



UNIVERSITÉ DE TOURS

ÉCOLE DOCTORALE MIPTIS

Laboratoire d'Informatique Fondamental
et Appliqué de Tours (EA 6300)

THÈSE présentée par :
Gaëtan GALISOT

pour obtenir le grade de : Docteur de l'université François - Rabelais de Tours

Discipline/ Spécialité : INFORMATIQUE

Segmentation incrémentale et interactive d'images médicales 3D

THÈSE DIRIGÉE PAR :

BROUARD Thierry Maître de Conférences, Université de Tours
RAMEL Jean-Yves Professeur, Université de Tours

RAPPORTEURS :

BLOCH Isabelle Professeur, Université Paris-Saclay
ROUSSEAU François Professeur, Institut Mines Télécom Atlantique

JURY :

BARILLOT Christian Directeur de recherche, IRISA Rennes
BLOCH Isabelle Professeur, Université Paris-Saclay
BROUARD Thierry Maître de Conférences, Université de Tours
JURIE Frédéric Professeur, Université Caen Normandie
RAMEL Jean-Yves Professeur, Université de Tours
ROUSSEAU François Professeur, Institut Mines Télécom Atlantique

Résumé

Cette thèse présente une nouvelle méthode de segmentation interactive et incrémentale d'images médicales 3D. Pour cela, nous proposons une nouvelle manière, plus locale, de modéliser les connaissances *a priori* décrivant les caractéristiques de chacune des structures anatomiques à segmenter. Cette modélisation s'inspire des méthodes se basant sur des atlas pour guider la segmentation mais propose de l'instancier de manière plus locale en permettant l'apprentissage, potentiellement incrémental, d'atlas locaux décrivant chacun uniquement une des structures d'intérêt dans un organe donné. Les atlas sont définis localement sur une zone autour de chaque région d'intérêt. Contrairement au cas classique définissant un seul template (espace de référence) représentant l'organe entier, nos atlas locaux possèdent tous leur propre espace de référence. Les informations provenant de chaque région sont donc séparées (indépendantes), il est aussi possible d'imaginer utiliser des modalités différentes pour apprendre chaque région. Pour lier les structures anatomiques les unes avec les autres, nous proposons d'exploiter un graphe attribué. Les nœuds du graphe correspondent alors aux régions (structures) à segmenter et les arcs représentent les relations spatiales existantes entre ces régions. Ces relations spatiales ont pour objectif de positionner les atlas locaux au sein de l'image entière. Celles-ci sont apprises et stockées par l'intermédiaire des arcs du graphe et sont des informations de distances entre les bords des boîtes englobantes associées à chaque région. Ces distances sont stockées sous forme d'intervalles de distance relative. Elles peuvent être apprises entre chacune des régions du graphe si la base d'apprentissage le permet. Ce *graphe des connaissances a priori* peut être appris à partir de très peu d'images et renferme les connaissances *a priori* associées à chaque région (ou sous-ensemble de régions) de manière répartie et indépendante afin de pouvoir être utilisées facilement et individuellement. Cette modélisation est générique et peut être appliquée à différents organes ou type d'images médicales 3D. Lors de la segmentation, le graphe des connaissances *a priori* est utilisé suivant un processus incrémental et il devient possible de réaliser des segmentations partielles, rapides tout en choisissant l'ordre de segmentation des différentes régions. Outre le choix de l'ordre d'extraction des régions, l'utilisateur peut intervenir avant et après chaque positionnement, segmentation locale afin de corriger le positionnement et la qualité de la segmentation obtenue automatiquement. Pour obtenir les segmentations locales, nous avons proposé une classification des voxels par champ de Markov dans le cas des atlas locaux probabilistes et une méthode de vote classique dans le cas des multi-atlas locaux. Enfin, notre méthode locale englobe un post-traitement de correction des erreurs systématiques qu'une méthode de segmentation peut produire et d'optimisation de l'adjacence entre régions. De nombreuses expérimentations ont permis de mettre en avant les avantages et les inconvénients de notre méthode ainsi que

RÉSUMÉ

de sélectionner les paramètres les plus adéquats. Elles ont également démontré l'efficacité de la méthode proposée par rapport aux méthodes de l'état de l'art en offrant une qualité comparable dans des délais significativement plus courts. Ces travaux ont débouché sur un prototype opérationnel qui a pu être testé lors d'un atelier organisé dans le cadre du projet régional NeuroGeo.

Table des matières

I	Introduction Générale	21
1	Introduction Générale	23
1.1	Contexte	23
1.2	Objectifs vis-à-vis de l'état de l'art et verrous scientifiques étudiés	25
1.3	Contributions	26
1.4	Organisation du manuscrit	27
II	Segmentation d'images médicales 3D	29
2	Les images cérébrales comme cadre d'expérience et d'application privilégié	31
2.1	Anatomie cérébrale	32
2.1.1	Cerveau humain	32
2.1.2	Cerveau de brebis	33
2.2	Imagerie par résonance magnétique	33
3	Modélisation des informations a priori	37
3.1	Les atlas	37
3.2	Recalage d'images	41
3.2.1	Transformations	42
3.2.2	Métriques	43
3.2.3	Optimisations	44
3.3	Modélisation des informations topologiques	44
3.3.1	Relations spatiales	44
3.3.2	Autres informations topologiques	47
3.4	Bilan	48
4	Segmentation guidée	49
4.1	Méthodes interactives	49
4.1.1	Graph-cut	49
4.1.2	Modèles déformables interactifs	51
4.2	Méthodes basées deep learning	54
4.3	Autres méthodes de classification des voxels	54
4.3.1	K-means	55
4.3.2	C-means	56
4.3.3	Mélange de gaussiennes	56
4.3.4	Chaines de Markov cachées	58

TABLE DES MATIÈRES

4.3.5	Champs de Markov Cachés	60
4.4	Fusion de données	61
4.4.1	Méthode de vote	62
4.4.2	Joint Label Fusion	65
4.4.3	Fusion floue	65
4.4.4	Théorie des croyances	66
4.5	Méthodes basées sur des atlas exploitant des informations locales	67
4.6	Correction <i>a posteriori</i>	70
4.7	Bilan	73
III Segmentation incrémentale et interactive d'images médicales 3D		77
5	Modélisation structurelle des données anatomiques	79
5.1	Modélisation locale de la forme	80
5.1.1	Construction des atlas probabilistes locaux	80
5.1.2	Multi-atlas locaux	85
5.1.3	Apprentissage incrémental	85
5.2	Apprentissage et modélisation des informations topologiques	86
5.2.1	Apprentissage des relations spatiales	86
5.2.2	Graphe de connaissances <i>a priori</i>	89
5.3	Complément de l'apprentissage pour une correction <i>a posteriori</i>	90
5.3.1	Exploitation d'informations probabilistes en remplacement des labels de régions	91
5.3.2	Ajout de classifieurs non-région	92
5.4	Bilan	94
6	Exploitation d'informations locales pour la segmentations d'images	95
6.1	Schéma de segmentation	95
6.1.1	Sélection de la région	96
6.1.2	Positionnement de la boîte englobante de la région	96
6.1.3	Classification des voxels	96
6.1.4	Correction <i>a posteriori</i>	97
6.2	Classification des voxels (à partir de l'exploitation des atlas locaux)	97
6.2.1	Initialisation	98
6.2.2	Atlas probabiliste et champs de Markov cachés	99
6.2.3	Multi-Atlas et méthode de vote	101
6.3	Positionnement automatique des régions d'intérêt	102
6.3.1	Sélection des informations disponibles	102

