, les anciennes valeurs sont réaffectées. L'algorithme renvoie la liste des germes, avec des seuils adaptés au problème en cours qui serviront à produire la segmentation finale du parenchyme rénal.

Dans la mesure où une combinaison de deux seuils est recherchée en fixant les autres valeurs, et ce pour chaque germe de $Igermes$, nous pouvons recalculer le nombre de combinaisons explorées suivant la formule suivante :

$$Nb = 2 * nbre\ germes * (2\alpha)^2 \quad (30)$$

Le facteur 2 en début de formule correspond au fait que nous effectuons 2 passes d'adaptation pour plus d'efficacité. En limitant le nombre de germes à 3 et en fixant la valeur $\alpha = 8$, nous obtenons au plus d'après la formule 30 un nombre de possibilités de 1536. Ceci est très inférieur au nombre de combinaisons totales pour ce nombre de paramètres ($16^6 = 16777216$) et permet d'adapter les seuils dans un délai raisonnable.

5.1.5/ LES PHASES DE RÉVISION ET DE CAPITALISATION

Comme nous l'avons présenté au début de ce chapitre dans la Figure 5.1, toutes ces étapes sont effectuées manuellement par un expert dans le système actuel. L'expert évalue visuellement la qualité de la segmentation, en fonction de ses connaissances, et décide si celle-ci est suffisante. Dans le cas contraire, il réalise à nouveau une segmentation par croissance de régions, qu'il guide manuellement (détermination de la position des germes et des valeurs de seuils), permettant de compléter la base de cas et d'augmenter ainsi la capacité du système. La phase de capitalisation est actuellement systématique lorsque le système n'est pas parvenu à produire une segmentation de qualité suffisante. L'objectif est de lui permettre de traiter à l'avenir un problème similaire. Si la segmentation calculée est pertinente, nous considérons qu'il n'est pas intéressant d'alourdir une base de connaissance ayant déjà permis de parvenir au résultat souhaité.

Dans la section suivante, nous aborderons la deuxième partie de la contribution de ce manuscrit dédiée à une méthode d'entraînement particulière pour la segmentation du néphroblastome par *Deep Learning*. Nous y décrirons la formalisation mathématique de cette méthode.

5.2/ *OV²ASSION : SEGMENTATION DU NÉPHROBLASTOME PAR Deep Learning*

Dans la section précédente, nous avons exposé une méthodologie pour réaliser la segmentation du parenchyme rénal pathologique. Les caractéristiques de ce parenchyme, contraste et haute intensité de pixels, nous ont amenés à privilégier une approche par RàPC couplée à un algorithme de croissance de régions. Mais à l'inverse, le néphroblastome est quant à lui souvent peu contrasté dans les images, et il est d'une homogénéité relative, ce qui rend toute segmentation par croissance de régions très sensible aux fuites vers les structures environnantes (notamment vers les muscles avec lesquels il n'existe pas de contrastes suffisant). Le *Deep Learning* permet de tenir compte non seulement des intensités des pixels, mais également d'un large panel de critères (forme, localisation dans l'image, . . .) que le réseau est capable d'apprendre avec un entraînement suffisant.

Ces travaux de thèse ont pour but de répondre à des contraintes bien particulières inhérentes au domaine d'application. Nous nous appuyons sur des données provenant du service d'oncopédiatrie du Centre Hospitalier Régional Universitaire de Besançon. Le nombre de cas traités de néphroblastomes est faible et de plus peu de segmentations complètes ont été effectuées par les radiologues (rappelons qu'une segmentation complète nécessite de 7 à 9 heures de travail effectuées par un médecin). Ainsi, le nombre de vérités terrain à disposition pour entraîner notre réseau de neurones de type CNN est limité : c'est un des enjeux majeurs de cette phase de notre travail.

Le besoin de données en quantité suffisante et suffisamment qualitative représente une limite majeure de toutes les approches basées sur les Réseaux de Neurones Artificiels (RNA, et particulièrement les CNN) et le *Deep Learning*. L'efficacité d'un CNN dépend de son architecture et du contexte dans lequel il est utilisé. Cependant, une grande partie de cette efficacité est également conditionnée par l'étape d'apprentissage.

L'apprentissage sur des données insuffisantes, ou trop peu représentatives, peut mener à un phénomène de sur-apprentissage empêchant une généralisation efficace du modèle calculé. Dit autrement, un réseau de neurones en situation de sur-apprentissage produira des résultats excellents sur les données sur lesquelles il a appris, mais il sera incapable de produire de bons résultats sur toutes nouvelles données non incluses dans l'ensemble d'apprentissage, ce qui rend le réseau inutile. Toute la problématique de cette phase est donc de trouver un bon compromis entre capacité à suivre les données d'apprentissage et capacité à généraliser à de nouvelles données afin d'éviter ce phénomène.

Dans cette démarche, il est courant de séparer la base de données en deux ensembles :

- un ensemble d'apprentissage,
- et un ensemble de validation.

À chaque itération de l'entraînement, le réseau est évalué en parallèle sur les deux ensembles afin de s'assurer que l'erreur diminue pour chacun d'eux. Lorsque le nombre d'itérations devient trop important, les deux courbes divergent, marquant le basculement du réseau dans une situation de sur-apprentissage.

À l'inverse de la méthode classique, notre approche consiste à rechercher ce sur-apprentissage pour obtenir un modèle le plus efficace possible pour un patient donné. L'avantage est de rendre utilisable un CNN, même avec un faible ensemble. Ceci implique

néanmoins de disposer de données représentatives de ce patient pour lancer l'apprentissage.

5.2.1/ PRÉSENTATION GÉNÉRALE DE *OV²ASSION*

La nécessité d'entraîner un *Convolutionnal Neural Network* (voir chapitre 2 pour plus de détails) sur une large base de données est un verrou important dans un contexte médical dans lequel l'accès aux données est difficile. De plus, l'étiquetage des images est une tâche fastidieuse pour laquelle des connaissances médicales poussées sont nécessaires (tâche dans laquelle un médecin doit être impliqué), limitant d'autant plus la taille de cette base.

Comme nous l'avons précédemment indiqué, nous ne disposons que d'un nombre limité d'images segmentées à partir de différents patients et la singularité de chacune des tumeurs renforce le problème. Leur taille et leur forme imprévisibles compliquent la tâche de la segmentation. Toutefois, pour un patient donné, le néphroblastome possède une texture similaire entre chacune des coupes et les intensités de pixels d'une coupe à l'autre restent également similaires. Nous explorons donc la possibilité d'utiliser un réseau de neurones profond entraîné spécifiquement pour ce patient, capable d'effectuer les segmentations automatiques de la tumeur sans intervention humaine.

Dans la mesure où nous souhaitons réaliser la segmentation pour un cas complet (pour la totalité des coupes de l'examen dans lesquelles la tumeur est présente), plusieurs segmentations initiales de ce patient doivent constituer l'ensemble d'apprentissage. L'objectif est à la fois de minimiser le nombre de données dans cet ensemble et d'optimiser les résultats, de telle manière qu'un maximum de prédictions soient calculées à partir d'un minimum de segmentations initiales. Ces segmentations initiales peuvent être obtenues manuellement ou peuvent être les résultats suffisamment pertinents d'une méthode de segmentation classique. Cette approche consiste donc à réaliser, dans un premier temps, un sur-apprentissage sur un ensemble de segmentations préalablement réalisées pour, dans un second temps, obtenir une segmentation complète du patient en utilisant un CNN. Le but du sur-apprentissage est ici de parvenir à un réseau de neurones spécifique d'un patient, mais celui-ci doit rester néanmoins suffisamment mesuré (ne pas aller dans un excès de sur-apprentissage) afin de permettre un bon résultat sur les coupes non incluses dans les données d'apprentissage. Plus globalement, la méthode consiste donc à réaliser un apprentissage sur $X\%$ des coupes afin de pouvoir effectuer dans un second temps la segmentation des $Y\%$ coupes restantes. Nous avons nommé cette méthode *OV²ASSION*, pour *Overlearning Vector for Valid Sparse SegmentatIOns*.

Par expérience, pour la segmentation de la tumeur chez un patient donné, nous n'avons observé qu'une faible différence d'une coupe voisine à une autre. Comme le montre la Figure 5.7, la quantité de données peut être minimisée en prenant seulement en compte certaines segmentations manuelles à différents niveaux de la tumeur de telle sorte que ces segmentations soient représentatives de l'ensemble.

Pour fonctionner, il est nécessaire que les coupes segmentées incluses dans l'ensemble d'apprentissage soient suffisamment similaires aux coupes voisines que le CNN doit segmenter. Ainsi, un écart entre les coupes utilisées pour l'entraînement ne doit théoriquement pas être trop important. Le problème est de déterminer le nombre de coupes qu'il est possible d'ignorer entre deux coupes choisies, correspondant à l'écart, pour construire un ensemble d'apprentissage de taille minimale, tout en maintenant une préci-

sion correcte des segmentations calculées. Nous proposons d'optimiser la segmentation de la tumeur pour un patient donné en minimisant le nombre de données d'apprentissage, c'est à dire en maximisant l'écart entre les coupes sélectionnées.

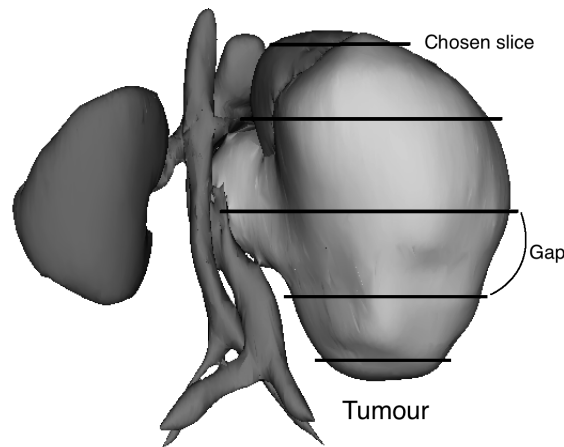


FIGURE 5.7 – Notions d'écart (*gap*) et de choix des coupes pour l'entraînement d'un CNN

5.2.2/ FORMALISATION

Pour explorer les différentes possibilités d'incorporation des coupes dans l'ensemble d'apprentissage LS (*Learning Set*), nous avons formalisé l'opération sous la forme d'un vecteur permettant de décider si une coupe doit ou non être ignorée lors de l'entraînement du réseau. Pour chaque élément de ce vecteur, 2 valeurs sont possibles : 1 si la coupe est utilisée pour l'entraînement, 0 dans le cas contraire. L'écart entre les coupes choisies a été fixé comme étant constant afin de s'assurer que LS soit représentatif de l'ensemble de la tumeur à segmenter : ceci permet d'exploiter l'information sur la tumeur de manière homogène, à différents niveaux, afin de parvenir à effectuer la segmentation de toutes les autres coupes du patient avec une précision suffisante. Par exemple, un écart de 2 signifie qu'une coupe est utilisée dans LS toutes les 3 coupes de l'examen, comme illustré par la suite dans le Tableau 5.1. Chaque écart/*gap* produit plusieurs vecteurs possibles selon la première coupe qui est prise en considération (avec un décalage d'un vecteur à un autre).

Pour faire ceci, nous utilisons les i vecteurs de V_g , avec un écart g , pour remplir l'ensemble d'apprentissage LS . V_g est une partie de V , l'ensemble de toutes les combinaisons de valeurs possibles tel que :

$$V_g \subset V = \{(0, 1)_{1...n}\} \quad (31)$$

Où n est le nombre de coupes à segmenter. Ce nombre varie généralement entre 40 et 60 selon d'une part les paramètres de l'examen, comme par exemple l'épaisseur de coupes, et d'autre part selon la taille de la tumeur chez le patient à traiter.

L'ensemble d'apprentissage LS du CNN contient toutes les coupes utilisées pour entraîner le réseau et est défini comme :

$$LS = \{S_j, \dots, S_k\} \quad (32)$$

Où les S_j, \dots, S_k représentent les coupes sélectionnées de la tumeur par le vecteur $(V_g)_i$ suivant les valeurs binaires de ses éléments.

Pour un écart donné, il existe plusieurs vecteurs possibles suivant le décalage que l'on choisit. La première coupe incluse dans LS peut en effet correspondre à la première coupe de l'examen (extrémité) ou à l'une des coupes suivantes jusqu'à couvrir l'écart g . V_g est donc en fait un ensemble de vecteurs. La notation suivante, $(V_g)_i$, est utilisée pour désigner le vecteur numéro i avec un écart (*gap*) g tel que :

$$V_g = \bigcup_{i=1}^h (V_g)_i \quad (33)$$

Où h est le nombre de vecteurs. Le premier vecteur $(V_g)_1$ correspond à celui prenant en compte la première coupe de l'examen (située à la première extrémité). Un exemple des 3 vecteurs possibles avec un écart $g = 2$ est présenté dans le Tableau 5.1. Tester tous les vecteurs pour un écart g donné permet une évaluation non seulement de l'écart optimal dans les résultats, mais également de l'impact de la position des coupes dans l'ensemble d'apprentissage LS .

	S_1	S_2	S_3	S_4	S_5	S_6	...	S_{60}	S_{61}
$(V_2)_1$	1	0	0	1	0	0	...	0	1
$(V_2)_2$	0	1	0	0	1	0	...	0	0
$(V_2)_3$	0	0	1	0	0	1	...	1	0

TABLE 5.1 – Exemple de V_2 pour $n = 61$

Suivant $(V_g)_i$, LS contient un ensemble restreint de coupes.

Nous pouvons construire LS en appliquant une simple fonction mathématique f qui sélectionne les coupes les plus pertinentes suivant les valeurs stockées dans $(V_g)_i$:

$$f : LS \times \{0, 1\} \mapsto LS \mid \forall i \in \{1, \dots, n\} \begin{cases} f(S_i, 0) = \\ f(S_i, 1) = S_i \end{cases} \quad (34)$$

La coupe S_i est incluse dans LS si la valeur à la position i du vecteur $(V_g)_i$ vaut 1 et est exclue sinon (valeur de 0).

Nous introduisons également une fonction F , utilisant f et définissant l'ensemble LS final suivant le vecteur $(V_g)_i$:

$$F : V \mapsto LS \mid F(V_g)_i = \bigcup_{j=1}^n \{f(S_j, ((V_g)_i)_{j-1})\} \quad (35)$$

5.2.3/ CARDINALITÉ

La cardinalité de l'ensemble des vecteurs est $Card_V = 2^n$ mais, dans la mesure où nous avons introduit la notion d'écart (*Gap*), toutes les combinaisons ne sont pas évaluées. La

cardinalité est relative à l'écart g choisi pour mener l'entraînement.

La cardinalité $Card_{V_g}$ pour un écart g a pour expression :

$$\forall g \in \{0, \dots, n - 2\} \quad Card_{V_g} = g + 1 \quad (36)$$

car avec un écart g entre 2 coupes incluses, l'ensemble des vecteurs possibles est g plus le premier vecteur.

La cardinalité pour tous les écarts évalués est $Card_{V_{test}}$:

$$Card_{V_{test}} = \sum_{g=0}^p (Card_{V_g}) \quad (37)$$

Où p est l'écart maximum. Dans le chapitre suivant nos résultats montrent que si ce Gap est trop élevé la précision des résultats décroît : nous nous sommes limité à un p de 10.

Synthèse du chapitre

Dans ce chapitre, nous avons présenté nos deux contributions majeures qui sont, d'une part, notre système de Raisonnement à Partir de Cas (RàPC) dédié à la segmentation d'images par une méthode de croissance de régions et, d'autre part, notre nouvelle méthode d'entraînement pour un CNN quelconque dédié à la segmentation du néphroblastome.

La principale contribution de ce manuscrit, dans le **domaine du RàPC**, est l'élaboration d'une phase d'adaptation pour la localisation des germes dans l'image à segmenter et pour les valeurs de seuils. Le nombre de germes reste lui inchangé dans le système actuel et a été limité à 3 maximum afin de limiter la complexité du problème et de faciliter la réutilisation d'une solution passée. L'adaptation des coordonnées des germes se fait sur un critère d'intensité de pixel. Lors de la remémoration, une intensité de référence est associée au germe. L'algorithme recherche en premier lieu une intensité identique dans une fenêtre de recherche et relâche la contrainte sur l'intensité progressivement jusqu'à trouver un bon pixel candidat pour accueillir le germe. L'adaptation des seuils se fait par l'intermédiaire d'une fonction score permettant d'évaluer la pertinence de la segmentation calculée pour un ensemble de seuils par rapport à un résultat attendu (segmentation similaire à celle produite pour le cas remémoré). Afin de limiter le coût en temps de calcul, une approche heuristique a été choisie pour n'explorer qu'une partie limitée des solutions possibles.

Dans le domaine du **Deep Learning**, comme notre contexte particulier impose un nombre très limité de données, l'objectif de notre méthode est de rechercher un sur-apprentissage spécifique à un patient. Ce sur-apprentissage se fait à partir d'un échantillon de coupes de la tumeur, préalablement segmentées et faisant office de vérités terrain (soit manuellement, soit par une technique de segmentation autre donnant des résultats suffisamment fiables), afin de parvenir à segmenter le reste des images.

Dit autrement, le principe de la méthode est de réaliser un apprentissage sur $X\%$ des coupes vérités terrain pour effectuer dans un second temps la segmentation des $Y\%$ restantes. Nous avons nommé cette méthode *OV²ASSION*, pour *Overlearning Vector for Valid Sparse SegmentatIOns*. La construction de l'ensemble d'apprentissage *LS* se fait par l'utilisation d'un vecteur contenant des poids binaires associés à chacune des coupes du patient. Ces poids permettent de sélectionner régulièrement (sur toute la hauteur de la tumeur) les coupes à inclure dans les données d'entraînement en accord avec une valeur d'écart g .

Le chapitre suivant est consacré à la présentation des résultats concernant la segmentation du parenchyme rénal par le système de RàPC et la segmentation de la tumeur selon la méthode *OV²ASSION*. Ces résultats seront également discutés afin de mettre en évidence leur intérêt ainsi que leurs limites.

RÉSULTATS ET DISCUSSIONS

Dans le chapitre précédent, nous avons détaillé la méthodologie et les concepts théoriques de nos contributions. D'une part, nous avons présenté le fonctionnement de notre système de RàPC, associé à un algorithme de croissance de régions, et plus particulièrement sa phase d'adaptation de la solution remémorée (paramètres de la segmentation). D'autre part, une méthode d'entraînement pour les CNN, intitulée *OV²ASSION*, a été décrite.

Dans le présent chapitre, les résultats des expérimentations concernant ces contributions sont présentés et discutés. Tout d'abord, nous introduirons de manière détaillée les métriques que nous avons utilisées pour effectuer l'évaluation des segmentations calculées par nos systèmes (RàPC et *Deep Learning*), ainsi que les méthodes pour déterminer les scores. Puis, nous exposerons dans une deuxième partie les premières expérimentations ayant permis d'orienter certains choix dans les contributions. Nous y justifierons l'architecture utilisée pour le *Deep Learning*, l'attribution des tâches de segmentations aux différents systèmes et la métrique de similarité exploitée pour la phase de remémoration du RàPC. Les 2 parties suivantes sont respectivement dédiées aux résultats des expérimentations pour la segmentation du parenchyme rénal par le système de RàPC, avec sa phase d'adaptation, ainsi qu'aux résultats de la méthode *OV²ASSION* pour la segmentation du néphroblastome. Les 2 étapes clés de la phase d'adaptation du RàPC ont été évaluées, à savoir la modification des coordonnées des germes et celle des valeurs de seuils, afin de mettre en évidence leur intérêt. Pour l'approche par *Deep Learning*, nous nous sommes intéressés dans un premier temps à l'influence d'une part de l'écart g sur les résultats et d'autre part à celle du choix du vecteur pour un écart donné (correspondant à la position de la première coupe incorporée dans l'ensemble d'apprentissage *LS*). Une discussion sur l'ensemble des résultats permet de clore ce chapitre.

6.1/ LES MÉTRIQUES POUR L'ÉVALUATION DES SEGMENTATIONS

Comme défini dans le chapitre 1, nous utilisons dans ce manuscrit 2 métriques différentes (et complémentaires) afin d'évaluer et de comparer les résultats. Ces 2 métriques sont les indices de Dice, aussi connu sous le terme d'indice de Sørensen-Dice ou simplement coefficient Dice, et l'indice de Jaccard, plus utilisé sous le nom IoU (*Intersection over Union*) en informatique. L'indice Dice est communément utilisé dans le milieu de l'imagerie médicale pour évaluer la pertinence d'une segmentation par rapport à une vérité terrain. L'indice IoU est quant à lui plutôt utilisé par la communauté en Traitement

d'Images et en Intelligence Artificielle. Le choix de ces 2 métriques permettront ainsi aux experts de l'imagerie et de la médecine de mieux évaluer les résultats obtenus.