TABLE DES MATIÈRES

6.3.2	Fusion des intervalles	103
6.3.3	Ordonnancement automatique de la segmentation	104
6.4	Détails sur l'étape de post-correction	105
6.4.1	Mode d'utilisation des classifieurs	105
6.4.2	Fusion des cartes de probabilité	105
6.4.3	Correction de l'adjacence	106
6.5	Bilan	110
IV	Intérêts de l'approche proposée et applications	111
7	Évaluation des performances et validation	113
7.1	Données et outils utilisés pour les expériences	113
7.1.1	Bases d'images	113
7.1.2	Validation des segmentations	118
7.2	Robustesse de la méthode vis à vis de ses paramètres	119
7.2.1	Robustesse de la méthode	119
7.2.2	Sélection du mode d'exploitation des relations spatiales	122
7.2.3	Impact de l'étape de post-correction	124
7.3	Sensibilité et généralité de la méthode proposée	128
7.3.1	Influence de la base d'apprentissage	129
7.3.2	Sensibilité vis à vis du séquençement et des positionnements	132
7.3.3	Intérêt de l'interactivité	142
7.4	Performances de la méthode proposée par rapport à l'état de l'art	145
7.4.1	Étude de l'influence du mode d'apprentissage	145
7.4.2	Comparaison en terme de qualité des résultats	150
7.4.3	Comparaison en terme de temps de calcul	155
8	Le prototype 3DBrainSeg et son utilisation	159
8.1	3DBrainSeg	159
8.1.1	Implémentation	159
8.1.2	Fonctionnalités	160
8.1.3	Paramétrage des méthodes	161
8.1.4	Interface graphique	163
8.2	Atelier « Utilisateurs » NeuroGeo et retours d'expériences	163
8.2.1	Organisation de l'atelier	164
8.2.2	Retours d'expériences	166
8.2.3	Bilan	168

TABLE DES MATIÈRES

V Conclusion	169
9 Conclusion	171
9.1 Bilan	171
9.2 Perspectives	174
Index	186

TABLE DES MATIÈRES

Liste des tableaux

4.1	Qualités et défauts des différentes catégories de méthodes de segmentation d'images médicales.	75
7.1	Liste des 13 régions segmentées lors des expériences sur les images de cerveaux humains	114
7.2	Liste des régions segmentées lors de l'atelier utilisateur du projet NeuroGeo sur les images de cerveau de brebis T2 ex vivo	116
7.3	Liste des régions segmentées lors des expériences sur le cœur humain	117
7.4	Écart moyen exprimé en pixels entre les bords de la vérité-terrain et le positionnement proposé pour différents types de distances.	123
7.5	Écart moyen en pixels entre les bords de la vérité-terrain et le positionnement proposé pour différents types de fusion de données.	124
7.6	Indices Dice moyens de la segmentation avec les différentes méthodes de corrections appliquées.	127
7.7	Rang moyen des régions dans l'ordre de segmentation. Les colonnes représentent quelle région a été positionnée et segmentée en premier. Les lignes représentent à quelle position cette région à été segmentée en moyenne sur les 20 images de test.	135
7.8	Dice Ratio pour les différentes expériences et structures du cœur	155

LISTE DES TABLEAUX

Table des figures

2.1	L'image (a) illustre l'encéphale humain composé du cerveau, du cervelet et du tronc cérébral. L'image (b) est un schéma d'une coupe transversale du cerveau humain et des structures anatomiques centrales (les images sont tirées de [3, 4])	32
2.2	Anatomie du cerveau de la brebis (l'image est tirée de [5]).	34
2.3	L'image (a) illustre une image IRM pondéré T1 et une image IRM pondéré T2. L'image (b) illustre un exemple d'image IRM fonctionnelle (les images sont tirées de [2, 6])	36
3.2	Atlas probabiliste du cerveau de la brebis composé d'une image template IRM T1 et de 3 cartes de probabilité de présence de la région (LCF, MB, MG). La probabilité d'appartenance des voxels aux régions est reliée à l'échelle d'intensité située à gauche (l'image est tirée de [33]).	39
3.1	Atlas ICBM du cerveau humain avec (a) l'image IRM T1, (b) la carte de probabilité de la matière blanche, (c) la carte de probabilité de présence de la matière grise et (d) la carte de probabilité de présence du liquide céphalo-rachidien [64]. Plus l'intensité est claire pour la probabilité d'appartenance est forte.	40
3.3	Illustration d'un atlas surfacique des régions corticales sur une version dilatée du cortex cérébral. Les régions proviennent de l'atlas de cerveau humain du template FreeSurfer "fsaverage" de l'hémisphère gauche. (l'image est tirée de [54])	41
3.4	Exemple d'une relation spatiale de type "distance". La figure (a) représente la région de référence, (b) la carte de distance, (c) représente la fonction de d'appartenance pour la relation " <i>distance inférieur à D</i> " et (d) l'ensemble flou correspondant. (e) représente la fonction d'appartenance pour la relation " <i>à une distance D</i> " et (f) l'ensemble flou correspondant. (g) représente la fonction d'appartenance pour la relation " <i>à une distance supérieur de D</i> " et (h) l'ensemble flou correspondant. Les zones d'intensité élevée correspondant à une probabilité élevée et les zones d'intensité faible à une probabilité faible. (l'image est tirée de [28])	46

TABLE DES FIGURES

3.5	Illustration des profondeurs r_{12} et r_{21} liées aux sillons S_1 et S_2 respectivement, qui sont utilisées pour définir l'attribut h_a présent dans l'arc entre le sillon S_1 et S_2	48
4.1	Exemple de graph-cut sur une image 3x3. (a) Deux graines ont été positionné sur l'image (objet O et fond B). (b) Le graphe correspondant est construit puis (c) la coupe de coût minimale est détectée fournissant (d) une segmentation binaire de l'image (l'image est tirée de[104]).	51
4.2	Les contours orthogonaux (a) (rouge et bleu) intersectent les coupes dans la troisième dimension (vert). Les points d'intersection sur la troisième dimension sont utilisés comme graines pour obtenir la segmentation en 3 dimensions (l'image est tirée de [74]).	52
4.3	Exemple de surface dragging sur une surface sphérique. (a) Représente le point de contrôle avant modification de la forme. (b) Représente la forme de la sphère après déplacement du curseur (l'image est tirée de [51]).	53
4.4	Architecture U-net dans le cas 2D pour une image d'entrée 572x572. L'image d'entrée est l'image à segmenter et l'image de sortie est l'image label (<i>end-to-end</i>). Le schéma illustre les différentes convolutions et les différents pooling appliqués à l'image (l'image est tirée de [78]).	55
4.5	Schéma du modèle génératif. I_n est l'image template de l'atlas n° n , L_n est l'image label correspondante. I est l'image à segmenter et L est l'image label que l'on cherche à déterminer. M est un champ aléatoire qui associe à chaque voxel de l'image L , l'index de l'atlas à partir duquel il a été généré. σ est un paramètre qui module l'importance de la ressemblance entre le template T_n et l'image I . β est un paramètre qui module le choix du champs M (variation entre une pondération totalement globale et totalement locale). ρ est un paramètre qui modélise l'importance de la distance entre les labels des atlas et de l'image à segmenter I (l'image est tirée de[83]).	64
4.6	Creation du multi region multi reference atlas. La figure (a) montre un exemple de 4 clusters utilisés pour une des région du cerveau. La figure (b) montre un exemple des 4 templates et cartes de probabilités pour un région donnée (l'image est tirée de [88]).	68
4.7	(a) Division de l'image en N sous-volumes cubiques, chacun segmenté par champ de Markov (en tissus). La figure (b) représente les interactions entre les différents champs de Markov (tissus et structures) (l'image est tirée de [84]).	70
4.8	Schéma de la construction et l'utilisation de la post-correction [99]	72
5.1	Schéma de la construction des différents atlas probabilistes locaux à partir d'une base d'images d'apprentissage. Exemple d'application sur une base d'images IRM T1 de cerveaux, la région 1 représente un Noyau caudé et la région M un Putamen.	82

TABLE DES FIGURES

5.2	Construction des points de contrôle de l'histogramme de référence. e_r^{ref} et s_r^{ref} sont fixés par l'utilisateur tandis que les q_r^{ref} sont calculés comme la moyenne des quartiles des histogrammes disponibles.	83
5.3	Illustration dans le plan $(0, \vec{x}, \vec{y})$ des relations spatiales de la région r_1 vers la région r_2 (8 distances dans le cas 2D). Quatre distances dans chaque direction font ici un total de 8 directions.	89
5.4	Exemple de graphe <i>a priori</i> dans le cas de 3 structures sous-corticales du cerveau humain (Putamen, Noyau caudé et Thalamus).	90
5.5	Illustration des caractéristiques utilisées pour définir les classifieurs Adaboost (dans le cas 2D). La région r est illustrée ici est le seuillage de la carte de probabilité provenant de la segmentation à corriger. (a) illustre un exemple pour un voxel v et une région r . La grille représente le voisinage autour du voxel v permettant de calculer les caractéristiques. (b) représente les valeurs de probabilité provenant du HMF sur le patch autour du voxel v . (c) représente les valeurs d'image réelle sur le patch autour du voxel v . Les données illustrées en rouge font référence aux données décrites dans l'équation 5.5.	93
6.1	Schéma représentant les différentes étapes nécessaires à la segmentation d'une région r	98
6.2	Étapes de correction <i>a posteriori</i>	108
6.3	Exemple de résultat de correction sur une partie d'image de 6x6 pixels. Deux régions R_1 et R_2 sont segmentées l'une après l'autre. Les processus $Seg.R_1$ et $Seg.R_2$ représentent les segmentations avec atlas locaux. Les processus $Correction$ représentent l'application des classifieurs Adaboost (avec et sans correction du problème d'adjacence).	109
7.1	Coupe coronale d'une image (a) de la base d'apprentissage et de l'image (b) labellisée correspondante	114
7.2	Coupe transversale d'une image IRM et des labels de structures anatomiques du cerveau humain (10 des 13 structures sont représentées)	115
7.3	Structures anatomiques du cerveau de brebis qui sont étudiées et segmentées dans le cadre des expériences sur les images IRM de cerveau de brebis (Image <i>B1</i>)	116
7.4	Image (a) scanner de cœur et image (b) IRM du cœur provenant de la base MM-WHS'17	117
7.5	Structures anatomiques du cœur liées aux images de la base CT scan du challenge MM-WHS'17	118
7.6	Schéma et liste des différents paramètres liés à la méthode de segmentation proposée.	119
7.7	Résultat du plan d'expérience de Tagushi pour observer l'impact des paramètres du champ de Markov et du recalage.	121

TABLE DES FIGURES

7.8	Résultat de segmentation de l'image (1003 de la base MICCAI'12). (7.8a) vérité terrain, (7.8b) LIS sans correction, (7.8c) LIS avec correction classique, (7.8c) LIS avec correction utilisant les probabilité du HMF, (7.8d) LIS avec correction utilisant les probabilités du HMF et des classifieurs <i>non-région</i> , (7.8f) LIS avec correction utilisant les probabilités du HMF, des classifieurs <i>non-région</i> et la correction de l'adjacence.	126
7.9	Distribution du ratio Dice pour chaque méthode pour les régions sous-corticales, la matière blanche et les régions corticales	128
7.10	Évolution de la qualité de la segmentation en fonction du nombre d'images présentes dans la base d'apprentissage. La moyenne est représentée par l'étoile bleu.	130
7.11	Évolution de la qualité de la segmentation en fonction du nombre d'images de la base d'apprentissage. La moyenne est représentée par l'étoile bleu.	131
7.12	Coupe transversal d'une image IRM T1 d'un cerveau de brebis in vivo (acquisition sur la plateforme CIRE).	132
7.13	Segmentation guidée par un utilisateur (7.13a) et guidée par les relations spatiales (7.13b) d'une image T1 de brebis in vivo	133
7.14	Qualité des segmentations en fonction de différents types d'ordre. Les relations spatiales sont utilisées et seule la première région est bien positionnée. Pour chaque type d'ordre, la plus mauvaise segmentation, le cas médian et la meilleure sont exposés.	136
7.15	Qualité de la segmentation en fonction du nombre de régions positionnées correctement (vérité-terrain), les autres régions étant placées automatiquement (relations spatiales).	137
7.16	Qualité de la segmentation par région en fonction du nombre de régions correctement positionnées (vérité-terrain), les autres régions sont placées automatiquement (relations spatiales). Les régions en abscisse sont ordonnées selon l'ordre de segmentation appliqué.	138
7.17	Erreur de positionnement en fonction des régions du cerveau humain, positionnement manuel (rouge) et automatique (bleu).	139
7.18	Erreur de positionnement en fonction des régions du cerveau de brebis, positionnement manuel (rouge) et automatique (bleu).	140
7.19	Qualité de la segmentation des régions lors d'une expérience utilisateurs. L'indice Dice est représenté en fonction de l'erreur en voxels par bord lors du positionnement.	141
7.20	(7.20a) Vérité-terrain du bulbe olfactif, (7.20b) segmentation avec la boîte englobante positionnée avec la vérité-terrain, (7.20c) segmentation avec la boîte englobante positionnée manuellement. (7.20d) Vérité-terrain de l'amygdale, (7.20e) segmentation avec la boîte englobante positionnée avec la vérité-terrain, (7.20f) segmentation avec la boîte englobante positionnée manuellement.	143