L'une comme l'autre donnent une valeur de similarité (allant de 0, pour une similitude inexistante à 1, pour une similitude parfaite) entre deux ensembles X et Y . Dans notre cas, X représente les pixels de la segmentation calculée, fournie par notre système, et Y représente les pixels appartenant à la vérité terrain (c'est à dire la segmentation désirée). Ces 2 indices peuvent être vus, chacun à leur manière, comme un taux de recouvrement entre les deux segmentations comparées.

Le coefficient Dice s'exprime comme un rapport entre le nombre de pixels ayant un label identique dans X et Y et la somme des pixels dans X et Y . Dans la mesure où Y est la vérité terrain de X , le nombre de pixels dans les deux images est identique. Ceci aboutit à la formule suivante permettant le calcul du score :

$$Dice = \frac{2 * |X \cap Y|}{|X| + |Y|} = \frac{n_{bonLabel}}{n_{pixels}} \quad (38)$$

Où :

- $n_{bonLabel}$ est le nombre de pixels correctement étiquetés dans X (en prenant pour référence Y),
- n_{pixels} est le nombre de pixels dans la segmentation X/Y .

L'indice IoU se calcule d'une manière un peu différente en effectuant, comme son nom l'indique, le rapport entre l'intersection $X \cap Y$ et l'union $X \cup Y$:

$$IoU = \frac{|X \cap Y|}{|X \cup Y|} = \frac{n_{bonLabel}}{2n_{pixels} - n_{bonLabel}} \quad (39)$$

Un indice Dice/IoU peut être également calculé pour chaque label dans l'image. Ceci permet, par souci de pertinence, d'ignorer le score sur le label de fond (*background*, ayant généralement pour valeur 0) car il augmente artificiellement la qualité de la segmentation calculée. Le nombre de pixels appartenant au *background* a en effet tendance à être très élevé par rapport à ceux décrivant la/les structures d'intérêt, induisant de fait une proportion importante de pixels correctement étiquetés qui faussent ainsi le résultat.

Soit :

- n_{ii} le nombre de pixels correctement étiquetés i ,
- n_{ji} le nombre de pixels faussement étiquetés un des autres labels au lieu de i ,
- n_{ij} le nombre de pixels faussement étiquetés i au lieu d'un autre label.

Dans ce cas, le score de Dice pour le label i peut s'exprimer de la manière suivante :

$$Dice_i = \frac{2 * |X_i \cap Y_i|}{|X_i| + |Y_i|} = \frac{2n_{ii}}{2n_{ii} + n_{ji} + n_{ij}} \quad (40)$$

De la même manière, nous pouvons réécrire la formule pour l'indice IoU comme suit :

$$IU_i = \frac{|X_i \cap Y_i|}{|X_i \cup Y_i|} = \frac{n_{ii}}{n_{ii} + n_{ji} + n_{ij}} \quad (41)$$

Cette nouvelle expression des indices Dice et IoU permet de mettre en évidence une formulation très proche. Finalement, il ne s'agit que d'une différence de pondération du terme n_{ii} qui devient prépondérant avec le Dice quand l'IoU ne le privilégie pas. En conséquence, le score de Dice prend surtout en compte le nombre de pixels correctement classés dans son calcul au détriment des pixels ayant le mauvais label. Le score IoU est donc plus punitif vis à vis des phénomènes de sur-segmentation ou de sous-segmentation, menant généralement à un score inférieur à celui du Dice.

Enfin, le calcul d'un score peut évidemment se faire autant sur une image (2D) que sur un ensemble de coupes (3D). Dans ce dernier cas, pour évaluer un score de segmentation pour un patient entier, il existe plusieurs approches. La première repose sur la détermination de l'indice en question pour chacune des coupes du patient afin d'en extraire une moyenne (Dice ou IoU moyen). La seconde consiste à calculer le score en une seule fois sur toutes les coupes en considérant que celles-ci forment une image 3D cohérente et non pas des images 2D indépendantes les unes des autres (Dice ou IoU global). Dans cette situation, X désigne la segmentation 3D calculée par le système et Y la segmentation 3D vérité terrain, et les éléments de comparaison (labels attribués) entre X et Y peuvent être vus comme les voxels de ces images 3D. Le choix de la méthode a un fort impact sur le résultat des calculs. La méthode globale revient à introduire une pondération entre les coupes selon l'importance de la structure d'intérêt s'y trouvant (nombre de voxels étiquetés comme appartenant à cette structure) alors que la méthode de la moyenne donne un poids équivalent à toutes les coupes. Les deux méthodes sont valides mais le choix de la méthodologie est important pour que les comparaisons des résultats entre eux soient pertinentes.

6.2/ EXPÉRIMENTATIONS LIMINAIRES

Des expérimentations liminaires ont été effectuées afin d'orienter les premiers choix faits dans l'architecture de la plate-forme COLISEUM-3D. Ces justifications portent tout d'abord sur le choix du réseau de neurones convolutif (CNN) sur lequel appliquer la méthode *OV²ASSION*, et ensuite sur le choix d'une approche RàPC pour segmenter le rein et d'une approche *Deep learning* pour segmenter la tumeur.

6.2.1/ LE CHOIX DU RÉSEAU DE NEURONES CONVOLUTIF

OV²ASSION, présentée dans le chapitre 5.2, est une méthode d'entraînement applicable à n'importe quel réseau de neurones convolutif. Les gains d'*OV²ASSION* peuvent donc être mesurés en l'appliquant à n'importe quel réseau. Toutefois, ce travail de thèse s'inscrit dans un projet qui demande d'obtenir des résultats suffisamment fiables et exploitables par un médecin, il est donc important d'utiliser dès le départ pour nos expérimentations le réseau dont les performances sont les plus prometteuses. Pour ce faire, nous nous sommes appuyés sur les résultats présentés dans la littérature puis nous avons testé préalablement les réseaux les plus performants à la segmentation de certaines

	FCN-8s	U-Net
Dice moyen test 1	0.91	0.65
Dice moyen test 2	0.90	0.89
Dice moyen test 3	0.89	0.83
Dice moyen (tous les tests)	0.90	0.79
Ecart type (tous les tests)	1.0×10^{-2}	1.25×10^{-1}

TABLE 6.1 – Performance et stabilité des réseaux FCN-8s et U-Net pour la segmentation du néphroblastome

images issues de notre base de tests (scanners d'un patient fourni par le Centre Hospitalier Régional Universitaire de Besançon et atteint d'un néphroblastome). Ainsi dans notre contexte, nous avons limité notre prospection à deux réseaux de neurones classiquement utilisés dans le domaine pour leurs performances respectives : FCN-8s et U-Net. Il existe néanmoins, comme nous l'avons exposé dans le chapitre 2, de nombreuses autres architectures.

La Tableau 6.1 montre les résultats préliminaires concernant la segmentation de la tumeur, obtenus pour ces deux CNN. FCN-8s a été entraîné durant 10000 itérations et U-Net durant 13500 itérations sur des données d'entraînement identiques. Les données d'apprentissage sont composées de 40 images scanner (*CT-scans*) provenant d'un patient, régulièrement espacées pour être représentatifs de l'ensemble de la tumeur. L'ensemble de test est composé des 20 autres images, provenant du même patient, non inclus dans l'ensemble d'entraînement. 3 tests ont été réalisés afin d'évaluer leurs performances et leur stabilité. Pour chacun des tests, les indices de Dice et IoU moyens ont été calculés à partir des segmentations manuelles réalisées par des chirurgiens et utilisées en tant que vérités terrain. La dernière ligne du tableau présente l'écart type sur les résultats permettant d'évaluer la stabilité du réseau sur nos données. Le tableau 6.1 montre que le réseau U-Net semble moins efficace sur nos données (Dice moyen égal à 0.79) que le réseau FCN-8s (Dice moyen égal à 0.90). De plus, nos tests préliminaires montrent une plus grande instabilité de U-Net. Pour toutes ces raisons, nous avons choisi de conduire nos expérimentations avec le réseau FCN-8s plutôt qu'avec le réseau U-Net.

6.2.2/ LE CHOIX DE L'APPROCHE À PRIVILÉGIER EN FONCTION DE LA STRUCTURE À SEGMENTER

Nous avons ensuite évalué les résultats provenant du système de RàPC et du CNN afin de déterminer quelle approche semble la meilleure pour segmenter les 2 structures suivantes : le parenchyme rénal et le néphroblastome. Le système de RàPC a été évalué avec une base de cas constituée de 10 images provenant d'un patient. Pour cette expérimentation, nous avons utilisé le processus d'adaptation pour la localisation des germes (présenté dans la section 5.1.4.1) mais sans l'adaptation des valeurs de seuils. Le réseau FCN-8s a été entraîné sur un ensemble d'apprentissage de 12 images et testé sur 6 images, choisies aléatoirement dans les 2 cas tels que les images de test ne soient pas incluses dans l'ensemble d'entraînement, provenant du même patient que pour le système de RàPC. Dans la littérature, les CNN donnent de bonnes performances pour la segmentation de larges structures, mais semblent moins efficaces en ce qui concerne les structures plus petites et plus complexes. Nous avons donc émis l'hypothèse que les

	RàPC-CR adapt pos		FCN-8s	
	Dice moyen	IoU moyen	Dice moyen	IoU moyen
Tumeur	0.92	0.86	0.94	0.89
Rein	0.83	0.71	0.36	0.23

TABLE 6.2 – Dice et IoU pour la segmentation du parenchyme rénal et du néphroblastome par RàPC avec croissance de régions (avec adaptation de la position des germes) et CNN

CNN seraient meilleurs pour segmenter la tumeur, qui présente une structure large et régulière sur les images 2D, tandis que le parenchyme rénal, qui a des valeurs de pixels plus remarquables, pourrait être plus facile à segmenter par un algorithme de croissance de région. Ces hypothèses ont été vérifiées (cf section 6.3.2.3)

Le Tableau 6.2 montre ces résultats liminaires avec le calcul du coefficient Dice et IoU. FCN-8s a le meilleur score pour la segmentation de la tumeur, avec un Dice moyen de 0.94, alors que la croissance de régions guidée par RàPC (RàPC-CR) est meilleure pour la segmentation du rein. Ceci confirme notre hypothèse de départ. Dans notre application, le rein est une structure plus petite qu'une tumeur et peut subir une importante déformation due à la présence du néphroblastome. De plus, les CNN sont connus pour être une approche très performante mais ils doivent en contrepartie apprendre sur une large base de données pour exprimer leur potentiel. Ceci peut en partie expliquer les faibles valeurs pour la segmentation du rein obtenues avec FCN-8s, qui doit effectuer des segmentations dans une situation lui étant défavorable (seulement 12 images dans les données d'apprentissage). Mais ceci reste en accord avec le contexte de ce travail dans lequel la quantité de données est très limitée. Concernant la tumeur, les 2 approches ont une performance similaire avec un Dice de 0.92 pour le RàPC-CR et de 0.94 pour FCN, avec un léger avantage pour le CNN. Un autre intérêt d'un réseau de neurones est que son entraînement est simple à mettre en œuvre (à condition de disposer des données étiquetées) comparativement à la construction très chronophage d'une base de cas. En plus de cette considération pratique, un algorithme de croissance de régions est sensible au phénomène de fuite dès lors que les contrastes entre l'objet à segmenter et les structures environnantes sont faibles : ce qui est régulièrement le cas avec les tumeurs (notamment vis-à-vis des tissus musculaires). De ces résultats, nous pouvons en conclure que le CNN apparaît comme étant une solution plus intéressante pour segmenter le néphroblastome, tandis qu'un algorithme de croissance de régions, guidé par RàPC, est clairement plus indiqué pour la segmentation du parenchyme rénal.

6.2.3/ ÉVALUATION DES MÉTRIQUES DE SIMILARITÉS

Dans la section 5.1.3, nous avons présenté 2 calculs de similarité issus de la littérature, nommés s_1 et s_2 , pour effectuer la remémoration du cas stocké le plus proche. s_1 se base sur les travaux de P. Perner [perner1999segCBR] pour calculer une valeur de similarité entre le cas à résoudre et chacun des cas stockés. s_2 reprend la formule de P. Perner pour la similarité sur les méta-données mais modifie la partie sur les informations statistiques de l'image en utilisant le critère MSSIM (*Mean Structural SIMilarity*) des travaux de Wang *et al.* [wang2004similarity].

Pour effectuer les tests et observer le comportement de ces deux métriques s_1 et s_2 , nous avons construit une base de cas spécifique. Cette base est constituée de 10 cas provenant d'un même et unique patient. Chaque cas représente une coupe différente de ce patient. La base est organisée de telle sorte que chaque cas forme un couple avec un autre cas suffisamment proche pour être susceptible d'être remémoré. Ces couples sont simplement formés de deux coupes relativement voisines du patient. Les couples sont les suivants : (1, 6), (2, 7), (3, 8), (4, 9) et (5, 10).

Pour permettre la remémoration du cas stocké le plus pertinent, plusieurs conditions peuvent être imaginées :

- La première est d'être capable de rendre compte de la différence entre 2 cas donnés. Plus cette différence est importante, moins la valeur de similarité doit être élevée. De cette façon, le système peut sélectionner un cas qui est perçu, suivant les critères pris en compte pour le calcul de la similarité, comme étant le plus proche.
- La deuxième condition, liée à la première, est de donner par construction une similarité de 1 (ou le plus proche possible de cette valeur) entre 2 cas identiques.
- Enfin, dans une base suffisamment grande, une similarité avec un haut pouvoir discriminant peut être intéressante. Cependant, nous nous plaçons ici dans la situation d'une base contenant un nombre limité de cas.

Le Tableau 6.3 montre les différentes valeurs de similarité, suivant ces 2 formules s_1 et s_2 , entre un cas donné de la base et chaque cas la composant (y compris avec lui-même). La dernière colonne montre le cas remémoré pour chaque cas cible et chaque mesure de similarité. Le cas remémoré est celui ayant obtenu la plus haute valeur de similarité avec le cas cible, excluant le cas cible lui-même (nous nous plaçons volontairement dans une situation dans laquelle la base ne contient pas le cas cible à l'identique). Les résultats sont équivalents pour les deux similarités (s_1 et s_2) et, pour chaque cas, le cas remémoré est le même. Ces résultats montrent que quelle que soit la similarité utilisée, le cas remémoré x est celui appartenant au couple (*cas cible*, x). Il y a seulement 2 différences. La formule de P. Perner apparaît comme étant plus discriminative, avec une plus haute différence entre les similarités d'une image à une autre. MSSIM semble être une similarité plus efficace dans la mesure où 2 cas identiques donnent une similarité de 1. En conséquence, nous avons fait le choix d'utiliser le critère MSSIM (correspondant à s_2).

6.3/ ÉVALUATION DU SYSTÈME DE RÀPC ET PLUS PARTICULIÈREMENT DE SA PHASE D'ADAPTATION

Cette section du chapitre est dédiée à l'évaluation du système de RàPC et plus particulièrement de son processus d'adaptation. Ce processus est divisé en deux étapes qui seront évaluées de manière séparée. La première étape est l'adaptation de la position des germes afin de s'assurer que ceux-ci soient placés à des coordonnées optimales. Cette adaptation de la position se fait en suivant la méthodologie décrite dans la section 5.1.4.1. La seconde étape concerne l'adaptation des seuils des germes suivant la méthode présentée dans la section 5.1.4.2. Pour chacune de ces étapes, les conditions expérimentales sont tout d'abord décrites avant de présenter les résultats obtenus (valeur de scores : Dice/IoU).

		Base de cas										
		1	2	3	4	5	6	7	8	9	10	cas remémoré
Cas 1	s_1	0.999	0.788	0.537	0.299	0.155	0.908	0.733	0.458	0.227	0.100	6
	s_2	1.000	0.950	0.903	0.876	0.862	0.995	0.935	0.894	0.870	0.859	6
Cas 2	s_1	0.788	0.999	0.749	0.511	0.368	0.696	0.945	0.670	0.440	0.312	7
	s_2	0.950	1.000	0.966	0.935	0.915	0.936	0.995	0.956	0.926	0.910	7
Cas 3	s_1	0.537	0.749	0.999	0.761	0.618	0.446	0.805	0.920	0.690	0.562	8
	s_2	0.903	0.966	1.000	0.980	0.960	0.889	0.981	0.997	0.970	0.953	8
Cas 4	s_1	0.299	0.512	0.761	0.999	0.857	0.207	0.566	0.841	0.929	0.801	9
	s_2	0.876	0.935	0.980	1.000	0.990	0.862	0.951	0.988	0.997	0.982	9
Cas 5	s_1	0.156	0.368	0.618	0.857	0.999	0.064	0.423	0.698	0.928	0.944	10
	s_2	0.862	0.915	0.960	0.989	1.000	0.849	0.932	0.970	0.995	0.996	10
Cas 6	s_1	0.908	0.696	0.446	0.207	0.064	0.999	0.641	0.366	0.136	0.008	1
	s_2	0.995	0.936	0.889	0.862	0.850	1.000	0.920	0.879	0.856	0.846	1
Cas 7	s_1	0.733	0.945	0.805	0.566	0.423	0.641	0.999	0.725	0.495	0.367	2
	s_2	0.935	0.995	0.980	0.951	0.932	0.920	1.000	0.971	0.942	0.925	2
Cas 8	s_1	0.458	0.670	0.920	0.841	0.698	0.366	0.725	0.999	0.770	0.642	3
	s_2	0.894	0.956	0.997	0.988	0.970	0.879	0.971	1.000	0.978	0.962	3
Cas 9	s_1	0.227	0.440	0.690	0.929	0.928	0.136	0.495	0.770	0.999	0.872	4
	s_2	0.870	0.926	0.970	0.997	0.996	0.856	0.942	0.978	1.000	0.988	4
Cas 10	s_1	0.100	0.312	0.562	0.801	0.944	0.008	0.367	0.642	0.872	0.999	5
	s_2	0.859	0.910	0.953	0.982	0.996	0.846	0.925	0.962	0.988	1.000	5

TABLE 6.3 – Comparaison des deux métriques de similarité (s_1 et s_2) pour chaque cas durant une validation croisée

6.3.1/ L'ADAPTATION DE LA POSITION DES GERMES

Nous nous intéressons tout d'abord à l'évaluation de la première étape de l'adaptation, à savoir celle des coordonnées des germes.

6.3.1.1/ LE CONTEXTE DE L'ÉVALUATION

Nous avons évalué l'adaptation des coordonnées des germes du système RàPC avec un ensemble de 10 images scanners, numérotées de 1 à 10, provenant d'un seul patient et nous avons utilisé une méthode de validation croisée. La base utilisée est identique à celle ayant servi à l'évaluation des métriques de similarité (section 6.2.3). Elle est donc également organisée en couples de cas suffisamment proches entre eux. La validation croisée consiste à successivement et temporairement supprimer un cas de la base pour le résoudre à l'aide des 9 autres restants. Bien entendu, nous pourrions segmenter une image déjà existante dans la base de cas mais ceci consisterait seulement à reprendre la solution manuelle (avec ses paramètres) et à l'appliquer dans les mêmes conditions. Dans cette situation, une phase d'adaptation deviendrait inutile et donc impossible à évaluer. Une méthodologie pertinente est donc de trouver une bonne solution pour un problème qui n'a pas déjà été traité par le système (ou qui n'a pas été incorporé manuellement dans sa base de cas). Nous avons calculé les scores pour chacun d'entre eux (Dice et IoU) ainsi que les moyennes et médianes correspondantes.

Nous souhaitons évaluer ici la capacité du système à modifier une solution existante pour trouver celle qui serait optimale et non pas la précision absolue des segmentations. La solution optimale signifie le meilleur ensemble de paramètres pour la croissance de régions déterminé par l'expert. Dans cette section, nous avons donc choisi de ne pas comparer les résultats du système aux vérités terrain (segmentations manuelles des médecins) mais plutôt d'effectuer cette comparaison avec les meilleures segmentations obtenues par un expert grâce à une croissance de régions. Ceci permet de s'affranchir des éventuelles limites de cette méthode pour se concentrer sur la capacité du système à adapter au mieux une solution remémorée au problème cible à résoudre.