TABLE DES FIGURES

7.21	Qualité de la segmentation (ratio Dice) d'une image de brebis (<i>B2</i>) segmentée par différents utilisateurs comparée à un positionnement selon la vérité-terrain.	145
7.22	Qualité des segmentations obtenues à l'aide des multi-atlas locaux et de multi-atlas globaux. Les diagrammes de boites représentent les résultats provenant des 20 images de test disponibles pour chaque type de région.	147
7.23	Qualité des segmentations obtenues à l'aide des atlas probabilistes locaux et des multi-atlas locaux lorsque les boites englobantes sont positionnées avec la vérité-terrain. Les diagrammes de boites représentent les résultats provenant des 20 images de test disponibles pour chaque type de région.	149
7.24	Qualité des segmentations obtenues à l'aide des atlas probabilistes locaux et des multi-atlas locaux lorsque les boites englobantes sont positionnées avec les relations spatiales (sauf première région). Les diagrammes de boites représentent les résultats provenant des 20 images de test disponibles pour chaque type de région.	149
7.25	Qualité de la segmentation des images de cerveaux humains de notre méthode LIS (régions bien placées), de la segmentation multi-atlas JLF et de Freesurfer. Les diagrammes de boites représentent les résultats provenant des 20 images de test disponibles pour chaque type de région.	151
7.26	Dice moyen de chaque méthode sur la base MICCAI'12	152
7.27	Qualité de la segmentation des images brebis ex vivo de notre méthode et de la segmentation multi-atlas JLF. Les diagramme de boites représentent les résultats provenant des 6 images disponibles.	152
7.28	Résultat de segmentation du cas médian de l'expérience (E1) sur les images CT scan et sur les images IRM.	155
7.29	Temps de calcul mesurés sur des machines différentes nécessaire pour obtenir la segmentation de 13 régions anatomiques. Le temps de la méthode JLF comprend le temps de recalage des images ainsi que le temps de calcul de segmentation a proprement dit. Le temps de calcul de la notre méthode correspond au temps nécessaire à l'enchaînement des 13 segmentations sans tenir compte des temps d'interaction.	156
7.30	Évolution du temps moyen nécessaire à la segmentation des 13 régions des images de cerveaux de brebis ex-vivo en fonction du nombre d'image déjà segmenté par l'utilisateur.	158
8.1	Diagramme d'activité du logiciel 3DBrainSeg. En plus ces interactions, l'utilisateur peut enregistrer ou charger une segmentation.	162
8.2	Interface graphique du logiciel 3DBrainSeg. Exemple avec une image de cerveau de brebis de la base NeuroGeoEx et 3 régions segmentées.	164
8.3	Visualisation du positionnement d'une boite englobante (Noyau caudé d'une image de cerveau de brebis de la base NeuroGeoEx).	165

TABLE DES FIGURES

Notations

Notations liées aux atlas locaux :

I^n	$n^{\text{ème}}$ image de la base d'apprentissage
L^n	$n^{\text{ème}}$ image label de la base d'apprentissage
R	Ensemble des régions apprises présentes dans le graphe de connaissances <i>a priori</i>
N_r	Nombre de couples d'images avec images labels associées pour la construction des informations de la région r
R_n	Ensemble des régions labellisées dans l'image L_n
I_r^n	Sous-image de la région r dans la $n^{\text{ème}}$ image de la base d'apprentissage
L_r^n	Sous-image de la région r dans la $n^{\text{ème}}$ image label de la base d'apprentissage
A_r	Atlas de la région r composée du couple $\{T_r, P_r\}$
T_r	Image template de la région r
P_r	Carte de probabilité de la région r

Notations liées aux relations spatiales :

dim_x	Dimension de la région r dans la direction \vec{x}
$d_{r_1 r_2}^{xij}$	Distance dans la direction \vec{x} entre le bord i de la région r_1 et le bord j de la région r_2
$dr_{r_1 r_2}^{xij}$	Distance relative dans la direction \vec{x} entre le bord i de la région r_1 et le bord j de la région r_2
$Max_{dr_{r_1 r_2}^{xij}}$	Distance relative maximum parmi les $dr_{r_1 r_2}^{xij}$ de la base d'apprentissage
$Min_{dr_{r_1 r_2}^{xij}}$	Distance relative minimum parmi les $dr_{r_1 r_2}^{xij}$ de la base d'apprentissage
$Xmin_{r'}^{ij}$	Position minimum dans l'image cible du bord j provenant du bord i de la région r'
$Xmax_{r'}^{ij}$	Position maximum dans l'image cible du bord j provenant du bord i de la région r'
B_r^j	Position du bord j de la région r dans l'image à segmenter

Notations des résultats et des classifieurs :

HMF	Champ de Markov
Y	Image cible à segmenter
Y_r	Sous-image à l'intérieur de la boîte englobante de la région r positionné dans l'image Y
Z_r	Sous-image segmentée après l'étape de classification par champ de Markov
S	Image segmentée "finale" (après la segmentation d'une nouvelle région).
α_i	Coefficient d'importance de l'atlas dans le champ de Markov du voxel v_i
β	Coefficient d'importance du voisinages dans le champ de Markov
P_r^{HMF}	Carte de probabilité d'appartenance à la région r à la sortie du champ de Markov
C_r	Classifieur Adaboost lié à la région r
$C_{\bar{r}}$	Classifieur Adaboost lié à l'extérieur de la région r
P_r^{corr}	Carte de probabilité d'appartenance à la région r à la sortie de l'étape de correction
$P_{\bar{r}}^{corr}$	Carte de probabilité d'appartenance à l'extérieur de la région r à la sortie de l'étape de correction

Première partie

Introduction Générale

Chapitre 1

Introduction Générale

1.1 Contexte

Le développement des techniques d'imagerie (IRM, scanner, ultrason, etc.) a ouvert de nouvelles problématiques de traitement d'images et en particulier de segmentation d'images médicales. La quantité mais aussi la qualité des données disponibles sont de plus en plus élevées, les capacités de calcul évoluent également très rapidement laissant apparaître de nouveaux besoins et de nouvelles opportunités vis-à-vis des méthodes de traitement et d'analyse envisageables. De nouveaux outils permettant d'aider, voire de remplacer, les biologistes ou les médecins apparaissent régulièrement. Citons simplement les récentes expériences concernant la comparaison de diagnostics faits, à partir d'images de pathologies, par des médecins et par des algorithmes à base de réseaux de neurones profonds montrant la supériorité des méthodes automatiques sur les taux de reconnaissance finalement obtenus.

La majeure partie des outils et des méthodes de segmentation d'images médicales développée aujourd'hui est construite pour être exploitée dans un contexte bien spécifique comme pour le cas d'images médicales d'un organe spécifique, pour une espèce donnée (humains, rats, moutons, etc.) et avec éventuellement aussi une pathologie particulière. La détection de maladies (Alzheimer, sclérose en plaques, etc.) par le biais de la segmentation de certaines zones anatomiques caractéristiques est un cas classique d'utilisation. Dans le but de devenir des routines cliniques, ces outils sont très souvent automatiques : l'utilisateur n'intervient pas ou très peu dans le processus qui est donc, en contrepartie, très spécifique à un problème particulier.