6.3.1.2/ LES RÉSULTATS DE L'ADAPTATION DES COORDONNÉES

Tout d'abord, afin d'illustrer les effets de cet algorithme d'adaptation, la Figure 6.1 montre une comparaison entre une exécution du système de RàPC avec et sans cette étape pour la segmentation du cas 2. Le parenchyme rénal est représenté dans cette figure par un label gris clair et la tumeur par un label gris foncé. Le label blanc correspond à la colonne vertébrale du patient. Dans cet exemple, nous nous intéressons à la segmentation du rein dans la mesure où nous dédions le système de RàPC à cette tâche. De la gauche vers la droite, nous avons l'image à segmenter (CT-scan), puis une segmentation par croissance de régions effectuée sans l'algorithme d'adaptation et enfin cette même segmentation (réalisée avec les mêmes paramètres) avec l'algorithme d'adaptation. Comme le montre la Figure 6.1, le parenchyme rénal est ici en deux parties à cause de la présence du néphroblastome. Sans le processus d'adaptation, le système est parvenu à segmenter correctement la partie inférieure, mais a échoué à segmenter la partie supérieure. Ceci vient du fait que le germe correspondant a été placé, lors de la remémoration, en dehors du rein. Notre étape d'adaptation a pu corriger cette erreur de positionnement du germe

et elle a ainsi pu significativement améliorer la justesse de la segmentation.

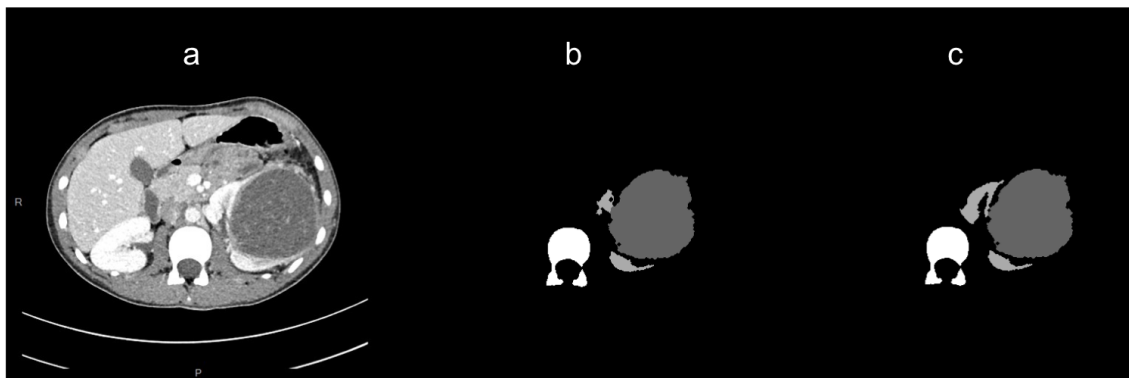


FIGURE 6.1 – Comparaison d’une segmentation du rein, en gris intermédiaire, par le système de RàPC avec et sans adaptation de la position des germes : (a) CT-scan (b) segmentation sans adaptation et (c) segmentation avec adaptation de la position

Sur les 10 cas de cette base de tests, les germes sont seulement placés à la bonne position dans 60% des situations si nous n’effectuons pas d’adaptation après la remémoration de la solution. Nous définissons comme une bonne position celle permettant de placer un germe dans le parenchyme rénal. Grâce à l’introduction de l’adaptation de cette position, le placement des germes a été amélioré avec un taux de bonnes positions de 100%.

Le Tableau 6.4 montre les indices Dice et IoU obtenus pour la segmentation du rein. Conformément à notre méthodologie de validation croisée, le numéro du cas indiqué dans la première colonne correspond à l’image testée en tant que cas cible, avec les 9 autres cas en tant que cas sources servant à sa résolution. Les cas mis en évidence en gris dans le tableau sont les cas présentant une amélioration de la segmentation grâce à l’adaptation des coordonnées des germes. Comme indiqué dans la section 6.3.1.1, les scores de Dice et IoU ont été calculés ici par rapport à la meilleure segmentation obtenue par l’expert grâce à la croissance de région et non pas par rapport aux vérités terrain des chirurgiens. Dans la plupart des cas, le système de RàPC parvient à se remémorer les bonnes coordonnées pour placer correctement les germes dans l’image (à l’intérieur du parenchyme rénal) même sans phase d’adaptation. Pour ces cas, l’adaptation ne permet pas d’améliorer les résultats dans la mesure où le RàPC n’en a pas besoin. Toutefois, plusieurs segmentations (2,3,6 et 9) ne donnent pas initialement un résultat optimal. Ceci est particulièrement vrai pour le cas 6 avec une valeur de Dice et IoU de 0. Ceci indique que les germes ont été particulièrement mal placés, entraînant un échec total de la segmentation. Après l’implémentation de l’adaptation, les résultats pour ces cas se sont très clairement améliorés. Le cas 2 passe d’un Dice de 0.559 à un Dice de 0.999 et le cas 6 voit une amélioration encore plus importante d’un Dice de 0 à un Dice de 0.997. De manière générale, sur l’ensemble des 10 cas testés, la moyenne des scores Dice et IoU passe respectivement de 0.815 à 0.978 et 0.765 à 0.959.

La position des germes n’est pas le seul paramètre nécessaire à une croissance de régions pour obtenir un bon résultat, même s’il s’agit ici du seul paramètre subissant une adaptation au cas cible. Il est pertinent de s’intéresser également aux autres paramètres et en particulier à l’influence des valeurs des seuils sur le résultat. Chaque germe est associé à deux seuils différents (seuil global et seuil local) conditionnant la manière dont chaque région va croître.

	RàPC-CR sans adapt pos		RàPC-CR avec adapt pos	
	Dice	IoU	Dice	IoU
Case 1	0.998	0.995	0.998	0.995
Case 2	0.559	0.388	0.999	0.999
Case 3	0.843	0.728	0.993	0.987
Case 4	0.894	0.809	0.894	0.809
Case 5	0.997	0.995	0.997	0.995
Case 6	0.000	0.000	0.998	0.997
Case 7	0.927	0.865	0.927	0.865
Case 8	1.000	1.000	1.000	1.000
Case 9	0.936	0.879	0.973	0.948
Case 10	0.998	0.996	0.998	0.996
Moyenne	0.815	0.765	0.978	0.959
Médiane	0.931	0.872	0.997	0.995

TABLE 6.4 – Scores de Dice et IoU pour la segmentation du parenchyme rénale avec une croissance de régions guidé par RàPC selon l'existence d'une phase d'adaptation de la position des germes

	Cas remémoré	Erreur globale	Erreur locale
Cas 1	6	Pas d'erreur	Pas d'erreur
Cas 2	7	Pas d'erreur	Pas d'erreur
Cas 3	8	Pas d'erreur	Pas d'erreur
Cas 4	9	Erreur (20.9 %)	Erreur (17.5 %)
Cas 5	10	Erreur (3.6 %)	Pas d'erreur
Cas 6	1	Pas d'erreur	Pas d'erreur
Cas 7	2	Pas d'erreur	Pas d'erreur
Cas 8	3	Pas d'erreur	Pas d'erreur
Cas 9	4	Erreur (29.7 %)	Erreur (14.5 %)
Cas 10	5	Pas d'erreur	Pas d'erreur

TABLE 6.5 – Erreurs moyennes pour les seuils remémorés : globale et locale

Le Tableau 6.5 présente pour chaque cas segmenté les erreurs relatives sur les seuils remémorés durant la phase de remémoration. Les erreurs ont été calculées par rapport aux seuils optimaux déterminés durant la segmentation par croissance de régions guidée manuellement par un expert et sont une moyenne des erreurs pour tous les germes placés dans le rein (chaque germe a ses propres seuils et donc ses propres erreurs relatives sur ces seuils). La plupart des germes n'ont pas d'erreur sur leurs seuils après la remémoration, ce qui indique que les cas remémorés étaient suffisamment proches des problèmes cibles (similarités très importantes) pour que les seuils soient identiques. Toutefois, pour quelques cas, il existe une erreur importante (cas 4 et 9), allant de 14.5 à 29.7%. Le cas 4 correspond d'ailleurs au cas de la base donnant le moins bon résultat dans le Tableau 6.4 avec un score Dice de 0.894, très inférieur à la moyenne qui est de 0.978. Ceci peut expliquer ce moins bon résultat, mais semble contredit par les cas 7 et 9. Le cas 7 présente une erreur nulle sur ses seuils alors qu'il correspond à l'une des moins

bonnes segmentations dans le Tableau 6.4 (score Dice de 0.927). Au contraire, le cas 9 présente une erreur importante (jusqu'à 29.7%) tout en donnant un score supérieur, très proche de la moyenne (score Dice de 0.973). Ces résultats, *a priori* contradictoires, peuvent s'expliquer par le fait que les seuils sont fortement liés à la position des germes. Un même seuil dans la même image peut donner un résultat différent si la position n'est pas strictement la même. Comme la position mémorisée, même après adaptation, n'est pas identique à celle déterminée lors de la croissance de régions guidée manuellement, il est logique de constater cette différence.

Ceci montre donc la principale limitation de ce système de RàPC pour la segmentation d'images. Les valeurs de seuils étant fortement liées aux coordonnées des germes, il est tout autant nécessaire de les adapter pour optimiser les résultats d'une segmentation par croissance de régions.

6.3.1.3/ COMPARAISON AVEC D'AUTRES APPROCHES POUR LA SEGMENTATION D'IMAGES

Le système de RàPC est comparé avec 3 autres approches de segmentation d'images : une croissance de régions et une méthode par ligne de niveaux (plus connu sous son terme anglophone *level set*), toutes deux guidées par un expert (détermination empirique des paramètres), et un CNN. La segmentation par *level set* a été effectuée sous le logiciel de traitement d'image Image J. Pour le CNN, nous avons utilisé une implémentation de l'architecture FCN-8s provenant des travaux de Long *et al.* [long2015fcn]. Il a été entraîné sur un ensemble d'apprentissage de 10 images (CT-scans) correspondant à la base de cas du système et suivant une validation croisée (entraînement sur 9 images et test sur la dixième) tout comme pour le système RàPC afin d'effectuer la comparaison dans les mêmes conditions. Le Tableau 6.6 montre les résultats de cette comparaison entre les différentes méthodes. La version du système de RàPC évalué est celle avec notre processus d'adaptation pour les coordonnées des germes utilisées précédemment. Les évaluations ont été faites en calculant les scores Dice et IoU, tout comme pour les expérimentations précédentes. Cependant, nous avons cette fois considéré les segmentations manuelles des chirurgiens pédiatriques en tant que vérités terrains pour calculer ces scores. Ils représentent donc la précision de référence des segmentations. Les cellules en couleur grise montrent les performances les plus mauvaises (nous considérons ici une performance comme mauvaise si son score de Dice est inférieur à 0.70).

Les meilleurs résultats dans le Tableau 6.6 ont été obtenus avec le système de RàPC qui parvient à faire jeu égal avec la croissance de régions guidées par un expert dans la majorité des cas. Les deux approches permettent d'atteindre un score de Dice moyen de 0.83. Pour le cas 2, nous avons même la surprise de constater que le système de RàPC est parvenu à légèrement dépasser le score de la croissance de régions manuelle (0.89 vs 0.88). Seul le cas 4 montre une différence significative entre les deux (0.74 vs 0.79), indiquant une insuffisance de l'adaptation. Ces résultats confirment ceux obtenus dans le Tableau 6.4 précédent. Globalement, l'adaptation des coordonnées des germes a permis au système de RàPC d'exprimer des performances très similaires à celles d'un humain devant utiliser le même algorithme de segmentation. La technique des *level sets* montre de bons résultats pour la plupart des images, mais a échoué à segmenter le cas 8 correctement. De manière générale, la croissance de régions (avec ou sans RàPC) a réalisé les segmentations les plus pertinentes. Le CNN a en revanche produit les pires

	RàPC-CR avec adapt pos		CR sans RàPC (assistance de l'expert)		Level Set		FCN-8s	
	Dice	IoU	Dice	IoU	Dice	IoU	Dice	IoU
Cas 1	0.92	0.85	0.92	0.85	0.93	0.87	0.73	0.58
Cas 2	0.89	0.80	0.88	0.79	0.88	0.79	0.78	0.64
Cas 3	0.74	0.59	0.75	0.60	0.73	0.58	0.65	0.48
Cas 4	0.74	0.59	0.79	0.65	0.75	0.60	0.52	0.35
Cas 5	0.84	0.72	0.84	0.72	0.83	0.71	0.53	0.36
Cas 6	0.95	0.90	0.95	0.90	0.90	0.83	0.47	0.31
Cas 7	0.82	0.70	0.82	0.69	0.76	0.62	0.59	0.42
Cas 8	0.76	0.62	0.76	0.62	0.18	0.10	0.60	0.42
Cas 9	0.86	0.76	0.86	0.76	0.84	0.73	0.73	0.57
Cas 10	0.76	0.61	0.77	0.63	0.73	0.58	0.35	0.21
Moyenne	0.83	0.71	0.83	0.72	0.75	0.64	0.59	0.43
Médiane	0.83	0.71	0.83	0.70	0.79	0.66	0.59	0.42

TABLE 6.6 – Comparaison de notre système de RàPC avec quelques autres approches pour la segmentation du rein pathologique

scores avec un Dice moyen de seulement 0.59. Ce résultat est logique compte tenu qu'un CNN se base sur l'expérience et nécessite une large base de données pour générer des segmentations intéressantes. Les conditions d'expérimentations étaient donc très défavorables à une approche par *Deep Learning*, alors que le RàPC s'appuie davantage sur de la connaissance.

6.3.2/ L'ADAPTATION DES SEUILS DES GERMES

L'adaptation des coordonnées des germes permet d'améliorer radicalement la qualité des segmentations, principalement lorsque le germe a été initialement positionné à l'extérieur du rein. Cette adaptation trouve également une utilité dans d'autres situations en permettant de positionner le germe dans une configuration proche du cas remémoré (intensité du germe la plus proche possible de l'intensité de référence, voir section 5.1.4.2 pour plus de détails). Cependant, les expérimentations ont démontré une insuffisance de l'adaptation de ce seul paramètre pour produire des segmentations optimales. Ainsi, il est nécessaire de s'intéresser au second paramètre important pour lancer une croissance de régions qui est la valeur des seuils associés aux germes.

6.3.2.1/ LE CONTEXTE DE L'ÉVALUATION

Pour évaluer la seconde étape de l'adaptation (modification des seuils), les expérimentations s'appuient sur une nouvelle base de données que nous avons voulu plus large et plus complète. Cette nouvelle base est composée de 33 images scanners (CT-scans), donnant autant de cas différents. Nous restons donc dans une situation dans laquelle un cas représente une seule et unique coupe d'un examen. Ces 33 images ont été segmentées par une croissance de régions guidée manuellement par un expert pour permettre la modélisation des cas correspondants.

Les cas sont extraits des examens de 3 patients différents, que nous nommerons ici respectivement P1, P2 et P3. Le patient P1 possède un examen constitué de 40 images d'où sont extraites 12 images incorporées dans la base de cas. Le patient P2 possède 5 coupes dont 13 ont été segmentées par un expert puis stockées. Enfin, P3 possède le plus large examen avec 88 clichés scanners dont 8 sont conservés. Le Tableau 6.7 résume les informations sur la constitution de cette base. Notons que nous nous limitons ici pour chaque patient aux coupes dans lesquelles le rein pathologique est présent. Il ne s'agit donc pas des examens entiers.

	Nbre d'images	Nbre d'images dans la base	Nbre d'images testées
P1	40	12	28
P2	55	13	42
P3	88	8	80
Total	183	33	150

TABLE 6.7 – Constitution de la base de cas pour l'évaluation de l'adaptation des seuils

Le choix du nombre de coupes à inclure dans la base pour chaque patient repose sur la recherche d'un compromis entre la minimisation de la taille de cette base et les perfor-

mances du système. D'une part, traiter les coupes pour les segmenter et les incorporer dans la base de cas suivant la formalisation vue dans la section 5.1.2 est une tâche très chronophage. D'autre part, il est intéressant de parvenir à exploiter au maximum la connaissance à disposition en limitant au mieux la nécessité de l'enrichir. Ceci est le rôle d'une phase d'adaptation efficace donnant au système la capacité d'adapter une solution existante pour la rendre pertinente même quand le problème cible n'a pas de correspondance identique dans la base de cas. Suivant la complexité de la tâche de segmentation chez un patient donné, le nombre de coupes nécessaires, pour que l'adaptation de la solution débouche sur des performances satisfaisantes, varie. La quantité d'images retenue pour alimenter la base de cas pour chaque patient, présentée dans le Tableau 6.7, correspond à ce nombre. Typiquement, le patient P3 présente un rein à l'aspect majoritairement sain, rendant donc la segmentation plus simple. L'inclusion de seulement 8 coupes pour ce patient est suffisante quand le patient P1 en nécessite 12 pour éviter des segmentations chaotiques (pas de propagation, phénomène de fuite ou encore segmentation de la mauvaise structure).

Il y a donc un total de 33 images dans la base de cas du système RàPC et celui-ci est évalué sur un ensemble de tests de 150 images. L'adaptation se fait à la fois sur la position et sur les valeurs de seuils. Ces 2 étapes, très liées l'une à l'autre, sont cette fois-ci évaluées conjointement. L'évaluation s'est faite une nouvelle fois par le calcul des scores de Dice et IoU entre les segmentations calculées et les vérités terrain des chirurgiens.

Afin d'améliorer la qualité des segmentations calculées, un post-traitement a été appliqué sur les images en sortie du système RàPC. Le parenchyme rénal étant organisé en 2 tissus distincts que sont le cortex et la médulla (voir section 1.1.2.1 pour plus de détails), il est nécessaire de segmenter ces 2 tissus pour que l'évaluation par rapport aux vérités terrain soit pertinente (les chirurgiens n'ont pas distingué les 2 tissus lors de leurs segmentations). Or, la segmentation de la médulla par une croissance de régions impliquerait un trop grand nombre de germes à placer dans l'image car elle apparaît comme étant composée de nombreuses régions du rein non connectées entre elles sur les clichés scanner. Cependant, dans la grande majorité des situations, une segmentation correcte du cortex rénale isole la médulla du *background* sous la forme de cavités dans le parenchyme. Le post-traitement consiste donc à appliquer un algorithme comblant les cavités fermées (non connectées à l'extérieur du rein) comme illustré par la Figure 6.2.

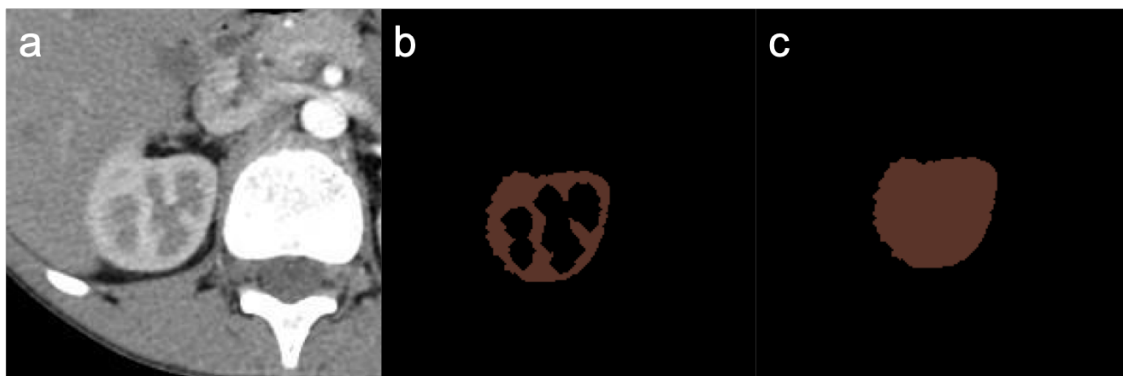


FIGURE 6.2 – Illustration des effets du post-traitement sur les segmentations calculées : (a) CT-scan (b) segmentation du système de RàPC sans post-traitement (c) segmentation avec post-traitement

6.3.2.2/ LA DÉTERMINATION DES POIDS POUR LE CALCUL DU CRITÈRE GLOBAL

L'étape d'adaptation des seuils utilise un score global pour que le système soit en mesure d'évaluer la pertinence d'une segmentation associée à une combinaison de valeurs de seuils. Ce score global passe par le calcul de 3 critères différents que sont la superficie de la segmentation, son centre de masse et l'orientation de la forme. Ces critères sont déterminés par comparaison des valeurs provenant de la segmentation calculée pour le problème cible avec des valeurs de référence stockées dans le cas source, provenant de sa propre segmentation. Ils permettent ainsi de juger de la proximité entre les deux segmentations. Assumant la validité des hypothèses en RàPC (voir section 3.1.2.2), nous pouvons approcher le résultat attendu pour le problème cible avec le résultat obtenu par le passé pour le cas source.

L'expression du score global a été donnée par l'équation 25. Elle utilise un ensemble de 3 poids a , b et c déterminant l'influence de chacun des critères dans le calcul. La détermination de ces poids est empirique. Dans cette partie, nous expérimentons l'adaptation des valeurs de seuils pour les germes, suivant l'algorithme 1, avec différents triplets (a, b, c) .

Nous évaluons ici 3 versions de l'algorithme d'adaptation intitulées respectivement AdaptV1, AdaptV2 et AdaptV3 :

- **AdaptV1** : La première version de l'algorithme d'adaptation ne tient compte que d'un critère sur la superficie de la segmentation pour modifier les valeurs de seuils. Elle équivaut donc à un triplet de poids $(1,0,0)$.
- **AdaptV2** : La deuxième version de l'algorithme d'adaptation tient compte des 3 critères (superficie, centre de masse et orientation) mais sans normalisation. Cette absence de normalisation implique d'utiliser des poids très élevés pour compenser la différence d'échelle entre les critères.
- **AdaptV3** : reprend l'algorithme AdaptV2 mais cette fois en normalisant les critères tels qu'ils sont définis dans la section 5.1.4.2.

Le Tableau 6.8 résume les résultats obtenus pour les 3 patients P1, P2 et P3 en indiquant, pour chacun d'entre eux le score de Dice et le score IoU. Les scores ont été calculés pour chacune des coupes constituant l'examen du patient (sauf celles présentes dans la base de cas), indépendamment les unes des autres, puis un score moyen est reporté dans le tableau. Un grand nombre de combinaisons de poids a été testé durant les expérimentations mais seuls les résultats d'un échantillon de ces combinaisons sont présentés. Pour l'algorithme AdaptV2, par souci de méthodologie (étudier l'effet des critères normalisés nous a paru plus pertinent), nous avons choisi de ne garder que la combinaison donnant, d'après nos tests, les meilleures performances en tant que référence.

En ne prenant en compte que la superficie pour évaluer si une segmentation est bonne, AdaptV1 parvient directement à obtenir de bons scores avec un indice Dice de 0.810 pour le patient P1, 0.827 pour le patient P2 et 0.819 pour le patient P3. L'introduction de 2 critères supplémentaires par AdaptV2 permet d'avoir une nette amélioration pour le patient P3 (0.842) mais donne un résultat très similaire pour les 2 autres. AdaptV3 correspondant à une volonté de corriger un biais méthodologique dans le calcul des critères en les normalisant entre 0 et 1. Ceci permet également de manipuler les poids avec une plus grande facilité. Les meilleurs résultats sont obtenus pour le triplet $(20, 10, 1)$

qui permettent d'augmenter la précision des segmentations pour le patient P3 (0.853) tout en ayant des résultats très proches pour les 2 premiers.

	AdaptV1		AdaptV2 (1,150,4500)		AdaptV3 (1,1,1)		AdaptV3 (8,2,1)		AdaptV3 (20,10,1)	
	Dice	IoU	Dice	IoU	Dice	IoU	Dice	IoU	Dice	IoU
P1	0.810	0.692	0.805	0.683	0.783	0.668	0.784	0.670	0.807	0.687
P2	0.827	0.711	0.831	0.717	0.825	0.713	0.834	0.722	0.832	0.719
P3	0.819	0.729	0.842	0.761	0.853	0.776	0.838	0.756	0.853	0.777

TABLE 6.8 – Mesures de Dice et IoU moyennes pour la segmentation du parenchyme rénale (pathologique) avec notre système de RàPC amélioré par une adaptation de la position des germes et des seuils

Le Tableau 6.9 présente les mêmes résultats mais avec un calcul pour les scores différents. Cette fois, nous appliquons la méthode de calcul dite « globale » comme détaillé dans la section 6.1. Le score est déterminé pour un patient en une seule fois en considérant l'ensemble de l'examen comme une seule et même image 3D. Cette méthode donne des résultats plus pertinents car elle s'affranchit d'un biais de calcul pouvant artificiellement diminuer ou augmenter le score (voir section 6.1). L'évolution des valeurs est globalement la même qu'avec les moyennes mais nous parvenons à des scores plus élevés. Nous nous rendons ainsi compte qu'AdaptV3, avec un triplet de poids (20, 10, 1), améliore la précision des segmentations calculées par rapport à AdaptV1, alors que cette amélioration n'est pas perceptible en utilisant les moyennes (Tableau 6.8). En particulier, les résultats pour le patient P3 deviennent très bons avec un score de Dice de 0.897. La différence de performances entre les patients P1 et P2 et le patient P3 vient du fait que leurs tumeurs respectives sont à des stades différents. Alors que pour les 2 premiers il s'agit d'un diagnostic tardif, avec un néphroblastome très développé déformant énormément le rein, le stade de la tumeur chez le patient P3 est plus précoce. Dans la pratique, cela se traduit par un rein ayant une allure beaucoup plus saine sur une grande partie des coupes de l'examen, rendant la segmentation plus simple. Ces résultats montrent également qu'il est difficile de trouver un triplet de poids (a, b, c) optimal pour tous les patients. Certains triplets peuvent donner le meilleur résultat sur l'un d'entre eux et les pires sur un autre. Le triplet (20, 10, 1) nous apparaît ici comme étant le plus pertinent car il permet d'obtenir de bonnes segmentations sur tous les patients testés, mais celui-ci est susceptible de perdre en pertinence sur d'autres. Plutôt que de chercher des poids consensuels, il pourrait être plus intéressant d'introduire ce paramètre dans la modélisation d'un cas pour le rendre spécifique à un patient. Nous aborderons à nouveau cette évolution possible dans les perspectives de ce travail.

La Figure 6.3 permet de mettre en évidence l'importance de cette phase d'adaptation dans notre système de RàPC, ainsi que son efficacité, avec une série d'exemples. Les deux dernières colonnes correspondent respectivement au résultat d'une segmentation sans et avec la phase d'adaptation.

Pour les 4 images présentées, le système a échoué à correctement segmenter le parenchyme rénal en l'absence d'adaptation. Les raisons expliquant cet échec sont multiples :

- Pour les images 1 et 2, les valeurs de seuils mémorisés ont été trop faibles pour faire croître correctement les régions et produire la bonne segmentation.

	AdaptV1		AdaptV2 (1,150,4500)		AdaptV3 (1,1,1)		AdaptV3 (8,2,1)		AdaptV3 (20,10,1)	
	Dice	IoU	Dice	IoU	Dice	IoU	Dice	IoU	Dice	IoU
P1	0.826	0.704	0.826	0.703	0.651	0.482	0.806	0.675	0.830	0.710
P2	0.817	0.691	0.824	0.701	0.816	0.689	0.827	0.704	0.824	0.700
P3	0.867	0.765	0.888	0.798	0.899	0.816	0.882	0.789	0.897	0.814

TABLE 6.9 – Mesures de Dice et IoU globales pour la segmentation du parenchyme rénale (pathologique) avec notre système de RàPC amélioré par une adaptation de la position des germes et des seuils

- À l'inverse, la segmentation de l'image 4 est un exemple du phénomène de fuite problématique avec ce genre d'algorithme.
- Pour l'image 3, le problème vient d'un mauvais positionnement des germes qui amène à segmenter une grande partie de la tumeur en plus du rein.

Pour ces 3 cas, l'activation de la phase d'adaptation a permis d'augmenter significativement la qualité du résultat, même si celle-ci n'est pas parvenue à produire une segmentation parfaite.

6.3.2.3/ COMPARAISON AVEC UNE APPROCHE DEEP LEARNING PAR $OV^2ASSION$ POUR LA SEGMENTATION DU REIN

La méthode $OV^2ASSION$ est évaluée dans la suite de ce chapitre. Cependant, afin de confirmer le résultat des expérimentations liminaires (section 6.2.2) nous avons effectué une comparaison des performances entre notre système de RàPC (avec sa phase d'adaptation complète) et un CNN (FCN-8s), entraîné d'après notre méthode $OV^2ASSION$, pour la segmentation du parenchyme rénal pathologique. L'intérêt est que cette méthode place le CNN dans une situation favorable (segmentations à calculer proches de celles incluses dans l'ensemble d'apprentissage LS) lui permettant d'obtenir de bonnes performances. Elle représente donc un bon moyen de juger du potentiel de FCN-8s pour effectuer la segmentation du rein par rapport à une approche RàPC telle que nous l'avons conçue.

La base de cas du système de RàPC est identique à celle des expérimentations précédentes. Les patients testés le sont également (P1, P2 et P3). Nous avons utilisé le RàPC avec sa phase d'adaptation complète (coordonnées des germes et valeurs des seuils), avec la combinaison de poids (a, b, c) déterminée comme étant optimale dans la section précédente, à savoir $(20, 10, 1)$. Pour effectuer la comparaison dans des conditions similaires, nous avons paramétré la méthode $OV^2ASSION$ avec un écart $g = 4$ et un vecteur $(V_4)_1$, tel que les données sur lesquelles s'appuient les 2 systèmes d'Intelligence Artificielle soient comparables. La constitution des deux bases de données est présentée dans le Tableau 6.10. Seul le nombre de données pour le patient P3 est sensiblement différent (2 fois plus important pour FCN-8s).

Le Tableau 6.11 présente les scores de Dice et IoU des deux approches, calculés selon la méthode globale pour être plus représentatifs et uniquement sur les images non présentes dans les bases de données respectives. Pour les 3 patients, le système de

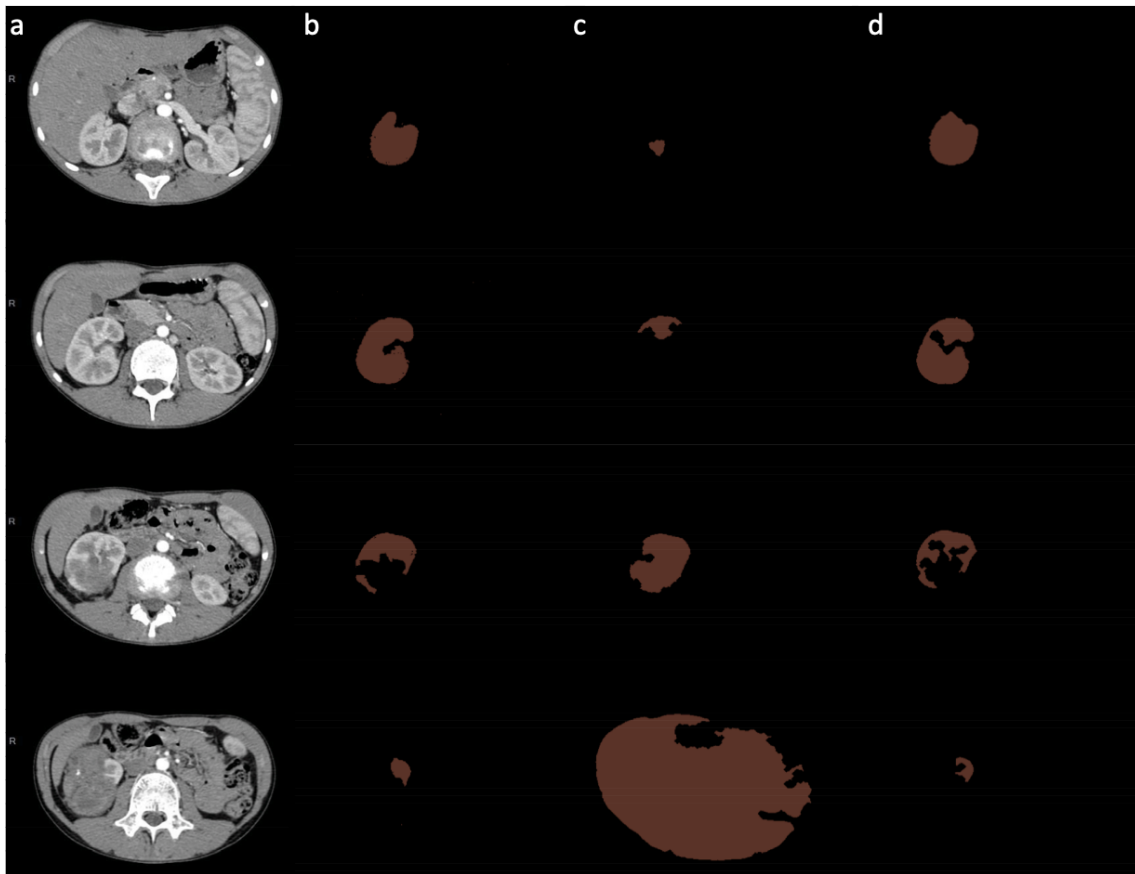


FIGURE 6.3 – Illustrations de l'intérêt de la phase d'adaptation dans notre système de RàPC avec un triplet (20,10,1) : (a) CT-scan (b) vérité terrain (c) segmentation sans adaptation (d) segmentation avec adaptation

	Nbre total d'images	Nbre d'images base de cas du RàPC	Nbre d'images base d'apprentissage de FCN-8s ($g = 4$)
P1	40	12	8
P2	55	13	11
P3	88	8	18

TABLE 6.10 – Contenu des bases de données pour le RàPC et FCN-8s dans le contexte d'une comparaison des performances

RàPC est parvenu, grâce à sa phase d'adaptation, à calculer des segmentations d'une précision plus importante que celles données par FCN-8s. Les deux systèmes offrent des performances très proches pour le patient P3 avec un avantage en faveur du RàPC. Le rein pathologique du patient P3 ayant une apparence saine sur un grand nombre de coupes, ceci peut expliquer pourquoi le CNN parvient à d'aussi bons résultats. Toutefois, le RàPC conserve là encore des performances supérieures tout en s'appuyant sur une quantité de données bien plus faible que FCN-8s (8 images contre 18).

Ces résultats nous confortent dans notre choix d'une part d'utiliser une approche RàPC

	RàPC-CR avec adaptation complète		FCN-8s $OV^2ASSION$ $g = 4$	
	Dice	IoU	Dice	IoU
P1	0.830	0.710	0.763	0.617
P2	0.824	0.700	0.729	0.574
P3	0.897	0.814	0.881	0.788

TABLE 6.11 – Comparaison des scores de Dice et IoU pour la segmentation du parenchyme rénale (pathologique) entre notre système de RàPC (avec adaptation complète) et FCN-8S entraîné selon la méthode $OV^2ASSION$

pour la segmentation du rein et d'autre part de réserver une approche *Deep Learning* pour la tumeur.

6.4/ L'ÉVALUATION D'UN CNN ENTRAÎNÉ SUIVANT LA MÉTHODE $OV^2ASSION$

Pour l'évaluation des performances d' $OV^2ASSION$, nous nous plaçons dans l'hypothèse où un échantillon des images de l'ensemble des coupes (constituant l'examen du patient) a été correctement segmenté au préalable. L'ambition de ces tests est de déterminer le nombre d'images scanner initialement nécessaires au CNN pour s'entraîner afin de calculer des segmentations avec une précision suffisante. Pour ces résultats, nous avons considéré, en accord avec la méthode $OV^2ASSION$ décrite dans la section 5.2.1, que les images incorporées dans l'ensemble d'apprentissage LS correspondent à des coupes régulièrement espacées dans l'examen suivant un écart g .

6.4.1/ L'INFLUENCE DE L'ÉCART g ET DU VECTEUR $(V_g)_i$ SUR LES PERFORMANCES

Les premiers résultats de la segmentation de la tumeur sont présentés dans le Tableau 6.12. Comme pour toutes les expérimentations précédentes, nous avons évalué les performances de notre approche en calculant les scores de Dice et IoU entre les vérités terrain (fournies par les chirurgiens) et les segmentations prédites (calculées par le CNN). Ces premiers résultats ont pour objectif d'observer l'influence d'une part de l'espacement g et d'autre part du choix du vecteur $(V_g)_i$ (conditionnant la position des coupes incluses dans LS) sur la précision des segmentations calculées par le système. Les impacts de l'écart et de la position des coupes sont évaluées simultanément. Pour chaque écart g , nous avons testé le réseau seulement sur les coupes non incluses dans l'ensemble d'apprentissage LS , à l'exception de l'écart $g = 0$ pour lequel toutes les coupes de l'examen y sont présentes. L'espacement $g = 0$ nous a servi de référence pour les étapes d'entraînement et de tests qui ont donc été réalisées sur les mêmes données. La deuxième colonne du Tableau 6.12 correspond aux différents vecteurs $(V_g)_i$ possibles pour un même écart g (l'espacement est le même mais il y a différents décalages possibles). Le premier vecteur a une valeur de 1 pour son premier élément, le deuxième vecteur a une valeur de 1

pour son deuxième élément et ainsi de suite. Chaque vecteur i décale ses valeurs de 1 position par rapport au vecteur $i - 1$ et ainsi de suite jusqu'à retomber sur la configuration du vecteur $(V_g)_1$. Nous calculons une moyenne et une médiane des scores (Dice et IoU) pour chacun des vecteurs possibles suivant l'espacement g utilisé et un score général pour chaque écart (moyenne des scores obtenus pour les vecteurs de cet espacement). La dernière colonne indique si le vecteur correspondant permet d'inclure les deux extrémités de la tumeur (si la coupe supérieure et la coupe inférieure sont toutes les deux dans l'ensemble d'apprentissage LS).

6.4.1.1/ L'ENTRAÎNEMENT DU CNN

Nous avons repris l'architecture FCN-8s pour nos expérimentations de la méthode $OV^2ASSION$, pour ses performances et son faible nombre de couches en comparaison aux autres CNN de la littérature. Nous utilisons les mêmes taux d'apprentissage/*learning rate* (1^{-12}), momentum (0.99) et décroissance des poids/*weight decay* (0.0005) que dans l'article original de Long *et al.* [long2015fcn] et nous avons repris les paramètres pré-entraînés sur la base PASCAL VOC 2012 [everingham2015pvc] pour les couches conv1 à fc7. La dernière convolution du réseau a été modifiée pour fonctionner avec 2 canaux afin de permettre la prédiction de deux scores, le *background* et la tumeur, au lieu des 21 labels de la base PASCAL VOC 2012. Les données d'entraînement proviennent du patient P2 pour un total de 61 coupes. Le nombre de coupes de l'examen diffère de ceux exploitées dans la section 6.3.2.3 pour ce même patient car la tumeur ayant une taille plus importante que le rein pathologique, celle-ci se retrouve sur davantage d'images. Le réseau a donc été entraîné sur un ensemble d'apprentissage maximum de 61 coupes, suivant l'espacement sélectionné, provenant d'un seul patient.

D'un point de vue matériel, l'entraînement du réseau s'est déroulé sur un GPU Tesla Kepler K40 du Mésocentre de l'Université de Franche-Comté. Plusieurs vecteurs ont été testés avec différents espacements, allant de 0 à 10, et pour chacun d'entre eux, nous avons évalué toutes les possibilités conformément à $Card_{V_{test}}$ (Equation 37). Nous avons donc testé 65 possibilités de vecteurs, chaque vecteur ayant une taille de 61 éléments déterminant quelles sont les coupes utilisées pour l'apprentissage. Pour le vecteur avec un espacement $g = 0$, toutes les coupes ont été sélectionnées pour l'entraînement.