Pourtant, il existe des cadres d'applications variés de segmentation d'images anatomiques 3D. Bien que l'Humain reste encore très fortement privilégié, l'imagerie médicale animale mérite aussi d'être soutenue en particulier pour des applications inenvisageables sur l'Humain mais qui permettent de faire des avancées et des découvertes fondamentales sur les fonctionnements biologiques. Le rongeur est beaucoup utilisé dans le cadre des expérimentations animales. Des images gros animaux du type brebis sont aussi acquises afin d'être étudiées. Comme pour n'importe quel type d'images médicales, la labellisation manuelle ou l'analyse de ces images est un travail fastidieux pour lequel il serait intéressant de disposer d'outils facilitant le travail des biologistes. Les nombreux outils existants, dédiés au cerveau humain, ne sont pas directement utilisables dans le cas de l'imagerie animale.

L'anatomie du cerveau animal, comme celui de la brebis, semble trop différente de l'anatomie humaine pour utiliser les informations apprises sur des bases d'images de cerveaux humains. Les méthodes de segmentation associées ne sont, de plus, pas assez souples pour permettre de gérer les images d'autres espèces. En outre, la quantité de données disponibles (en particulier labellisées) pour les images animales est beaucoup plus faible ce qui rend encore plus difficile l'apprentissage des connaissances *a priori* nécessaires à l'exploitation de certaines méthodes (réseaux profonds, segmentation multi-atlas). Il paraît cependant difficile, ou en tout cas loin d'être optimal, d'envisager de concevoir et de produire des nouveaux outils spécifiques à chaque fois qu'une nouvelle espèce ou de nouvelles structures anatomiques sont étudiées.

Le projet NeuroGeo est un projet financé par la région Centre-Val-de-Loire dont l'un des objectifs est de répondre à cette problématique en essayant de concevoir un outil de segmentation d'images médicales 3D capable de supporter plus de variabilités dans les données qu'il pourrait analyser (images cérébrales provenant de différentes populations ou espèces, avec ou sans pathologie, mono ou multimodales). Ce projet rassemble des biologistes de l'INRA (Institut National de la Recherche Agronomique), des chercheurs en neuro-anatomie de l'INSERM (Institut National de la Santé et de la Recherche Médicale) et enfin des chercheurs en informatique du Laboratoire d'Informatique Fondamentale et Appliqué de Tours (LIFAT). Les travaux réalisés durant cette thèse rentrent dans le cadre de ce projet et ont pour but de développer des méthodes de segmentation interactives d'images médicales 3D pouvant répondre à ces exigences de généralité.

Le projet NeuroGeo a donc permis de fixer un cadre applicatif pour ces travaux de thèse ; l'objectif principal étant de répondre aux besoins des biologistes de l'INRA. Ces biologistes cherchent à obtenir des segmentations d'images de cerveaux de brebis pour les raisons suivantes :

- La détection et la compréhension des liaisons neuro-anatomiques entre les structures nécessitent par exemple la segmentation préalable de ces mêmes structures.
- La création d'atlas encore peu existants ou incomplets pour les espèces animales étudiées. Une définition du cortex cérébral du cerveau de brebis est par exemple en cours de réalisation.
- L'étude de l'évolution temporelle de certaines structures du cerveau de brebis au cours de la croissance.

Ces différents objectifs applicatifs, nous ont permis de définir plusieurs caractéristiques pour la méthode de segmentation visée :

- Bien que principalement dédiée aux cerveaux de brebis, la méthode proposée devra être générique ou facilement adaptable afin de pouvoir segmenter des images 3D très diverses provenant d'images de différentes modalités d'acquisition. Elle doit permettre la segmentation d'images provenant du cerveau d'autres espèces (animales ou humaines) mais aussi éventuellement d'autres organes anatomiques.
- La méthode de segmentation devra nécessiter peu de données étiquetées (base d'apprentissage de très petite taille) compte-tenu du faible nombre de données disponibles.
- Une autre contrainte forte est la volonté de pouvoir n'effectuer qu'une segmentation partielle des structures anatomiques présentes dans l'organe à étudier. Il s'agit ici de

1.2. OBJECTIFS VIS-À-VIS DE L'ÉTAT DE L'ART ET VERROUS SCIENTIFIQUES ÉTUDIÉS

pouvoir réduire les temps de calcul et aussi d'envisager une approche incrémentale permettant d'accumuler petit à petit des connaissances *a priori* sur les données à traiter qui pourront être réinjectées dans le système afin qu'il devienne plus performant ou complet au fil du temps (apprentissage incrémental)

- La méthode de segmentation a pour objectif final d'être manipulée par les biologistes (voire même des collégiens) et non des informaticiens. Les interactions avec le logiciel doivent donc être intuitives et ne pas nécessiter de connaissances des algorithmes de segmentation sous-jacents.

1.2 Objectifs vis-à-vis de l'état de l'art et verrous scientifiques étudiés

Les méthodes de segmentation d'images médicales existantes ne répondent pas à tous ces critères. Comme nous le verrons au travers du panorama des méthodes de segmentation d'images médicales que nous avons pu dresser dans l'état de l'art, les outils permettant la segmentation d'un large éventail de régions ou d'organes sont en général coûteux en terme de temps d'exécution. Ils effectuent généralement des traitements sur l'image entière (à part peut être ceux visant à segmenter des tumeurs ou une structure particulière uniquement).

Les méthodes de segmentation basées sur des atlas ou basées sur de l'apprentissage automatique peuvent nécessiter une base d'apprentissage de taille importante. Ces méthodes deviennent performantes lorsque cette base d'apprentissage est capable de représenter une population entière, ce qui n'est pas toujours facile à obtenir. Les méthodes basées sur des atlas sont, de plus, généralement des approches globales et nécessitent par exemple un recalage de toute l'image ce qui peut être très coûteux en temps si l'on veut un recalage fin.

Enfin, peu de méthodes de segmentation interactives ou incrémentales ont été proposées jusqu'à présent. Les algorithmes ne peuvent ainsi pas être guidés ou corrigés par un expert durant leur exécution. Si le résultat de la segmentation n'est pas correct, l'utilisateur n'a pas d'autres alternatives que la correction manuelle. Dans le meilleur des cas, il peut changer certains paramètres et recommencer le processus de segmentation entièrement (potentiellement long). Certaines méthodes de segmentation interactives ont bien été développées mais nécessitent un nombre important d'interactions. Les méthodes de type ITK-Snap [106] nécessitent un temps utilisateur non négligeable pour la segmentation de chaque région. De plus, leur utilisation n'est pas toujours intuitive et facilement appréhendable pour des non-experts en traitement d'images.

Afin d'apporter des contributions à ces problématiques, nos travaux se sont focalisés sur les verrous scientifiques suivants :

- Comment apprendre automatiquement et modéliser efficacement, à partir de peu de données labellisées, les connaissances *a priori* indispensables à la segmentation d'images médicales 3D ?
- Comment permettre une alimentation et une amélioration continue (incrémentale) de ces connaissances *a priori* aussi bien dans un objectif de spécialisation à une population donnée qu'au contraire, pour une diversification vers d'autres populations

(selon le choix de l'utilisateur) ?