6.4.1.2/ L'INFLUENCE DE L'ESPACEMENT

Le Tableau 6.12 résume les valeurs de scores pour un écart g allant de 0 à 5, selon les différents vecteurs, mais les évaluations ont été faites jusqu'à un espacement $g = 10$. Comme nous pouvions le prévoir, les meilleurs résultats sont obtenus pour un espacement nul ($g = 0$), pour lequel les tests sont réalisés sur les données d'apprentissage, avec un score de Dice de 0.926 et un écart-type de 0.106. Si nous traçons la courbe des scores moyens en fonction de l'espacement (plusieurs vecteurs pouvant correspondre à un même espacement), nous obtenons la Figure 6.4. Nous ne pouvons pas observer de distinctions claires concernant un écart particulier qui serait incontestablement meilleur que les autres. La diminution des scores est globalement linéaire mais nous pouvons toutefois observer une faible différence de ces scores entre un espacement de $g = 2$ et un espacement de $g = 3$ (avec un Dice approximatif d'un peu moins de 0.90) alors que passer de l'un à l'autre correspond à une réduction substantielle de 15% de la quantité de

Écart	Vecteur	Dice moyen ± écart-type	IoU moyen ± écart-type	Dice moyen de l'écart	IoU moyen de l'écart	Dice médian	IoU médian	Dice médian de l'écart	IoU médian de l'écart	Les 2 extrémités incluses ?
0	V_0	0.926 ± 0.106	0.875 ± 0.135	0.926	0.875	0.955	0.914	0.955	0.914	Oui
1	$(V_1)_1$	0.916 ± 0.121	0.861 ± 0.149	0.910	0.852	0.952	0.908	0.950	0.906	Oui
	$(V_1)_2$	0.903 ± 0.138	0.843 ± 0.167			0.949	0.902			Non
2	$(V_2)_1$	0.914 ± 0.129	0.856 ± 0.165	0.899	0.840	0.947	0.899	0.945	0.896	Oui
	$(V_2)_2$	0.896 ± 0.136	0.832 ± 0.169			0.937	0.882			Non
	$(V_2)_3$	0.887 ± 0.196	0.832 ± 0.205			0.951	0.906			Non
3	$(V_3)_1$	0.925 ± 0.134	0.874 ± 0.173	0.897	0.836	0.961	0.925	0.944	0.894	Oui
	$(V_3)_2$	0.890 ± 0.140	0.824 ± 0.175			0.936	0.880			Non
	$(V_3)_3$	0.884 ± 0.169	0.820 ± 0.190			0.936	0.879			Non
	$(V_3)_4$	0.888 ± 0.174	0.827 ± 0.188			0.943	0.892			Non
4	$(V_4)_1$	0.920 ± 0.114	0.867 ± 0.141	0.885	0.816	0.949	0.903	0.936	0.880	Oui
	$(V_4)_2$	0.881 ± 0.148	0.810 ± 0.180			0.930	0.870			Non
	$(V_4)_3$	0.887 ± 0.154	0.821 ± 0.180			0.941	0.888			Non
	$(V_4)_4$	0.882 ± 0.169	0.817 ± 0.189			0.937	0.881			Non
	$(V_4)_5$	0.853 ± 0.207	0.783 ± 0.219			0.924	0.859			Non
5	$(V_5)_1$	0.903 ± 0.152	0.846 ± 0.172	0.880	0.816	0.946	0.897	0.935	0.878	Oui
	$(V_5)_2$	0.889 ± 0.141	0.820 ± 0.160			0.927	0.863			Non
	$(V_5)_3$	0.884 ± 0.153	0.816 ± 0.176			0.940	0.886			Non
	$(V_5)_4$	0.873 ± 0.184	0.807 ± 0.202			0.937	0.881			Non
	$(V_5)_5$	0.871 ± 0.190	0.810 ± 0.209			0.932	0.872			Non
	$(V_5)_6$	0.862 ± 0.221	0.799 ± 0.225			0.930	0.870			Non

TABLE 6.12 – Évaluation des segmentations de la tumeur prédites par FCN-8s, entraîné selon la méthode $OV^2ASSION$, en fonction de l'espacement et du vecteur position utilisés

données d'apprentissage. Un écart optimal pourrait donc être $g = 3$, qui donne un score Dice de 0.897 avec 26% de coupes initialement segmentées, afin d'obtenir les meilleures performances avec un minimum de données à labelliser en amont. Cependant, pour que cette méthode soit utilisable dans la pratique, il est nécessaire de minimiser au mieux les données nécessaires pour l'apprentissage. Dans cette optique, un écart $g = 4$ semble être un bon compromis pratique car permettant une bonne segmentation à partir d'un examen étiqueté à 20%, ce qui est acceptable pour les chirurgiens. Même avec un espacement $g = 10$, FCN-8s parvient tout de même à produire des segmentations avec un score moyen Dice de 0.83 (ceci à partir d'un examen étiqueté à seulement 10%).

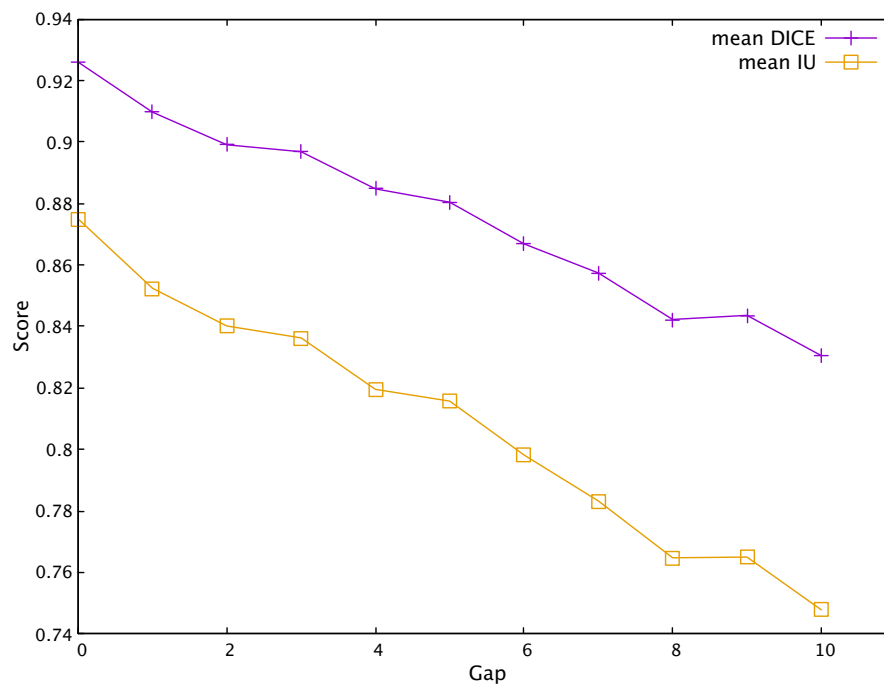


FIGURE 6.4 – Évolution des scores moyens de Dice et IoU en fonction de l'espacement utilisé

6.4.1.3/ L'INFLUENCE DE LA POSITION

Les différents vecteurs dans le Tableau 6.12 permettent de déterminer l'influence de la position des coupes retenues pour constituer l'ensemble d'apprentissage LS . Pour chaque valeur d'écart g , les meilleurs vecteurs (ceux donnant les scores les plus élevés) ont été mis en évidence en gris. La dernière colonne apporte une information supplémentaire en indiquant si les 2 extrémités de l'examen sont compris ou non dans LS . Ceci dépend à la fois du nombre de coupes dans l'examen en question et de la valeur g utilisée pour réaliser l'entraînement. Pour que les 2 extrémités appartiennent à LS , une double condition doit être vérifiée :

$$\begin{cases} ((V_g)_i)_0 = 1 \\ n \bmod (g + 1) = 1 \end{cases} \quad (42)$$

- La première condition impose que la première coupe (première extrémité) soit incluse dans l'ensemble d'entraînement en vérifiant la valeur du premier élément du

vecteur. Il s'agit donc concrètement de tous les vecteurs $(V_g)_1$.

- Ensuite la seconde condition vérifie si la dernière coupe est incluse également ou non, sachant que la première est présente, en fonction du nombre de coupes n et de l'espacement g .

Pour les données du Tableau 6.12, ces 2 conditions sont validées pour tous les vecteurs $(V_g)_1$ testés.

Les résultats montrent tout d'abord que les meilleures performances sont obtenues par les vecteurs $(V_g)_1$, quel que soit l'espacement évalué. Ce qui indique que la présence des extrémités dans l'ensemble d'entraînement a effectivement un impact sur la qualité des segmentations prédites par le réseau. Ainsi, une tendance se dégage avec une diminution progressive des scores (à quelques exceptions près) lorsque le décalage s'accroît. Par exemple, le score moyen de Dice pour les vecteurs $(V_2)_1$, $(V_2)_2$ et $(V_2)_3$ sont respectivement de 0.914, 0.896 et 0.887. Les écarts types suivent une tendance inverse avec pour ces exemples des valeurs respectives de 0.129, 0.136 et 0.196. Ainsi, moins les extrémités sont prises en compte dans LS , plus les performances du réseau se dégradent autant en terme de précision des segmentations qu'en terme de stabilité (plus les écarts-types sont importants plus il y a une dispersion des résultats autour de la moyenne). Ces résultats peuvent être expliqués par le fait que la tumeur présente généralement une forme et un aspect plus imprévisible au niveau de ses extrémités. À ce niveau de la tumeur, le passage d'une coupe à une autre peut entraîner un changement important de l'aspect du néphroblastome sur les images. Il peut donc être important d'inclure ces extrémités dans LS pour maximiser la pertinence du modèle élaboré durant l'entraînement et augmenter les performances du réseau sur les segmentations non apprises.

Le Tableau 6.12 nous montre également que les scores médians sont plus élevés que les scores moyens. Le corollaire de ces résultats est l'existence de valeurs extrêmes plus faibles tirant vers le bas les scores moyens. Il y a donc spécifiquement des coupes que le réseau n'est pas parvenu à segmenter correctement alors que ses performances sont très bonnes sur la majorité d'entre elles. Ces coupes correspondent aux extrémités de la tumeur, confirmant l'interprétation précédente.

6.4.2/ VUE D'ENSEMBLE DES PERFORMANCES D' $OV^2ASSION$ SUR PLUSIEURS PATIENTS

Les tests ont été réalisés sur une cohorte de 8 patients différents, issus de la base de données du projet ASKid. Trois des patients sont ceux utilisés pour effectuer l'évaluation de la phase d'adaptation du RàPC (section 6.3.2.1) et la comparaison des performances entre le RàPC et le *Deep Learning* pour la segmentation du parenchyme rénal (section 6.3.2.3). La base a été complétée par 5 autres patients afin de la rendre plus représentative et augmenter ainsi la pertinence des résultats. Le Tableau 6.13 expose plus précisément le contenu de cette base de données ayant servi à l'apprentissage et à l'évaluation du système $OV^2ASSION$ (associé à une architecture FCN-8s). Tous les tests ont été réalisés avec un espacement $g = 4$, en utilisant le premier vecteur $(V_4)_1$ (correspond à la prise en compte systématique de la première coupe de l'examen) et un entraînement en 10000 itérations. Tout comme lors de l'évaluation du système de RàPC, les examens n'ont pas été pris en compte dans leur totalité et nous nous sommes concentrés sur les coupes dans lesquelles la tumeur est visible. À ce stade, l'intervention manuelle dans le

processus est nécessaire, que ce soit pour sélectionner les coupes à segmenter ou pour effectuer une segmentation préalable afin de lancer l'entraînement du réseau.

Pour chaque patient, le réseau FCN-8s a été initialement pré-entraîné sur la base Pascal VOC 2012 [everingham2015pvc]. De la même manière que dans la section 6.4.1.1, la dernière convolution du réseau a été modifiée pour fonctionner avec 2 canaux afin de permettre la prédiction de deux scores (*background* et tumeur).

	<i>OV²ASSION</i> $g = 4, (V_4)_1$		
	Nbre d'images total	Nbre d'images base d'apprentissage de FCN-8s	Nbre d'images base de test de FCN-8s
P1	47	10	37
P2	61	13	48
P3	88	18	70
P4	30	6	24
P5	59	12	47
P6	82	17	65
P7	37	8	29
P8	26	6	20
Total	430	90	340

TABLE 6.13 – Base de données utilisée pour l'entraînement du réseau FCN-8s selon la méthode *OV²ASSION*

Ainsi, la base de données est composée au total de 430 images dont 90 ont servi pour l'entraînement de FCN-8s et 340 pour l'évaluation à proprement parlé en tant que données tests. Chaque patient a été segmenté et évalué de manière indépendante des autres, en reprenant l'entraînement à partir du modèle pré-entraîné.

Le Tableau 6.14 donne l'ensemble des scores pour évaluer les performances du réseau sur les 8 patients. Il présente les scores moyens (moyenne des scores obtenus pour chaque coupe composant l'examen du patient), associés à leur écart-type pour estimer la dispersion, ainsi que les scores globaux.

	Dice moyen \pm écart-type	IoU moyen \pm écart-type	Dice global	IoU global
P1	0.849 \pm 0.131	0.754 \pm 0.144	0.884	0.792
P2	0.867 \pm 0.149	0.788 \pm 0.180	0.917	0.846
P3	0.840 \pm 0.173	0.754 \pm 0.203	0.899	0.816
P4	0.893 \pm 0.107	0.821 \pm 0.152	0.925	0.861
P5	0.870 \pm 0.139	0.789 \pm 0.158	0.902	0.821
P6	0.906 \pm 0.115	0.844 \pm 0.153	0.945	0.896
P7	0.793 \pm 0.192	0.688 \pm 0.208	0.844	0.731
P8	0.734 \pm 0.221	0.615 \pm 0.214	0.796	0.661

TABLE 6.14 – Évaluation des performances du réseau FCN-8s entraîné selon la méthode *OV²ASSION* sur une cohorte de 8 patients

Les scores de Dice moyens varient de 0.734 pour le patient P8 à 0.906 pour le patient P6. Comme attendu, les scores globaux montrent de meilleures performances avec des scores variant jusqu'à 0.945 pour le patient P6. Les patients P7 et P8 sont ceux ayant été les moins bien segmentés. Ceci s'explique par la faible taille de la tumeur avec une superficie proche d'un rein pathologique. Comme l'ont montré nos expérimentations liminaires (section 6.2.2), FCN-8s voit ses performances chuter pour la segmentation de petites structures. La Figure 6.5 illustre la segmentation de quelques coupes du patient P8. Ces résultats ne sont donc pas surprenants face à d'autres patients présentant une tumeur à un stade avancé, d'une superficie importante sur les images (Figure 6.6) même si un problème similaire se retrouve justement au niveau des extrémités des tumeurs, quel que soit leur taille.

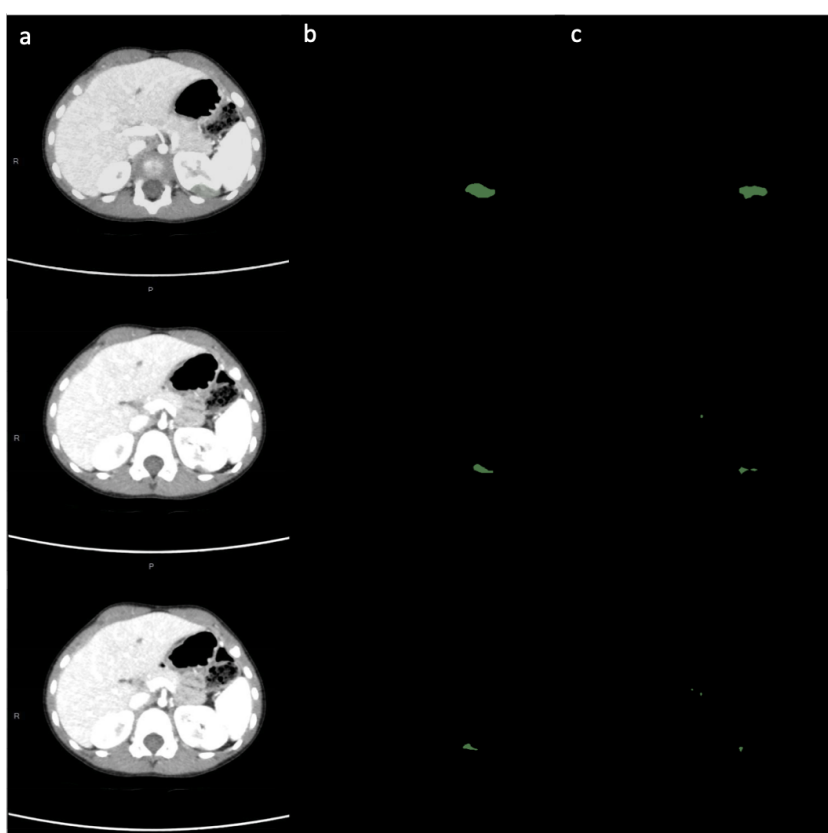


FIGURE 6.5 – Illustration d'une segmentation du patient P8 : (a) CT-scan (b) vérité terrain (c) segmentation par FCN-8s

D'une manière globale, les performances de FCN-8s, couplé à la méthode $OV^2ASSION$, restent satisfaisantes. La précision des segmentations prédites pourrait être améliorée davantage grâce à l'utilisation de post-traitements. Tout d'abord, l'usage d'un algorithme de fermeture (similaire dans l'esprit à celui appliqué aux segmentations du système de RàPC) pourrait éliminer certaines imprécisions du réseau en rendant les segmentations plus homogènes alors qu'un algorithme d'ouverture permettrait l'élimination du bruit potentiel dans l'image. Ensuite, l'usage de *Conditionnal Random Field* (CRF) est largement répandu dans la littérature pour améliorer la qualité des segmentations en sortie d'un réseau de neurones. Cependant, nous nous sommes concentrés pour ces expérimentations sur les performances du système $OV^2ASSION$ en lui-même.

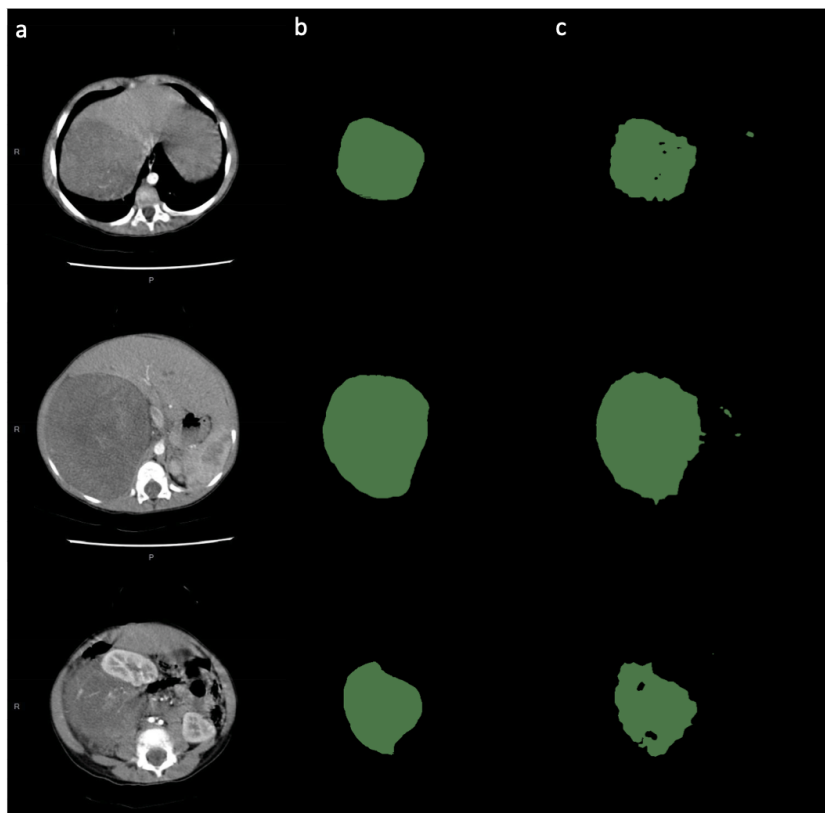


FIGURE 6.6 – Illustration d'une segmentation du patient P6 : (a) CT-scan (b) vérité terrain (c) segmentation par FCN-8s

6.5/ DISCUSSION GÉNÉRALE SUR LES RÉSULTATS DES ÉVALUATIONS ET LEURS LIMITES

Tout d'abord, nos résultats ont démontré que notre système de RàPC pouvait sensiblement améliorer la précision des segmentations du parenchyme rénal, malgré de fortes déformations induites par la présence du néphroblastome. En utilisant cette approche, les reins pathologiques ont été mieux segmentés qu'avec les autres méthodes testées et comparées. Ces bons résultats sont fortement liés à l'existence d'une phase d'adaptation au sein du système pour permettre d'optimiser une solution existante pour un nouveau problème. Celle-ci permet tout d'abord d'améliorer la probabilité que les germes soient correctement placés tout en s'assurant de la cohérence du pixel germe en terme d'intensité.

Nous avons ajouté à cet algorithme une contrainte spatiale sous la forme d'une fenêtre de recherche maximale afin de limiter la distance par rapport à leur position initiale. Ceci est en accord avec l'une des hypothèses fortes derrière un système de RàPC, à savoir que de faibles différences entre 2 problèmes correspondent à de faibles différences dans les solutions permettant de résoudre ces problèmes (hypothèse de cohérence, voir la section 3.1.2.2). L'ajout de cette contrainte supplémentaire permet également de limiter les risques d'un germe placé dans la mauvaise structure si celle-ci possède un niveau de gris caractéristique proche de celui du parenchyme rénal, même si ce risque demeure. Ensuite, l'adaptation se fait dans un second temps au niveau des valeurs de seuils, en

lien étroit avec les coordonnées des germes. Sans parvenir à produire systématiquement les résultats espérés, l'adaptation de la solution mémorisée a permis d'éliminer complètement les phénomènes de fuites pour les données testées. Une comparaison des performances lorsque cette adaptation est activée ou désactivée a clairement mis en évidence cet apport.