- Comment exploiter, durant la segmentation, qu'une sous-partie des connaissances *a priori* disponibles afin de réduire les temps de calcul (gain d'efficacité) en laissant l'utilisateur se focaliser uniquement sur les structures anatomiques qui l'intéresse ?
- Comment modifier les méthodes de segmentation afin de permettre à l'utilisateur de guider, d'aider efficacement les algorithmes afin de les rendre plus génériques tout en restant efficaces (recherche d'un compromis optimal entre plasticité et efficacité des méthodes) ?

1.3 Contributions

Afin de répondre aux deux premiers questionnements mentionnés ci-dessus, nous proposons une nouvelle manière, plus locale, de modéliser les connaissances *a priori* décrivant les caractéristiques anatomiques de chacune des structures des organes à segmenter. Cette modélisation s'inspire des méthodes se basant sur des atlas pour guider la segmentation mais propose de l'instancier de manière plus locale en permettant la création ou l'apprentissage incrémental d'atlas locaux décrivant chacun uniquement une des structures d'intérêt dans un organe donné. Il devient alors nécessaire d'associer des informations de relations spatiales (topologiques) à ces différents atlas locaux pour obtenir une modélisation complète de l'organe ou des sous-parties de l'organe que l'on désire représenter (analyser). Nous proposons pour cela d'exploiter un graphe attribué pour lier les structures anatomiques (représentées par des atlas locaux) les unes avec les autres. Les nœuds du graphe correspondent alors aux régions (structures anatomiques) à segmenter et les arcs représentent les relations spatiales existantes entre ces régions. Les atlas locaux seront stockés comme attributs associés aux nœuds du graphe. Les relations spatiales peuvent également être apprises (informations de distances relatives entre régions) et stockées comme attributs associés aux arcs du graphe. Le graphe (que nous appellerons graphe des connaissances *a priori*) peut être appris à partir de très peu d'images et représente la scène à analyser. Il renferme les connaissances *a priori* associées à chaque région (ou sous-ensemble de régions) de manière indépendante les unes par rapport aux autres afin de pouvoir être utilisées facilement individuellement. Cette modélisation n'est pas spécifiquement dédiée aux cerveaux de brebis mais reste générale et applicable à différents organes ou type d'images médicales 3D.

Nous présentons ensuite un mode d'utilisation de cette modélisation des connaissances *a priori* représentées sous forme d'un graphe. Nous proposons une nouvelle méthode de segmentation fonctionnant de manière incrémentale et interactive. Pour cela, il nous a fallu apporter des adaptations significatives aux méthodes de segmentation globale nous ayant servi de sources d'inspiration : dans notre chaîne de traitement, chaque région est segmentée après positionnement d'un atlas défini localement en appliquant une classification de voxels à l'aide d'un champ de Markov permettant de tenir compte des connaissances *a priori* et des informations provenant de l'image à segmenter. Le positionnement de chacun des atlas locaux peut être automatisé à l'aide des informations de relations spatiales apprises et stockées dans le graphe. La segmentation est alors réalisée suivant un processus incrémental et il devient possible de réaliser des segmentations partielles, rapides en faisant varier ou

en choisissant l'ordre de segmentation des différentes régions. Outre le choix de l'ordre d'extraction des régions, l'utilisateur peut intervenir avant et après chaque positionnement, segmentation locale afin de corriger, de manière interactive, le positionnement et la qualité de la segmentation obtenue automatiquement.

Enfin, notre méthode locale englobe un post-traitement de correction des erreurs systématiques qu'une méthode de segmentation peut produire et d'optimisation de l'adjacence entre régions. Ce post-traitement a été adapté à notre approche de manière à gérer le caractère incrémentale la segmentation.

1.4 Organisation du manuscrit

Le manuscrit est organisé en 3 parties principales :

- La partie 1 présente le cadre dans lequel se situe ces travaux de thèse c'est-à-dire la segmentation d'images 3D médicales. Elle comporte les chapitres suivants :
 - Le chapitre 1 présente le cadre d'application principal de la thèse. Les caractéristiques et les difficultés liées à l'anatomie cérébrale et aux images IRM y sont présentées.
 - Le chapitre 2 correspond à l'état de l'art ; il présente les différentes méthodes de segmentation d'images médicales existantes. Les méthodes de segmentation guidées par l'utilisateur ou par des informations *a priori* y sont particulièrement abordées.
- La partie 2 décrit le cœur du travail réalisé et présente le détail des différentes contributions de la thèse :
 - Le chapitre 3 décrit la manière dont nous proposons de modéliser les connaissances *a priori* nécessaires à la segmentation incrémentale d'une scène. Le mode d'apprentissage des atlas locaux, des relations spatiales ainsi que des classificateurs nécessaires durant l'étape de post-correction sont présentés dans ce chapitre.
 - Le chapitre 4 se focalise sur le processus de segmentation à proprement parlé. Il décrit les méthodes de segmentation que nous avons mis en place avec leur mode d'exploitation des informations *a priori* préalablement apprises et stockées dans le graphe des connaissances *a priori*.
- La partie 3 présente le résultat des nombreuses expérimentations que nous avons menées durant cette thèse et qui ont permis de valider l'outil de segmentation 3DBrain-Seg finalement fourni aux biologistes.
 - Le chapitre 5 présente les bases d'images utilisées ainsi que le plan d'expériences appliqué afin de sélectionner les paramètres de la méthode et de caractériser ses différentes propriétés. Une comparaison des résultats de notre méthode avec ceux obtenus par plusieurs méthodes de l'état de l'art est aussi présentée dans ce chapitre
 - Le chapitre 6 présente l'outil de segmentation qui découle de la méthode proposée dans ces travaux. Les retours obtenus lors d'un atelier « utilisateurs » réalisé dans le cadre du projet NeuroGeo sont aussi décrits dans ce chapitre.

1.4. ORGANISATION DU MANUSCRIT

Enfin, une conclusion générale et des perspectives synthétisent l'ensemble des travaux réalisés, les résultats obtenus et les voies de recherche ouvertes durant ces trois années.

Deuxième partie

Segmentation d'images médicales 3D

Chapitre 2

Les images cérébrales comme cadre d'expérience et d'application privilégié

L'imagerie cérébrale est un ensemble de techniques apparues à partir du début du XX^e siècle, permettant d'observer le cerveau dans des conditions *in vivo*. Plusieurs types d'imagerie cérébrale ont été développés afin de produire de nouvelles informations. On peut notamment citer l'imagerie par rayon X et le scanner [45], l'imagerie par tomographie par émission de positron (TEP) [92] ou l'image par résonance magnétique (IRM) [30]. Ces différentes technologies ont permis d'apprendre de nouvelles connaissances sur la structure et le fonctionnement du cerveau humain. Elles ont ensuite été appliqués sur le cerveau animal, en particulier sur les cerveaux de petits animaux (de type rongeur comme la souris) lors d'expérimentations. Les cerveaux d'animaux de taille plus importante sont aussi étudiés. Leurs propriétés neuro-anatomiques sont plus proches de celles de l'Homme. De plus, le fonctionnement neurologique de certaines structures anatomiques intéresse les biologistes afin d'améliorer les conditions d'élevage des animaux. C'est dans ce contexte que les neurobiologistes de l'INRA de Nouzilly étudient ce type de cerveau et en particulier celui de la brebis. L'ensemble de ce travail de thèse se déroule en parallèle du projet NeuroGeo. Ce dernier vise à développer des outils de segmentation interactifs d'images IRM de cerveaux de brebis. Certains outils existent déjà pour le cerveau humain, cependant ils ne sont pas directement utilisables sur le cerveau d'autres espèces. L'anatomie du cerveau de brebis est différente de celle du cerveau humain rendant les informations *a priori* existantes sur le cerveau humain inexploitable dans le cas du cerveau de brebis.

Les travaux présentés dans ce manuscrit décrivent des méthodes et outils de segmentations pouvant s'appliquer à différents types d'images médicales. Néanmoins, il nous a semblé important de repréciser certains points du cadre applicatif (image IRM de cerveaux de brebis) ayant guidés nos réflexions. Nous allons présenter rapidement l'anatomie du cerveau humain ainsi que du cerveau de la brebis. Ensuite, nous présenterons rapidement les bases du fonctionnement de l'imagerie IRM et ses caractéristiques.

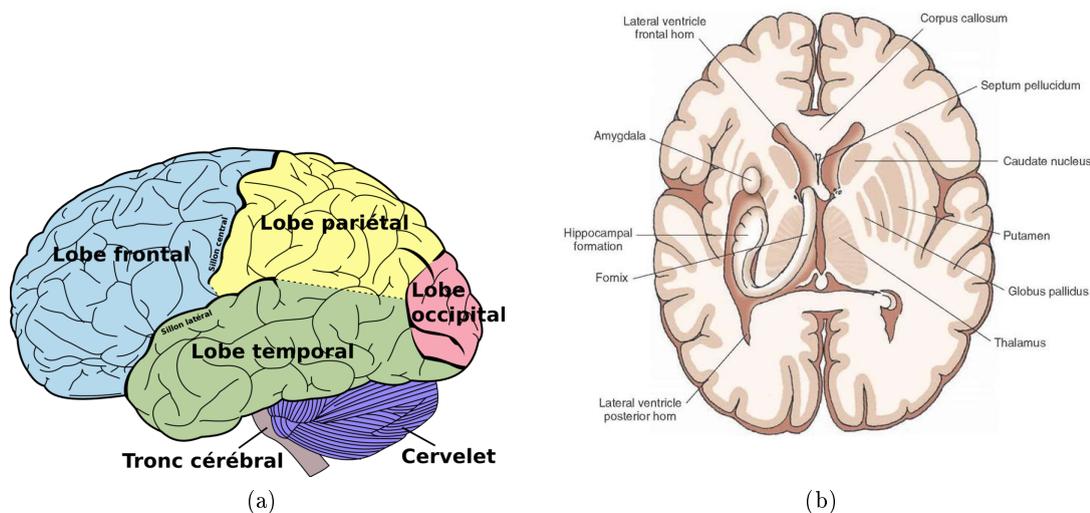


FIGURE 2.1 – L'image (a) illustre l'encéphale humain composé du cerveau, du cervelet et du tronc cérébral. L'image (b) est un schéma d'une coupe transversale du cerveau humain et des structures anatomiques centrales (les images sont tirées de [3, 4])

2.1 Anatomie cérébrale

2.1.1 Cerveau humain

L'intérieur de la boîte crânienne, dénommé encéphale, est constitué du cerveau, du tronc cérébral et du cervelet. Le cerveau forme la majeure partie de l'encéphale et il est divisé en deux hémisphères. La figure 2.1a montre un schéma général de l'encéphale humain. Il est composé de trois matières différentes. La matière blanche (MB) est un tissu principalement constitué d'axones qui est chargé de faire circuler l'information dans le système nerveux. Elle se trouve dans la partie centrale du cerveau ainsi que dans la moelle épinière. La matière grise (MG) est un tissu composé principalement de la partie dendritique des neurones. Celle-ci est située à la surface du cerveau nommée cortex ainsi que dans les noyaux gris centraux. Enfin, le liquide céphalo-rachidien (LCR) est le liquide dans lequel baigne le cerveau. Celui-ci permet, entre autres, d'amortir les chocs et joue un rôle de protection immunologique. Il est produit et renouvelé par le système ventriculaire.

Outre les différents tissus, le cerveau est composé de différentes structures anatomiques distinctes jouant chacune un rôle fonctionnel précis.

Le cortex cérébral est la partie extérieure du cerveau d'une épaisseur de quelques millimètres. Sa surface est importante grâce à la présence de sillons (creux) et de gyrus (bosses). Ces sillons sont aussi utilisés pour définir des sous-régions au sein du cortex cérébral. Le cortex est lié à de nombreuses fonctions neurologiques telles que la motricité et la sensibilité.

Différentes structures sous-corticales ont un impact important sur les fonctions cognitives. La figure 2.1b décrit le positionnement d'une partie de ces structures sur une coupe

axiale du cerveau humain. Le thalamus est une structure anatomique du centre du cerveau composée de matière grise. Il serait un relais et un filtre pour les informations sensorielles. Les noyaux gris centraux sont constitués de plusieurs structures anatomiques composées de matière grise, dont font partie les noyaux caudés, les putamens et les pallidums. L'une des fonctions les plus connues est leur rôle dans le contrôle moteur. L'hippocampe est une structure appartenant au cortex. Elle joue un rôle important dans les fonctions mémorielles. Enfin, les ventricules latéraux font partie du système ventriculaire et participent à la création du liquide céphalo-rachidien. La détection et l'étude de ces différentes structures peut permettre une meilleure compréhension du fonctionnement cérébral ainsi que la détection de certaines maladies. Alzheimer est, par exemple, une maladie qui impacte directement ces structures, se traduisant par une atrophie cérébrale ou par l'apparition de lésions. Ces modifications peuvent ensuite être détectées et analysées par le biais d'images IRM.

2.1.2 Cerveau de brebis

La figure 2.2 décrit l'anatomie générale du cerveau de la brebis. L'une des principales différences entre le cerveau de brebis et le cerveau humain est la taille. La taille moyenne du cerveau humain est de $17 \times 14 \times 13$ cm alors que celle de la brebis n'est que de $8.5 \times 6 \times 5$ cm. Ce qui implique que la résolution des images comparée à la taille du cerveau est plus faible dans le cas de la brebis. De plus, la brebis étant un gros animal, les techniques d'imagerie adaptées aux petits animaux (rats) ne sont pas utilisables *in vivo*. Une autre différence avec l'humain est la position et l'angle du tronc cérébral. Enfin, on peut aussi remarquer la présence de deux importants bulbes olfactifs à l'avant du cerveau.