La principale limite du système actuel reste néanmoins la faible taille de la base de cas que l'adaptation ne permet pas de compenser complètement lorsque l'on cherche à segmenter un nouveau patient. Bien entendu, cette limite devient d'autant plus problématique lorsque le rein de ce patient est d'un aspect original (inattendu) par rapport aux expériences vécues par le système. Au-delà de la question d'une forme/position inédite de la tumeur (et donc du rein par extension), il se pose également le problème de la latéralité du néphroblastome. Il existe des situations dans lesquelles les caractéristiques du rein à segmenter peuvent être très proches d'un cas stocké, ce qui devrait logiquement entraîner une segmentation pertinente, mais pour lesquelles le rein pathologique se trouve de l'autre côté dans ce nouveau problème cible par rapport à ce cas stocké. Ceci aboutit alors à une incapacité du système à calculer une segmentation correcte alors que, selon toutes vraisemblances, toutes les conditions sont réunies pour que l'algorithme soit efficace. La conséquence est une sous-exploitation de la connaissance à disposition du système, impliquant de compléter la base de cas pour maintenir son efficacité et prévoir les cas miroirs. Si l'on ajoute le fait qu'inclure de nouveaux cas est très chronophage, l'exploitation optimale de cette connaissance est un point essentiel à travailler. Il convient également de s'interroger sur la pertinence des critères de score utilisés pour l'adaptation des seuils, à savoir la superficie, le centre de masse et l'orientation de la forme segmentée. Ces critères apparaissent comme étant très intéressants pour décrire l'image mais ils sont insuffisants à l'usage pour garantir la convergence de l'algorithme vers la meilleure segmentation possible. Ainsi, la détermination des poids à donner pour chacun de ces critères demeure une piste d'amélioration que nous retrouverons dans nos perspectives.

Puis, nous avons proposé et évalué une méthode d'entraînement singulière, *OV²ASSION*, pour permettre l'utilisation de CNN sur un faible jeu de données. Nous avons montré que cette méthode permettait d'obtenir des résultats relativement précis (score de Dice aux alentours de 0.90 pour la plupart des patients) avec un entraînement sur seulement 20% des coupes préalablement segmentées (correspond à un espacement $g = 4$). L'influence du vecteur, conditionnant l'inclusion ou non d'une coupe dans l'ensemble d'apprentissage, s'est avérée non négligeable. Il a été mis en évidence que le score des segmentations était corrélé à l'inclusion des extrémités de la tumeur dans *LS*. Lorsque les extrémités ne sont pas incluses, la précision des segmentations sur nos données a tendance à diminuer. Ceci peut s'expliquer par la conjonction de 2 phénomènes :

- D'une part, nous avons constaté expérimentalement que les coupes au niveau des extrémités des tumeurs avaient tendance à être moins homogènes entre elles. Dit autrement, le nombre de différences d'une coupe à une autre augmentent au niveau de ces extrémités, permettant de réduire l'intérêt d'un sur-apprentissage.
- D'autre part, les structures à segmenter ont une superficie beaucoup plus faible, amenant le CNN sur un terrain qui ne lui est pas favorable.

Rappelons que le principal intérêt médical de ces segmentations est de visualiser l'interface entre la tumeur et le rein pathologique, afin de pouvoir qualifier l'invasion des tissus sains (et notamment des cavités rénales) par les tissus cancéreux. Les extrémités ont

donc finalement assez peu d'impact sur la pertinence médicale des segmentations.

Toutefois, l'influence du vecteur position (Tableau 6.12) s'est effectué dans un contexte dans lequel il était possible d'inclure les 2 extrémités ce qui valide la double condition (Équation 42). Il est évident que dans la pratique, il y a souvent des situations dans lesquelles cette configuration n'est pas possible. Dans ces cas, les vecteurs optimaux ne sont probablement pas ceux incluant l'une des extrémités mais plutôt ceux offrant le meilleur compromis pour représenter de manière harmonieuse les singularités de la tumeur. Ainsi, l'existence d'une variabilité différente suivant la partie du néphroblastome concernée, qui est assez faible en son centre et qui augmente vers les extrémités, pousse à se demander si l'usage d'un écart constant pour construire l'ensemble d'apprentissage LS ne conduirait pas en fait à un sur-apprentissage non homogène. Le choix de l'espacement, quant à lui, ne peut en réalité se faire sans un objectif en terme de performance. Nous avons choisi dans la majorité de nos expérimentations un écart $g = 4$ car celui-ci nous semblait le plus pertinent afin de limiter l'intervention humaine dans le processus. Cependant, ces résultats doivent être mis en perspectives avec les besoins des chirurgiens concernant la précision des segmentations afin qu'ils soient médicalement exploitables. Ceci pourrait également être évalué en mesurant la variabilité intra-chirurgien (variabilité moyenne lors de la réalisation de plusieurs segmentations successives des mêmes images) et inter-chirurgiens (variabilité moyenne lors de la réalisation de mêmes segmentations par des médecins différents). En fonction de ces résultats, le protocole $OV^2ASSION$ pourrait être adapté, ou bien en améliorant autant que nécessaire la qualité des segmentations (par l'affinement potentiel des paramètres d'entraînement ou par l'application de post-traitements), ou bien en revoyant à la baisse les ambitions de la méthode et en diminuant l'écart g utilisé.

Synthèse du chapitre

Dans ce chapitre, nous avons présenté l'ensemble de nos expérimentations autour des 2 contributions majeures de ce travail de thèse, ainsi que les résultats correspondants. Les premières expérimentations nous ont permis d'orienter le choix de l'approche utilisée en fonction de la structure à segmenter : une approche RàPC pour la segmentation du parenchyme rénale et une approche *Deep Learning* pour la segmentation du néphroblastome. Une comparaison entre les performances des architectures FCN-8s et U-Net sur nos données nous a également permis de privilégier FCN-8s. Nous avons ensuite évalué d'une part le système RàPC, et en particulier l'apport de sa phase d'adaptation, et d'autre part les performances de FCN-8s après un entraînement selon la méthode *OV²ASSION* avec différents paramètres.

Pour le RàPC, les résultats ont démontré que notre système pouvait sensiblement améliorer la précision des segmentations du parenchyme rénal, malgré de fortes déformations induites par la présence du néphroblastome (scores de Dice global variant de 0.830 à 0.897 pour les 3 patients évalués). En utilisant cette approche, les reins pathologiques ont été mieux segmentés qu'avec les autres méthodes testées et comparées. Ces bons résultats sont fortement liés à l'existence d'une phase d'adaptation au sein du système pour permettre d'optimiser une solution existante pour un nouveau problème. Les expérimentations ont également permis de déterminer un triplet de poids (20, 10, 2) optimal pour l'adaptation des seuils sur nos données actuelles. Il est cependant possible que ces poids ne soient pas efficaces quel que soit le patient (ainsi il est possible manuellement de les modifier). En parallèle, la méthode *OV²ASSION* FCN-8s est parvenue à obtenir un score de Dice moyen de 0.885 sur le patient évalué, pour un apprentissage effectué à partir d'un étiquetage préalable de seulement 20% des données. En reprenant cet espacement et en calculant cette fois-ci les scores globaux, FCN-8s parvient à des performances comprises entre 0.796 et 0.945 suivant le patient. Cette précision pourrait encore être améliorée par l'application de post-traitements tels que des ouvertures/fermetures ou des CRF.

CONCLUSION ET PERSPECTIVES

CONCLUSION GÉNÉRALE

Les travaux présentés dans ce mémoire ont consisté à formaliser une plate-forme dédiée à la segmentation d'images médicales de type CT-scan (images scanner), et intitulée COLISEUM-3D (**COL**laborative platform with artificial Intelligence for **SE**gmentation of **tU**moral kidney in **M**edical images in **3D**). Plus précisément, il s'agissait de proposer une solution automatisée pour faciliter la prise en charge de jeunes patients atteints par un néphroblastome, une tumeur infantile rénale représentant une urgence médicale de par sa rapidité d'évolution. L'imagerie est en effet un outil primordial pour permettre la meilleure orientation de la thérapie, en accord avec les recommandations de la Société Internationale d'Oncologie Pédiatrique (SIOP), notamment en fonction de l'envahissement tumoral des cavités rénales.

Le cœur de ce travail a permis de proposer 2 solutions de segmentation par IA pour extraire des images des structures d'intérêts, que sont le parenchyme rénale et le néphroblastome, avec un verrou important que représentait la quantité de données limitée à disposition. La première solution dédiée à la segmentation du parenchyme exploite un système de Raisonnement à Partir de Cas (RàPC) pour guider un algorithme de croissance de régions. En particulier, nous avons imaginé une phase d'adaptation pour 2 des paramètres importants pour initialiser un tel algorithme, à savoir les coordonnées des germes et les valeurs de seuils. Nous avons pu mettre en évidence l'efficacité de cette adaptation et la nette amélioration des performances qu'elle induit. Sa présence a permis de fortement limiter les phénomènes de fuite, qui sont courants lorsque l'on effectue une segmentation par croissance de régions. Cette adaptation a également permis d'augmenter les probabilités de segmenter la structure souhaitée en plaçant correctement les germes dans l'image.

La seconde contribution se concentre sur la segmentation du néphroblastome par une approche *Deep Learning* intitulée *OV²ASSION* (*Overlearning Vector for Valid Sparse Segmentations*). Il s'agit d'une méthode d'entraînement pour un CNN permettant de s'affranchir de la nécessité habituelle d'avoir accès à une grande quantité d'images étiquetées par un expert pour réaliser l'apprentissage. Le principe est de segmenter manuellement au préalable un échantillon de coupes de la tumeur. Cet échantillon devant être représentatif, les segmentations se font avec un espacement régulier afin d'avoir un modèle tenant compte de l'ensemble des variations de la tumeur sur toute sa hauteur. Les résultats obtenus sont prometteurs et permettent d'atteindre des performances élevées dès lors que l'espacement n'est pas trop important.

Ainsi, d'un côté notre système de RàPC est parvenu à segmenter les images de 3 patients avec un score global de Dice allant de 0.824 à 0.897. Ces bons résultats ont largement été rendus possibles grâce à l'existence d'une phase d'adaptation efficace, permettant de pallier le manque de cas stockés dans la base en modifiant une solution

pré-existante. Les expérimentations menées ont permis d'aboutir à un paramétrage optimal de l'adaptation des seuils sur les patients testés avec un triplet de poids (20, 10, 1), commandant respectivement l'influence de la superficie, du centre de masse et de l'orientation de la forme produite dans l'évaluation des segmentations possibles suivant différentes combinaisons de seuils. Une comparaison des performances entre le système de RàPC et notre méthode *OV²ASSION*, réalisée dans des conditions similaires pour les 2 approches, a permis de confirmer la primauté du RàPC pour la segmentation du parenchyme rénal. Concernant les performances de *OV²ASSION* pour la segmentation de la tumeur, nous sommes parvenus à obtenir avec un espacement $g = 4$ un score de Dice moyen de 0.885 sur un patient, à partir de la segmentation en amont d'un nombre de coupes représentant 20% seulement de l'examen. Des évaluations complémentaires ont confirmé ces résultats sur l'ensemble de nos données, constitué de 8 patients différents représentant un total de 430 images (réparties en un ensemble d'apprentissage de 90 images et un ensemble de tests de 340). Exprimés en score global de Dice sur les 8 patients évalués, la segmentation du néphroblastome donne des résultats situés entre 0.796, pour le patient avec le score le plus faible, et 0.945 pour celui avec le score le plus élevé.

Au final, ces 2 modules de segmentation pour COLISEUM-3D ont été implémentés en donnant de bonnes performances. Les travaux concernant le recalage des images, préalable nécessaire à toute fusion des segmentations, ont également bien avancé et ont jeté les bases de futurs approfondissements. Les perspectives qui suivent permettent de donner de nombreuses pistes de poursuite pour ces travaux.

PERSPECTIVES

Ces travaux offrent différentes perspectives qui nous paraissent très intéressantes afin d'une part d'améliorer les performances et la robustesses des solutions proposées, et d'autre part de compléter la plate-forme COLISEUM-3D en concevant les modules manquants pour la rendre opérationnelle. Certaines de ces perspectives concernent l'amélioration du système de RàPC pour la segmentation du parenchyme rénal, notamment vis-à-vis de sa phase d'adaptation. D'autres se concentrent plutôt sur une amélioration des performances pour la segmentation du néphroblastome par *Deep Learning*.

A- AMÉLIORATION DU SYSTÈME DE RÀPC POUR LA SEGMENTATION DU REIN

La première perspective intéressante pour le futur de la plate-forme COLISEUM-3D est l'amélioration des performances et de la robustesse du système de RàPC et de son algorithme de croissance de régions. Cette amélioration possible concerne en particulier la phase d'adaptation du système.

L'un des problèmes actuels de ce système, abordée dans la section 6.5 est la faible taille de la base de cas qui ne permet pas de faire face à certaines situations inédites. Un exemple est celle d'une latéralité du néphroblastome différente qui empêche le système de correctement réaliser la segmentation alors même que la connaissance à sa disposition, que ce soit en terme de forme, d'intensité de pixels ou encore de position relative par rapport à la tumeur, devrait être suffisante pour la mener à bien. Il suffirait ainsi au système pour résoudre ce problème de placer les germes en miroir par rapport à un point

de repère (comme la colonne vertébrale). Nous sommes donc ici dans une situation typique dans laquelle la connaissance du système est sous-exploitée. Il s'agit clairement d'une limite à notre adaptation qui demeure trop rigide. Il serait intéressant de modifier cette adaptation pour y inclure une étape de construction d'une nouvelle solution. Plutôt que de se remémorer une solution entière à partir d'un unique cas stocké, et d'adapter cette solution *a posteriori*, nous gagnerions davantage de flexibilité en construisant une solution originale à partir de plusieurs cas différents de la base, apportant chacun une brique de connaissance utile pour résoudre le cas cible.

Concernant plus précisément l'étape d'adaptation des seuils pour la croissance de régions, son efficacité dépend beaucoup du choix des poids (a, b, c) . Les résultats de nos expérimentations nous ont amené à privilégier un certain triplet ayant empiriquement donné les meilleures performances sur nos données. Cependant, rien ne nous permet d'affirmer que ce jeu de poids est le plus efficace quel que soit le patient traité. La suite logique de ces expérimentations est de les élargir à une cohorte de patients plus importante afin de tenter de déterminer des poids universels. Une autre approche envisageable, et *a priori* plus efficace, serait d'inclure ce jeu de poids dans la solution des cas. Ainsi, lors de la phase de remémoration, ces poids seraient remémorés de la même manière que les coordonnées des germes et les seuils associés. Il semble pas aberrant de partir du principe que deux patients suffisamment similaires partagent un même triplet (a, b, c) optimal. Ceci rendrait au final le système de RàPC moins dépendant de la quantité de données utilisée pour le concevoir, ce qui serait en adéquation avec les objectifs de ce travail.

B- UTILISATION D'UN ESPACEMENT ADAPTATIF POUR *OV²ASSION*

Lors de l'entraînement d'un CNN par *OV²ASSION*, nous partons du principe que la meilleure manière d'obtenir un ensemble d'apprentissage *LS* représentatif de ladite tumeur est de sélectionner à intervalle régulier des coupes qui sont, après segmentation manuelle pour en faire des vérités terrains, incluses dans *LS*. L'objectif est de parvenir à « capturer » les variations de la tumeur sur toute sa hauteur. Néanmoins, il apparaît que la variabilité de la tumeur n'est pas constante sur toute cette hauteur. Typiquement, le centre de la tumeur varie bien moins d'une coupe à une autre que ses extrémités. Dit autrement, les coupes au centre de la tumeur sont représentatives de plus de coupes que ne le sont celles situées aux extrémités. En conséquence, l'ensemble d'apprentissage *LS* serait plus efficace en incluant davantage de coupes issues des extrémités et moins de coupes issues du centre. Il pourrait donc être pertinent d'utiliser sur le néphroblastome un espacement adaptatif suivant cette variabilité (avec par exemple un écart de 1 ou 2 au niveau des extrémités et un écart de 4 au centre) plutôt qu'un espacement constant. Nous pouvons soit imaginer appliquer une série d'espacements prédéfinie sur tous les néphroblastomes, soit calculer l'espacement qui serait localement optimal pour chaque nouvelle tumeur. Un moyen d'y parvenir serait de déterminer un indice de similarité entre une coupe et les coupes voisines. Une haute similarité entraînerait un écart g important quand une similarité plus faible aurait pour conséquence d'utiliser localement un espacement plus restrictif.

C- $OV^2ASSION-r$: UNE VARIANTE DE $OV^2ASSION$

La méthode $OV^2ASSION$ consiste à désigner des coupes pour construire l'ensemble d'apprentissage LS . Le processus est binaire dans le sens où soit une coupe est incluse, et dans ce cas elle compte de la même manière que les autres dans l'entraînement, soit elle est exclue complètement et le réseau ne l'utilise pas pour son apprentissage. $OV^2ASSION-r$ est une variante de $OV^2ASSION$ partant de l'hypothèse qu'il pourrait être plus intéressant de segmenter de manière itérative les coupes, tout en ajustant le modèle (issu de l'apprentissage) au fur et à mesure du processus, plutôt que de simplement créer un modèle et de l'appliquer sur la totalité des coupes à segmenter en une seule fois.

Le processus serait décomposé en 3 étapes distinctes. Tout d'abord, LS est construit selon le vecteur $(V_g)_h$ utilisé. Un premier apprentissage est ensuite réalisé à partir des coupes segmentées initialement (que ce soit manuellement dans notre cas ou par une autre technique quelle qu'elle soit pourvu qu'elle donne un résultat suffisamment précise) pour calculer un premier modèle. Ce modèle permet de segmenter les coupes qui sont adjacentes aux coupes contenues dans LS . Enfin, LS est mis à jour, puis un nouveau modèle est calculé pour pouvoir effectuer les segmentations suivantes. Ces 3 étapes sont répétées autant que nécessaire pour segmenter la totalité de la tumeur.

Toutefois, toutes les coupes ne devraient pas avoir la même pertinence dans l'apprentissage (une coupe issue d'une segmentation manuelle est réputée plus fiable qu'une coupe obtenue à l'aide du CNN) et il faudrait introduire des poids pour chacune d'entre elles afin que leur influence ne soit pas égale. La Figure 6.7 illustre l'exécution de la méthode $OV^2ASSION-r$ sur 3 itérations dans lesquelles des poids décroissants (ici arbitraires pour l'illustration) sont appliqués pour l'apprentissage sur les nouvelles coupes segmentées, afin de tenir compte d'une erreur supposée croissante suivant la distance avec la plus proche vérité terrain.

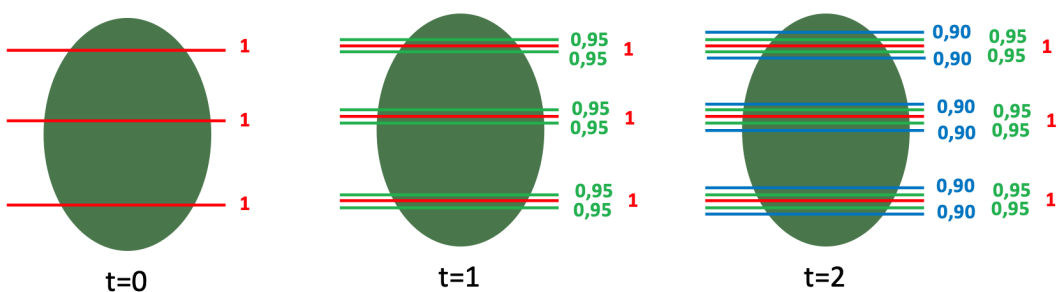


FIGURE 6.7 – Principe itératif d'une segmentation selon la méthode $OV^2ASSION-r$

Ainsi, plutôt que d'utiliser des vecteurs avec des éléments binaires, il serait souhaitable d'introduire des vecteurs à valeurs réelles qui pourraient être assimilables à des poids.

La formulation est très similaire à celle de $OV^2ASSION$ et elle en reprend les bases. De la même manière que pour $OV^2ASSION$, soit V l'ensemble de toutes les combinaisons possibles, le sous-ensemble V_g correspondant à l'écart g se définit par :

$$V_g \subset V = \{([0, 1]_1 \dots [0, 1]_n)\} \quad (43)$$

Où leurs éléments peuvent cette fois prendre des valeurs réelles comprises entre 0 et 1. Tout comme dans l'équation 33, V_g représente un sous-ensemble de vecteurs et chacun des vecteurs $(V_g)_i$ appartenant à ce sous-ensemble est désigné par son index i . De la même manière que pour $OV^2ASSION$, $(V_g)_i$ est utilisé pour construire progressivement l'ensemble LS entre chaque itération/apprentissage, mais cette fois-ci en pondérant l'influence des coupes ajoutées suivant l'erreur supposée qu'elles introduisent dans LS .

La construction de l'ensemble d'apprentissage LS se fait en appliquant une fonction f suivant le vecteur $(V_g)_i$ qui est définie par :

$$f : LS \times [0, 1] \rightarrow LS \mid \forall i \in \{1, \dots, n\} : f(S_i, w_{i-1}) = w_{i-1} \times S_i \quad (44)$$

La principale différence avec $OV^2ASSION$ est que l'action effectuée ne serait plus binaire. Les coupes S_i sont toutes incorporées dans LS suivant le poids w_i qui leur est attribué. Évidemment, un poids de 0 revient tout de même à exclure la coupe correspondante. Aussi, $OV^2ASSION - r$ implique de modifier dynamiquement le vecteur $(V_g)_i$, en changeant le poids de certaines coupes, à chaque itération pour compléter LS et incorporer progressivement les nouvelles segmentations de manière pondérée.

Toute la question se trouve dans la détermination des poids pour chacune des segmentations calculées. De manière rigoureuse, il serait souhaitable de pouvoir évaluer la fiabilité des segmentations *a posteriori* et d'adapter les poids correspondants en conséquence. Mais ceci exigerait d'effectuer cette évaluation par rapport à une vérité terrain que nous n'aurions pas à disposition dans une situation réelle. Il serait donc nécessaire de concevoir une fonction $w(i, g)$ donnant un poids w_i en fonction de l'espacement g utilisé et de la position i de la coupe en question. Nous imaginons cette fonction comme étant périodique pour retranscrire une évolution périodique de l'erreur : partant d'un poids maximum pour une coupe segmentée manuellement, puis diminuant en s'éloignant de cette coupe pour ensuite augmenter à nouveau en se rapprochant de la prochaine vérité terrain. Cette fonction pourrait également être adaptable à un pas particulier, suivant le résultat d'expérimentation et la quantification des erreurs du CNN, pour traduire une évolution plus au moins rapide du poids d'une coupe à une autre.

D- CONCEPTION D'UN SYSTÈME HYBRIDE RÀPC/DEEP LEARNING POUR LA SEGMENTATION DE LA TUMEUR

Les perspectives précédentes ont eu pour objectif d'améliorer les systèmes de RàPC et de CNN de manière indépendante afin d'en optimiser les performances respectives. Une nouvelle méthode pourrait consister à *hybrider* ces approches pour exploiter leurs points forts respectifs. En ce qui concerne la segmentation de la tumeur par $OV^2ASSION$, nous partons pour chaque nouveau patient d'un réseau certes pré-entraîné, mais pas sur un patient souffrant d'un néphroblastome. L'une des alternatives seraient d'avoir une approche plus classique en réalisant un pré-entraînement sur une cohorte de patients à notre disposition avant d'effectuer toute segmentation d'un nouveau patient. Ceci se heurte toutefois à nouveau au manque de données dont nous disposons pour le moment. Un moyen de contourner ce problème serait d'utiliser le système de RàPC déjà conçu pour la segmentation du parenchyme rénal. Lors de cette segmentation, le cas stocké le

plus proche du cas cible est remémoré et avec lui sa solution. Actuellement, la solution du RàPC ne concerne que le paramétrage de l'algorithme de croissance de régions. Cependant, il est tout à fait envisageable d'y inclure un jeu de poids (modèle) d'un CNN entraîné sur le patient remémoré pour la segmentation de la tumeur et repris comme point de départ pour la segmentation par *OV²ASSION* sur le nouveau patient. Ceci reviendrait à utiliser un réseau de neurones pré-entraîné sur le patient le plus proche déjà traité dans la base de cas. L'impact précis sur les performances est à déterminer, mais l'usage de réseaux pré-entraînés est une pratique largement répandue dans la littérature et qui a fait ses preuves. Cette nouvellement approche renforcerait la collaboration entre les différentes méthodes IA au sein de COLISEUM-3D et l'interopérabilité des modules qui composent la plate-forme.

E- SEGMENTATION DES CAVITÉS RÉNALES ET DES VAISSEAUX SANGUINS

Pour la segmentation des cavités sur les images en temps tardif, un protocole basé sur un second système de RàPC paraît tout à fait envisageable. Il est toutefois difficile d'appliquer un algorithme de croissance de régions dans cette situation. Contrairement au parenchyme rénal qui peut être en 1, 2 ou 3 parties maximum sur les images, il peut y avoir plusieurs parties distinctes formant le rein, les cavités rénales peuvent former un grand nombre de petites structures qui exigeraient tout autant de germes à placer pour parvenir à produire la segmentation. Cette approche serait donc assez peu efficace et délicate à conduire. Les conditions pour effectuer une bonne segmentation seraient trop contraignantes et même notre phase d'adaptation aurait en l'état du mal à compenser les erreurs. Il est toutefois possible d'exploiter une particularité des cavités par rapport au parenchyme : celles-ci concentrent une grande partie du produit de contraste en temps tardif, rendant leur intensité de pixels sur les clichés très élevée et caractéristique. Un simple seuillage serait donc suffisant pour parvenir à effectuer les segmentations à condition de trouver un moyen d'ignorer les os, ces derniers ayant également une intensité élevée. Les niveaux de gris des cavités étant très proches de celui des structures osseuses, un seuillage généralisé sur l'ensemble de l'image inclurait également ces cavités. Le RàPC permettrait d'apporter une solution originale à ce problème en rendant possible la remémoration d'une fenêtre délimitant une zone de l'image qui contiendrait les cavités. La segmentation serait ainsi le résultat d'un seuillage limité à cette fenêtre. Pour ce faire, il suffirait d'inclure dans la solution d'un cas les coordonnées de cette fenêtre, 2 cas similaires ayant de fortes chances d'avoir une localisation de leurs cavités respectives proches. Des tests supplémentaires sont néanmoins nécessaires pour évaluer la fiabilité de cette approche dans la pratique.

La segmentation des vaisseaux sanguins est beaucoup plus délicate, dans la mesure où ils sont difficilement discernés sur les images scanners (y compris par œil humain entraîné). La plus grande difficulté se trouve à la fois dans un contraste pouvant être très faible avec l'environnement immédiat des vaisseaux et dans l'imprévisibilité de leur localisation due à la présence du néphroblastome. La plupart du temps, les chirurgiens parviennent à effectuer cette segmentation de proche en proche. Ils segmentent les vaisseaux sanguins sur une coupe dans laquelle celle-ci est facile, ce qui les aide à les localiser sur les coupes voisines. Certaines fois, les vaisseaux sont complètement indiscernables et seule une connaissance *a priori* de cette localisation rend la segmentation possible. Nous pouvons envisager une méthodologie similaire consistant à segmenter dans un premier temps des coupes dans lesquelles les vaisseaux sanguins sont visibles,

puis seulement dans un second temps à s'occuper des images dans lesquelles la segmentation est plus complexe. Le RàPC, qui permet d'exploiter au mieux la connaissance, a très certainement un rôle à jouer une fois les premières segmentations réalisées. Un algorithme de croissance de régions semble toutefois inadapté dans ce contexte et il serait nécessaire de définir une nouvelle approche pour résoudre ce problème de segmentation.

F- PRISE EN COMPTE DES MOUVEMENTS RESPIRATOIRES LORS DU RECALAGE

Entre les clichés en temps vasculaire et ceux en temps tardif, il s'écoule un certain laps de temps qui rend nécessaire une étape de recalage pour compenser les mouvements involontaire du patient (ainsi que des paramètres d'acquisition pouvant être différents). Les mouvements qui correspondent à des translations et des rotations qui peuvent être corrigées par un recalage rigide. Néanmoins, un tel recalage ne permet pas de tenir compte des mouvements respiratoires qui induisent une déformation des tissus en plus d'un déplacement des organes. Ce problème peut être résolu en ayant recours à un recalage de type élastique, que l'on appelle également non linéaire ou encore non rigide. Dans ce cas, le recalage rigide sert d'initialisation au recalage élastique pour faciliter l'optimisation (recherche de la meilleure transformation à appliquer à l'image).

MA BIBLIOGRAPHIE PERSONNELLE

ARTICLES EN REVUES RÉFÉRENCÉES

- [marie19] Florent Marie, Lisa Corbat , Yann Chaussy, Thibault Delavelle, Julien Henriet et Jean-Christophe Lapayre. Segmentation of deformed kidneys and nephroblastoma using Case-Based Reasoning and Convolutional Neural Network. Expert Systems with Applications (journal Q1). Volume 127, 2019, Pages : 282 - 294

PUBLICATIONS EN CONFÉRENCE INTERNATIONALE AVEC COMITÉ DE LECTURE D'AUDIENCE INTERNATIONALE, DONT LES ACTES SONT PUBLIÉS

- [marie18] Florent Marie, Lisa Corbat , Yann Chaussy, Thibault Delavelle, Julien Henriet et Jean-Christophe Lapayre. Segmentation of Kidneys Deformed by Nephroblastoma Using Case-Based Reasoning. International Conference on Case-Based Reasoning (classé B dans core). Volume 127, 2018, Pages : 233 - 248. Best Student Paper Award

BIBLIOGRAPHIE

- [**aamodt1994cbr**] Agnar Aamodt and Enric Plaza. Case-based reasoning : Foundational issues, methodological variations, and system approaches. *AI communications*, 7(1) :39–59, 1994.
- [**alchourron1985revisionTheory**] Carlos Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change : Partial meet functions for contraction and revision. *Journal of Symbolic Logic*, 50(51) :0–530, 1985.
- [**attig2009study**] Anja Attig and Petra Perner. A study on the case image description for learning the model of the watershed segmentation. *Transactions on Case-Based Reasoning*, 2(1) :41–53, 2009.
- [**attig2012watershedSeg**] Anja Attig and Petra Perner. Incremental learning of the model for watershed-based image segmentation. In *Combinatorial Image Analysis*, pages 209–222. Springer, 2012.
- [**badri2015segnet**] Vijay Badrinarayanan, Ankur Handa, and Roberto Cipolla. Segnet : A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling. *arXiv preprint arXiv :1505.07293*, 2015.
- [**barioni2008metricAccess**] Maria Camila N Barioni, Humberto L Razente, Agma JM Traina, and Caetano Traina Jr. Accelerating k-medoid-based algorithms through metric access methods. *Journal of Systems and Software*, 81(3) :343–355, 2008.
- [**bichindaritz2010cbrHs**] Isabelle Bichindaritz and Cindy Marling. Case-based reasoning in the health sciences : Foundations and research directions. In *Computational Intelligence in Healthcare 4*, pages 127–157. Springer, 2010.
- [**binh2012improving**] Huynh Thi Thanh Binh, Mai Dinh Loi, and Nguyen Thi Thuy. Improving image segmentation using genetic algorithm. In *Machine Learning and Applications (ICMLA), 2012 11th International Conference on*, volume 2, pages 18–23. IEEE, 2012.
- [**brostow2009camVid**] Gabriel J Brostow, Julien Fauqueur, and Roberto Cipolla. Semantic object classes in video : A high-definition ground truth database. *Pattern Recognition Letters*, 30(2) :88–97, 2009.
- [**bueno2007geneticSim**] Renato Bueno, Agma JM Traina, and Caetano Traina Jr. Genetic algorithms for approximate similarity queries. *Data & Knowledge Engineering*, 62(3) :459–482, 2007.
- [**burgos2009cbrSeg**] Xavier P Burgos-Artizzu, Angela Ribeiro, Alberto Tellaeché, Gonzalo Pajares, and Cesar Fernández-Quintanilla. Improving weed pressure assessment using digital images from an experience-based reasoning approach. *Computers and electronics in agriculture*, 65(2) :176–185, 2009.

- [chandhok2012approach]** Chinki Chandhok, Soni Chaturvedi, and AA Khurshid. An approach to image segmentation using k-means clustering algorithm. *International Journal of Information Technology (IJIT)*, 1(1) :11–17, 2012.
- [chen1998kmeans]** Chang Wen Chen, Jiebo Luo, and Kevin J Parker. Image segmentation via adaptive k-mean clustering and knowledge-based morphological operations with biomedical applications. *IEEE transactions on image processing*, 7(12) :1673–1683, 1998.
- [chen2016multiScale]** Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L Yuille. Attention to scale : Scale-aware semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3640–3649, 2016.
- [chen2018deeplab]** Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab : Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4) :834–848, 2018.
- [chouraqui1986analogie]** Eugène Chouraqui. Le raisonnement analogique : sa problématique, ses applications. *Actes des Journées Nationales sur l'Intelligence Artificielle, Aix-les-Bains*, pages 107–117, 1986.
- [christ2016unetMed]** Patrick Ferdinand Christ, Mohamed Ezzeldin A Elshaer, Florian Ettliger, Sunil Tatavarty, Marc Bickel, Patrick Bilic, Markus Rempfler, Marco Armbruster, Felix Hofmann, Melvin D’Anastasi, et al. Automatic liver and lesion segmentation in ct using cascaded fully convolutional neural networks and 3d conditional random fields. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 415–423. Springer, 2016.
- [ciccek20163dUnet]** Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net : learning dense volumetric segmentation from sparse annotation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 424–432. Springer, 2016.
- [cordier2006adaptation]** Amélie Cordier, Béatrice Fuchs, and Alain Mille. Engineering and learning of adaptation knowledge in case-based reasoning. In *International Conference on Knowledge Engineering and Knowledge Management*, pages 303–317. Springer, 2006.
- [daquin2006adaptation]** Mathieu D’aquin, Jean Lieber, and Amedeo Napoli. Adaptation knowledge acquisition : A case study for case-based decision support in oncology. *Computational intelligence*, 22(3-4) :161–176, 2006.
- [devignes2018analogy]** Marie-Dominique Devignes, Yohann Fransot, Yves Lepage, Jean Lieber, Emmanuel Nauer, and Malika Smaïl-Tabbone. First steps toward finding relevant pathology-gene pairs using analogy. In *Workshop on Evolutionary Computation and CBR–EvoCBR 2018, at ICCBR-2018*, 2018.
- [dreyer1982prozessanalyse]** Helmut Dreyer and Wilfried Sauer. *Prozeßanalyse : elementare stochastische Methoden*. Verlag Technik, 1982.

- [dufour2012adaptation]** Valmi Dufour-Lussier, Florence Le Ber, Jean Lieber, and Laura Martin. Adapting spatial and temporal cases. In *International Conference on Case-Based Reasoning*, pages 77–91. Springer, 2012.
- [everingham2015pvc]** Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge : A retrospective. *International Journal of Computer Vision*, 111(1) :98–136, 2015.
- [ficet1999cbr]** Valérie Ficet-Cauchard, Christine Porquet, and Marinette Revenu. Cbr for the reuse of image processing knowledge : a recursive retrieval/adaptation strategy. In *International Conference on Case-Based Reasoning*, pages 438–452. Springer, 1999.
- [frucci2006watershed]** Maria Frucci. Oversegmentation reduction by flooding regions and digging watershed lines. *International Journal of Pattern Recognition and Artificial Intelligence*, 20(01) :15–38, 2006.
- [frucci2008watershedCBR]** Maria Frucci, Petra Perner, and Gabriella Sanniti Di Baja. Case-based-reasoning for image segmentation. *International Journal of Pattern Recognition and Artificial Intelligence*, 22(05) :829–842, 2008.
- [fuchs2000adaptation]** Béatrice Fuchs, Jean Lieber, Alain Mille, and Amedeo Napoli. An algorithm for adaptation in case-based reasoning. In *14th European Conference on Artificial Intelligence-ECAI'2000*, pages 45–49. IOS Press, 2000.
- [fukushima1982neocognitron]** Kunihiko Fukushima and Sei Miyake. Neocognitron : A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets*, pages 267–285. Springer, 1982.
- [glorot2011rectifier]** Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323, 2011.
- [goel2018reinforcementLearning]** Vik Goel, Jameson Weng, and Pascal Poupart. Un-supervised video object segmentation for deep reinforcement learning. *arXiv preprint arXiv :1805.07780*, 2018.
- [grimnes1996imageUnderstanding]** Morten Grimnes and Agnar Aamodt. A two layer case-based reasoning architecture for medical image understanding. In *European Workshop on Advances in Case-Based Reasoning*, pages 164–178. Springer, 1996.
- [gu2017cbrCancerDiagnosis]** Dongxiao Gu, Changyong Liang, and Huimin Zhao. A case-based reasoning system based on weighted heterogeneous value distance metric for breast cancer diagnosis. *Artificial intelligence in medicine*, 77 :31–47, 2017.
- [he2014pyramidPooling]** Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *European conference on computer vision*, pages 346–361. Springer, 2014.

- [he2016resNet]** Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [hemanth2009survey]** D Jude Hemanth, C Kezi Selva Vijila, and J Anitha. A survey on artificial intelligence based brain pathology identification techniques in magnetic resonance images. *Int. J. Rev. Comp*, 4 :30–45, 2009.
- [henriet2014equivox]** Julien Henriët, Pascal Chatonnay, and Pierre-Emmanuel Leni. An iterative precision vector to optimise the cbr adaptation of equivox. *Engineering Applications of Artificial Intelligence*, 35 :158–163, 2014.
- [hong2015decoupledNet]** Seunghoon Hong, Hyeonwoo Noh, and Bohyung Han. Decoupled deep neural network for semi-supervised semantic segmentation. In C Cortes, N D Lawrence, D D Lee, M Sugiyama, and R Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1495–1503. Curran Associates, Inc., 2015.
- [hudelot2008fuzzySeg]** Céline Hudelot, Jamal Atif, and Isabelle Bloch. Fuzzy spatial relation ontology for image interpretation. *Fuzzy Sets and Systems*, 159(15) :1929–1951, 2008.
- [jaiswal2013survey]** Varshali Jaiswal and Aruna Tiwari. A survey of image segmentation based on artificial intelligence and evolutionary approach. *IOSR Journal of Computer Engineering (IOSR-JCE)*, 15(3) :71–78, 2013.
- [jarmulak1998cbrClassif]** Jacek Jarmulak. Case-based classification of ultrasonic b-scans : Case-base organisation and case retrieval. In *European Workshop on Advances in Case-Based Reasoning*, pages 100–111. Springer, 1998.
- [jevtie2010adaptive]** Aleksandar Jevtić and Diego Andina. Adaptive artificial ant colonies for edge detection in digital images. In *IECON 2010-36th Annual Conference on IEEE Industrial Electronics Society*, pages 2813–2816. IEEE, 2010.
- [kalinovsky2016segnetMed]** A Alexander Kalinovsky and Vassili Kovalev. Lung image segmentation using deep learning methods and convolutional neural networks. In *Pattern Recognition and Information Processing - PRIP*. Minsk : Publishing Center of BSU, 2016.
- [kamnitsas2017deepMedic]** Konstantinos Kamnitsas, Christian Ledig, Virginia FJ Newcombe, Joanna P Simpson, Andrew D Kane, David K Menon, Daniel Rueckert, and Ben Glocker. Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation. *Medical image analysis*, 36 :61–78, 2017.
- [kohli2009crf]** Pushmeet Kohli, Philip HS Torr, et al. Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision*, 82(3) :302–324, 2009.
- [kolodner1983cyrus]** Janet L Kolodner. Reconstructive memory : A computer model. *Cognitive science*, 7(4) :281–328, 1983.
- [kolodner1996principlesCBR]** Janet L Kolodner. Making the implicit explicit : Clarifying the principles of case-based reasoning. *Case-based Reasoning : Experiences, Lessons and Future Directions*, pages 349–370, 1996.

- [koltun2012fullyCrf]** Vladlen Koltun and Philipp Krähenbühl. Efficient inference in fully connected crfs with gaussian edge potentials. *NIPS, CoRR*, pages 1–9, 2012.
- [krahenbuhl2011fullyConnectedCrf]** Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011.
- [krizh2012alexNet]** Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.
- [lecun1988backpropa]** Yann Le Cun, D Touresky, G Hinton, and T Sejnowski. A theoretical framework for back-propagation. In *Proceedings of the 1988 Connectionist Models Summer School*, pages 21–28. CMU, Pittsburgh, Pa : Morgan Kaufmann, 1988.
- [lecun1989generalization]** Yann LeCun et al. Generalization and network design strategies. *Connectionism in perspective*, pages 143–155, 1989.
- [lecun1998gradient]** Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) :2278–2324, 1998.
- [li2018hdenseunet]** Xiaomeng Li, Hao Chen, Xiaojuan Qi, Qi Dou, Chi-Wing Fu, and Pheng-Ann Heng. H-denseunet : hybrid densely connected unet for liver and tumor segmentation from ct volumes. *IEEE transactions on medical imaging*, 37(12) :2663–2674, 2018.
- [lieber2007conservativeAdapt]** Jean Lieber. Application of the revision theory to adaptation in case-based reasoning : The conservative adaptation. In *International Conference on Case-Based Reasoning*, pages 239–253. Springer, 2007.
- [lieber2008hdr]** Jean Lieber. *Contributions à la conception de systèmes de raisonnement à partir de cas*. PhD thesis, Université Henri Poincaré-Nancy I, 2008.
- [litjens2017Survey]** Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen AWM van der Laak, Bram van Ginneken, and Clara I Sánchez. A survey on deep learning in medical image analysis. *arXiv preprint arXiv :1702.05747*, 2017.
- [long2015fcn]** Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.
- [malek2001hybridCbr]** Maria Malek. Hybrid approaches for integrating neural networks and case based reasoning : From loosely coupled to tightly coupled models. In *Soft computing in case based reasoning*, pages 73–94. Springer, 2001.
- [melis1998reformulation]** Erica Melis, Jean Lieber, and Amedeo Napoli. Reformulation in case-based reasoning. In *European Workshop on Advances in Case-Based Reasoning*, pages 172–183. Springer, 1998.

- [meurie2005segmentation]** Cyril Meurie. *Segmentation d'images couleur par classification pixellaire et hiérarchie de partitions*. PhD thesis, Ecole doctorale SIMEM, Université de CAEN-Basse Normandie, France, 2005.
- [micarelli2000cbrRecognition]** Alessandro Micarelli, Alessandro Neri, and Giuseppe Sansonetti. A case-based approach to image recognition. In *European Workshop on Advances in Case-Based Reasoning*, pages 443–454. Springer, 2000.
- [mignotte2014criterionModel]** Max Mignotte and Charles Hélou. A precision-recall criterion based consensus model for fusing multiple segmentations. *Int J Signal Process Image Process Pattern Recognit*, 7(3) :61–82, 2014.
- [milletari2016unet]** Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net : Fully convolutional neural networks for volumetric medical image segmentation. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 565–571. IEEE, 2016.
- [minsky1975framework]** M MINSKY. A framework for representing knowledge. *The Psychology of Computer Vision*, 1975.
- [nasiri2015medicalCbr]** Sara Nasiri, Johannes Zenkert, and Madjid Fathi. A medical case-based reasoning approach using image classification and text information for recommendation. In *International Work-Conference on Artificial Neural Networks*, pages 43–55. Springer, 2015.
- [negny2010effective]** Negny Stéphane, Riesco Hector, et al. Effective retrieval and new indexing method for case based reasoning : application in chemical process design. *Engineering Applications of Artificial Intelligence*, 23(6) :880–894, 2010.
- [netter2014atlas]** Frank H Netter. *Atlas d'anatomie humaine, 6e éd., traduit par Pierre Kamina et Jean-Pierre Richer*. Issy-les-Moulineaux : Elsevier Masson, 2014.
- [ng2006kmeans]** HP Ng, SH Ong, KWC Foong, PS Goh, and WL Nowinski. Medical image segmentation using k-means clustering and improved watershed algorithm. In *Image Analysis and Interpretation, 2006 IEEE Southwest Symposium on*, pages 61–65. IEEE, 2006.
- [nguyen2018fusionModel]** Dac Cong Tai Nguyen, Said Benameur, Max Mignotte, and Frédéric Lavoie. Superpixel and multi-atlas based fusion entropic model for the segmentation of x-ray images. *Medical image analysis*, 48 :58–74, 2018.
- [noh2015deconvNet]** Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *IEEE International Conference on Computer Vision, ICCV 2015*, pages 1520–1528, Washington, DC, USA, 2015. IEEE Computer Society.
- [papandreou2015semisupervised]** George Papandreou, Liang-Chieh Chen, Kevin P Murphy, and Alan L Yuille. Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1742–1750, 2015.
- [perner1999segCBR]** Petra Perner. An architecture for a cbr image segmentation system. *Engineering Applications of Artificial Intelligence*, 12(6) :749–759, 1999.

- [perner2001cbrForImages]** Petra Perner. Why case-based reasoning is attractive for image interpretation. In *International Conference on Case-Based Reasoning*, pages 27–43. Springer, 2001.
- [perner2011knowledge]** Petra Perner and Anja Attig. Using prototype-based classification for automatic knowledge acquisition. In *Pattern Recognition, Machine Intelligence and Biometrics*, pages 197–212. Springer, 2011.
- [perner2016cbrForImages]** Petra Perner. Model development and incremental learning based on case-based reasoning for signal and image analysis. In *International Symposium Computational Modeling of Objects Represented in Images*, pages 3–24. Springer, 2016.
- [plantazTumWilms]** Dominique Plantaz. Néphroblastome ou tumeur de wilms. Accessed : 2017-10-11, 2005.
- [pal2004cbr]** Sankar K Pal and Simon CK Shiu. *Foundations of soft case-based reasoning*, volume 8. John Wiley & Sons, 2004.
- [ray1999colorSegmentation]** Siddheswar Ray and Rose H Turi. Determination of number of clusters in k-means clustering and application in colour image segmentation. In *Proceedings of the 4th international conference on advances in pattern recognition and digital techniques*, pages 137–143. Calcutta, India, 1999.
- [ray2016fcnMed]** Austin Ray. Lung tumor segmentation via fully convolutional neural networks. Technical report, Stanford University, 2016.
- [richter2003fourKnowledge]** Michael M Richter. Knowledge containers. *Readings in Case-Based Reasoning*. Morgan Kaufmann Publishers, 2003.
- [ricci2007svm]** Elisa Ricci and Renzo Perfetti. Retinal blood vessel segmentation using line operators and support vector classification. *IEEE transactions on medical imaging*, 26(10) :1357–1365, 2007.
- [rohlfing2007shape]** Torsten Rohlfing and Calvin R Maurer. Shape-based averaging. *IEEE Trans. Image Processing*, 16(1) :153–161, 2007.
- [ronneberger2015unet]** Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net : Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, 2015.
- [rosenblatt1958perceptron]** F Rosenblatt. The perceptron : a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6) :386–408, 1958.
- [riesbeck1989cbr]** Christopher K Riesbeck and Roger C Schank. *Inside case-based reasoning*. Lawrence Erlbaum Associates, Inc, 1989.
- [rumelhart1988learning]** David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3) :1, 1988.

- [schank1977scripts]** Roger C Schank and Robert P Abelson. *Scripts, plans, goals and understanding : An inquiry into human knowledge structures*. Lawrence Erlbaum, 1977.
- [schank1987memoryExpertSystem]** R Schank. Memory-based expert systems. Technical report, YALE UNIV NEW HAVEN CT, 1987.
- [schank1987memory]** R Schank. Memory-based expert systems. Technical report, YALE UNIV NEW HAVEN CT, 1987.
- [simonyan2014very]** Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv :1409.1556*, 2014.
- [siteWebGustRouss]** Néphroblastome. <https://www.gustaveroussy.fr/fr/nephroblastome>. Accessed : 2018-12-12.
- [siteWebitk]** Itk web site. <http://www.itksnap.org>. Accessed : 2019-01-07.
- [siteWebslicer]** 3d slicer web site. <http://slicer.org>. Accessed : 2019-01-07.
- [sherwood2016physiologie]** Lauralee Sherwood, Hillar Klandorf, and Paul Yancey. *Physiologie animale*. De Boeck Superieur, 2016.
- [smyth1998adaptationCriterion]** Barry Smyth and Mark T Keane. Adaptation-guided retrieval : questioning the similarity assumption in reasoning. *Artificial intelligence*, 102(2) :249–293, 1998.
- [szegedy2015conv]** Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.
- [tsumura1986expertSystem]** Hiroshi Tsumura, T Wasano, H Masuzawa, and H Miyake. Patient-oriented multidisciplinary medical consultation system. *MEDINFO*, 86, 1986.
- [tversky1977similarity]** Amos Tversky. Features of similarity. *Psychological review*, 84(4) :327, 1977.
- [vrooman2007knearest]** Henri A Vrooman, Chris A Cocosco, Fedde van der Lijn, Rik Stokking, M Arfan Ikram, Meike W Vernooij, Monique MB Breteler, and Wiro J Niessen. Multi-spectral brain tissue segmentation using automatically trained k-nearest-neighbor classification. *Neuroimage*, 37(1) :71–81, 2007.
- [wang2004similarity]** Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment : from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4) :600–612, 2004.
- [wilson2008medicalImageCbr]** David C Wilson and Dympna O’Sullivan. Medical imagery in case-based reasoning. In *Case-Based Reasoning on Images and Signals*, pages 389–418. Springer, 2008.
- [xia2017wnet]** Xide Xia and Brian Kulis. W-net : A deep model for fully unsupervised image segmentation. *arXiv preprint arXiv :1711.08506*, 2017.

- [xu1992classifiers]** Lei Xu, Adam Krzyzak, and Ching Y Suen. Methods of combining multiple classifiers and their applications to handwriting recognition. *IEEE transactions on systems, man, and cybernetics*, 22(3) :418–435, 1992.
- [zamperoni1995similarity]** Piero Zamperoni and V Starovoitov. How dissimilar are two grey-scale images ? In *Mustererkennung 1995*, pages 448–455. Springer, 1995.
- [zhao2017survey]** Bo Zhao, Jiashi Feng, Xiao Wu, and Shuicheng Yan. A survey on deep learning-based fine-grained object classification and semantic segmentation. *International Journal of Automation and Computing*, pages 1–17, 2017.
- [zheng2018semisupervised]** Qiang Zheng, Yihong Wu, and Yong Fan. Integrating semi-supervised and supervised learning methods for label fusion in multi-atlas based image segmentation. *Frontiers in neuroinformatics*, 12 :69, 2018.
- [zhou2006svm]** J Zhou, KL Chan, VFH Chong, and Shankar M Krishnan. Extraction of brain tumor from mr images using one-class support vector machine. In *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*, pages 6411–6414. IEEE, 2006.

TABLE DES FIGURES

1.1	Schéma du rein et de sa vascularisation [netter2014atlas]	19
1.2	Schéma du corpuscule rénal et de l'appareil juxta-glomérulaire [sherwood2016physiologie]	20
1.3	Schéma anatomique d'un néphron [sherwood2016physiologie]	21
1.4	Vue d'ensemble du système Rénine-Angiotensine-Aldostérone [sherwood2016physiologie]	23
1.5	Exemple d'échantillonnage d'un signal analogique	24
1.6	Illustration de l'échantillonnage d'une image	25
1.7	Exemple d'une segmentation d'images sur une image scanner	27
1.8	Exemple d'une atténuation d'un rayon X	29
1.9	Schéma du fonctionnement d'un scanner	30
1.10	Illustration de l'intérêt d'un fenêtrage	31
1.11	Images scanner en coupe d'un rein et du néphroblastome associées en temps vasculaire (à gauche) et en temps tardif (à droite)	32
1.12	Résultat d'une modélisation 3D (avec le logiciel 3D-Slicer) obtenue à partir des segmentations successives des images scanner (des différentes coupes) : en marron les reins, en vert la tumeur, en rouge l'aorte et les artères rénales et en bleu la veine cave inférieure et les veines rénales	33
1.13	Exemple d'une représentation possible des pixels d'une image 2D en niveaux de gris dans un espace à trois dimensions. L'axe vertical est la valeur du pixel et les deux autres axes sont ses coordonnées spatiales dans l'image	35
1.14	Les différentes catégories d'apprentissage automatique	35
1.15	Proposition de classification des différentes techniques de segmentation et des approches d'intelligence artificielle	39
2.1	Présentation d'un perceptron	44
2.2	Les fonctions d'activation	45
2.3	Perceptron multicouche	45
2.4	Le principe général de l'apprentissage	46
2.5	Principe de rétropropagation du gradient	47
2.6	Exemple d'architecture d'un CNN (<i>Convolution Neural Network</i>)	50
2.7	Le principe de la convolution	52

2.8	La convolution et les neurones artificielles	52
2.9	<i>Fully Convolutional Network</i> [long2015fcn]	54
2.10	Architecture des réseaux FCN-32s, FCN-16s et FCN-8s	54
2.11	Résultats des réseaux FCN-32s, FCN-16s et FCN-8s [long2015fcn]	55
2.12	Architecture du réseau DeconvNet [noh2015deconvNet]	56
2.13	Architecture du réseau DecoupledNet [hong2015decoupledNet]	56
2.14	Architecture du réseau SegNet [badri2015segnet]	57
2.15	Architecture du réseau U-Net [ronneberger2015unet]	58
2.16	Architecture du réseau W-Net [xia2017wnet]	60
2.17	Architecture basique du réseau DeepMedic [kamnitsas2017deepMedic]	61
2.18	Architecture du réseau DeepMedic [kamnitsas2017deepMedic]	62
2.19	Illustration du fonctionnement du réseau DeepLab [chen2018deeplab]	63
2.20	Principe de l' <i>Atrous Spatial Pyramid Pooling</i> (ASPP) [chen2018deeplab]	64
2.21	Architecture du réseau DeepLab [chen2018deeplab]	64
3.1	Le carré d'analogie appliqué au RàPC [lieber2008hdr]	71
3.2	Illustrations de l'interdépendance des 4 conteneurs majeurs dans un système de RàPC, issue de [richter2003fourKnowledge]	73
3.3	Exemple d'utilisation d'un slim-Tree issue de [negny2010effective]	76
3.4	Cycle RàPC issu de [aamodt1994cbr]	79
3.5	Cycle d'un système de RàPC complet	80
3.6	Fonctionnement de l'unité de segmentation du système de RàPC issu de [perner1999segCBR]	83
3.7	Fonctionnement de l'unité de maintenance du système de RàPC issue de [perner1999segCBR]	84
3.8	Fonctionnement d'un algorithme de ligne de partage des eaux (LPE)	88
3.9	Organisation de la base de cas en prototypes	89
4.1	Organisation de la plate-forme COLISEUM-3D	98
4.2	Organisation de la couche données de COLISEUM	99
4.3	Organisation de la couche segmentation de COLISEUM	100
4.4	Organisation de la couche de fusion de COLISEUM	102
4.5	Principe d'un vote majoritaire pour arbitrer un conflit entre plusieurs segmentations	102
4.6	Principe général d'un recalage d'images	103
4.7	Présentation simplifiée des différents types de recalages existants	104

4.8	Utilisation de points de repère anatomiques pour le recalage d'une image scanner (à gauche), avec une image TEP (Tomographie par Émission de Positions, à droite)	105
4.9	<i>Framework</i> du recalage automatique des images scanners en temps vasculaire/tardif	106
5.1	Vue d'ensemble de notre système de RàPC pour la segmentation	113
5.2	Fonctionnement d'un algorithme de croissance de régions	114
5.3	La modélisation d'un cas dans notre système de RàPC : partie problème et partie solution	115
5.4	Illustration de l'application d'un masque flou : (a) CT-scan (b) image après application d'un masque flou	115
5.5	Vue d'ensemble de l'évaluation des pixels voisins pour l'adaptation de la position d'un germe	119
5.6	Exemple d'une exécution de l'algorithme d'adaptation de la position d'un germe	120
5.7	Notions d'écart (<i>gap</i>) et de choix des coupes pour l'entraînement d'un CNN	127
6.1	Comparaison d'une segmentation du rein, en gris intermédiaire, par le système de RàPC avec et sans adaptation de la position des germes : (a) CT-scan (b) segmentation sans adaptation et (c) segmentation avec adaptation de la position	139
6.2	Illustration des effets du post-traitement sur les segmentations calculées : (a) CT-scan (b) segmentation du système de RàPC sans post-traitement (c) segmentation avec post-traitement	144
6.3	Illustrations de l'intérêt de la phase d'adaptation dans notre système de RàPC avec un triplet (20,10,1) : (a) CT-scan (b) vérité terrain (c) segmentation sans adaptation (d) segmentation avec adaptation	148
6.4	Évolution des scores moyens de Dice et IoU en fonction de l'espacement utilisé	152
6.5	Illustration d'une segmentation du patient P8 : (a) CT-scan (b) vérité terrain (c) segmentation par FCN-8s	155
6.6	Illustration d'une segmentation du patient P6 : (a) CT-scan (b) vérité terrain (c) segmentation par FCN-8s	156
6.7	Principe itératif d'une segmentation selon la méthode <i>OV²ASSION-r</i>	164

LISTE DES TABLES

2.1	Caractéristiques des CNN issus de la littérature	65
3.1	Exemple d'une représentation attributs-valeurs	74
5.1	Exemple de V_2 pour $n = 61$	128
6.1	Performance et stabilité des réseaux FCN-8s et U-Net pour la segmentation du néphroblastome	134
6.2	Dice et IoU pour la segmentation du parenchyme rénal et du néphroblastome par RàPC avec croissance de régions (avec adaptation de la position des germes) et CNN	135
6.3	Comparaison des deux métriques de similarité (s_1 et s_2) pour chaque cas durant une <i>validation croisée</i>	137
6.4	Scores de Dice et IoU pour la segmentation du parenchyme rénale avec une croissance de régions guidé par RàPC selon l'existence d'une phase d'adaptation de la position des germes	140
6.5	Erreurs moyennes pour les seuils remémorés : globale et locale	140
6.6	Comparaison de notre système de RàPC avec quelques autres approches pour la segmentation du rein pathologique	142
6.7	Constitution de la base de cas pour l'évaluation de l'adaptation des seuils	143
6.8	Mesures de Dice et IoU moyennes pour la segmentation du parenchyme rénale (pathologique) avec notre système de RàPC amélioré par une adaptation de la position des germes et des seuils	146
6.9	Mesures de Dice et IoU globales pour la segmentation du parenchyme rénale (pathologique) avec notre système de RàPC amélioré par une adaptation de la position des germes et des seuils	147
6.10	Contenu des bases de données pour le RàPC et FCN-8s dans le contexte d'une comparaison des performances	148
6.11	Comparaison des scores de Dice et IoU pour la segmentation du parenchyme rénale (pathologique) entre notre système de RàPC (avec adaptation complète) et FCN-8S entraîné selon la méthode <i>OV²ASSION</i>	149
6.12	Évaluation des segmentations de la tumeur prédites par FCN-8s, entraîné selon la méthode <i>OV²ASSION</i> , en fonction de l'espacement et du vecteur position utilisés	151
6.13	Base de données utilisée pour l'entraînement du réseau FCN-8s selon la méthode <i>OV²ASSION</i>	154

6.14 Évaluation des performances du réseau FCN-8s entraîné selon la méthode <i>OV²ASSION</i> sur une cohorte de 8 patients	154
---	-----

Résumé :

Le néphroblastome (ou tumeur de Wilms) touche le rein et est la tumeur cancéreuse abdominale infantile la plus fréquente. Lors de la thérapie, il est recommandé de préserver au mieux les fonctions rénales en sauvegardant le rein pathologique sous certaines conditions. L'information issue de l'imagerie est importante pour déterminer précisément ces conditions, mais nécessite idéalement une segmentation des images scanners. Cette tâche étant très chronophage, il est courant que cette information ne soit pas exploitée pleinement. L'Intelligence Artificielle (IA) représente une voie prometteuse pour l'automatisation des segmentations mais elle requiert généralement une grande quantité de données d'apprentissage. En particulier, le Raisonnement à Partir de Cas (RàPC) est une approche IA orientée connaissance qui intègre une connaissance métier et une adaptation d'une solution existante pour optimiser la résolution d'un nouveau problème. Les réseaux de neurones convolutifs (CNN) forment quant à eux une approche purement basée sur l'expérience. Nous proposons un système de RàPC couplé à un algorithme de croissance de régions pour réaliser la segmentation de reins déformés par un néphroblastome. En parallèle, une méthode d'entraînement pour des CNN, intitulée *OV²ASSION*, est également proposée pour la segmentation de la tumeur. Les deux approches ont pour but de pallier un manque de données d'apprentissage (peu de segmentations de néphroblastomes et de reins pathologiques sont disponibles). Les évaluations montrent de bonnes performances avec des segmentations calculées proches de celles faites par les radiologues et les chirurgiens.

Mots-clés : Segmentation, Intelligence Artificielle, Néphroblastome, RàPC, Deep Learning

Abstract:

Nephroblastoma (or Wilms' tumor) affects the kidney and is the most common childhood abdominal cancer tumor. During therapy, it is recommended to preserve kidney function as well as possible by preserving the pathological kidney under certain conditions. The information from the imagery is important to validate them, but partly involves segmentation of the scan images. As this task is very time-consuming, it is common for this information not to be fully exploited. Artificial Intelligence (AI) represents a promising way to automate segmentations but generally requires a large amount of learning data. Among other things, Case-Based Reasoning (CBR) is a knowledge-based AI approach that integrates business knowledge and an adaptation of an existing solution to optimize the resolution of a new problem. Convolutional Neural Networks (CNN) are a purely experiential approach. We propose a CBR system coupled with a region growing algorithm in order to segment kidneys deformed by nephroblastoma. In parallel, a training method for CNN, called *OV²ASSION*, is also proposed for tumor segmentation. Both approaches aim to addressing a lack of learning data (few segmentations of nephroblastoma and pathological kidneys are available). The evaluations show good performance with the calculation of segmentations similar to those made by radiologists and surgeons.

Keywords: Segmentation, Artificial Intelligence, Nephroblastoma, CBR, Deep Learning

SPIM

■ École doctorale SPIM 16 route de Gray F - 25030 Besançon cedex

■ tél. +33 (0)3 81 66 66 02 ■ ed-spim@univ-fcomte.fr ■ www.ed-spim.univ-fcomte.fr