Ces différences anatomiques ainsi que la faible quantité de données d'apprentissage provenant du cerveau de la brebis expliquent le fait que les outils standards liés au cerveau humain sont difficilement exploitables. La situation est identique dans le cas des images provenant d'autres animaux. C'est pourquoi, nous cherchons à développer des outils de segmentation que seraient plus génériques, et adaptables aux variabilités inter-populations au travers de plus d'interactivité.

2.2 Imagerie par résonance magnétique

L'imagerie par résonance magnétique (IRM) est l'un des moyens privilégiés pour l'étude du cerveau. C'est un type d'imagerie permettant d'acquérir des images 3D des différentes structures anatomiques composant le cerveau. L'image est basée sur le principe de résonance magnétique nucléaire [13, 75]. Ce principe exploite la propriété que possède certains atomes (spin) à pouvoir absorber une énergie et à la relâcher lorsqu'ils sont placés dans un champ magnétique et un champ électromagnétique radio-fréquence. La fréquence de résonance avec l'atome est précise et permet d'obtenir des informations sur la composition moléculaire d'un matériau ou d'un tissu. Cette fréquence dépend du champ appliqué. Dans le cas de l'imagerie IRM, on s'intéresse (quasi exclusivement) à la résonance du proton

2.2. IMAGERIE PAR RÉSONANCE MAGNÉTIQUE

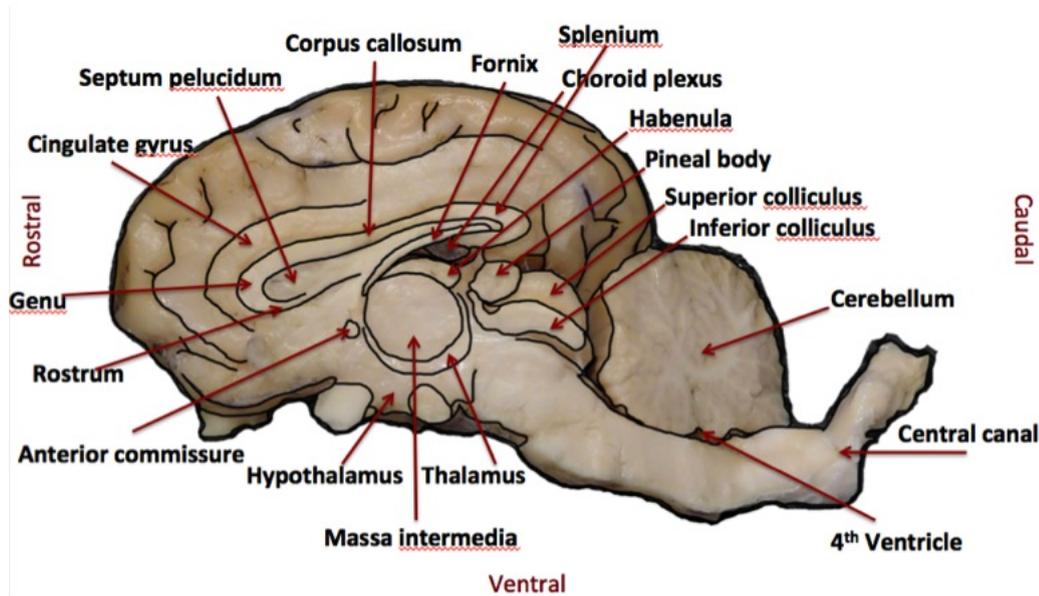


FIGURE 2.2 – Anatomie du cerveau de la brebis (l'image est tirée de [5]).

(H^+) qui est présent en grande quantité dans le cerveau par l'intermédiaire de la molécule d'eau (H_2O).

L'acquisition d'une image se déroule en plusieurs étapes. La première est de soumettre l'objet à un champ magnétique dans le but d'orienter le moment magnétique des protons dans la même direction. Ce champ magnétique est appelé champ longitudinal, noté B_0 , et il est de l'ordre du tesla. Des ondes radio-fréquence sont ensuite émises à une certaine fréquence afin de modifier l'orientation du moment magnétique des protons. Enfin, le champ électromagnétique radio fréquence est arrêté dans le but d'observer la relaxation (retour à l'équilibre) du champ magnétique.

Pour pouvoir localiser dans l'espace les informations de relaxation, des gradients de champ magnétique sont appliqués sur l'objet. Ces gradients permettent de rendre le champ magnétique dépendant de l'espace et de modifier la fréquence de résonance des protons en fonction de l'espace. Une transformée de Fourier inverse permet de reconstruire l'image spatialement.

Suivant les impulsions radio-fréquence appliquées et les caractéristiques du champ électromagnétique mesurées, différentes informations peuvent être obtenues. Le temps de relaxation T1 est le temps de relaxation longitudinal, c'est à dire dans la direction du champ \vec{B}_0 . Il représente donc le temps nécessaire pour que l'aimantation soit de nouveau dans la direction du champ longitudinal \vec{B}_0 après avoir été orientée dans une autre direction par le champ radio-fréquence. Ce temps de relaxation est dépendant de l'agitation moléculaire. Il est par exemple faible pour des tissus comme le LCR et plus élevé pour la matière blanche. C'est donc cette grandeur que l'on utilise lorsque l'on parle d'images IRM pondérées T1.

2.2. IMAGERIE PAR RÉSONANCE MAGNÉTIQUE

Sur une image T1 (cf. figure 2.3a), l'eau (LCR) est représentée par un signal d'amplitude faible alors que la graisse (MG et MB) est représentée par un signal d'amplitude plus forte.

Le temps de relaxation T2 est le temps de relaxation transverse. Lors de l'impulsion radio-fréquence, le champ magnétique tourne de façon cohérente dans le plan transversal. De la même manière, lors de l'arrêt de l'impulsion et à cause de l'agitation moléculaire, la cohérence des moments magnétiques diminue. Le temps nécessaire à la disparition de l'aimantation transversale est alors nommé T2. Une image IRM pondérée T2 est une image mesurant ce temps de relaxation qui est lui aussi dépendant du tissu. Le contraste d'une image T2 est globalement opposé au contraste de l'image T1. Dans une image T2 (cf. figure 2.3a), l'eau est décrite par un signal d'amplitude élevée alors que la graisse est décrite par un signal d'amplitude plus faible.

Un autre type d'imagerie est la densité de protons. Ce type d'images décrit la densité de protons des différents tissus du cerveau.

Ces modalités d'imagerie IRM produisent des images structurales et permettent de décrire l'anatomie du cerveau. L'IRM permet aussi de faire de l'imagerie fonctionnelle comme pourraient le faire les images TEP. Cet autre type d'imagerie vise à détecter l'activité cérébrale et est nommé IRMf. En effet, l'hémoglobine se trouve sous deux formes, l'une oxygénée et l'autre désoxygénée. L'hémoglobine oxygénée n'est pas paramagnétique alors que l'hémoglobine oxygénée l'est. Les deux formes ne réagissent donc pas de la même façon lors d'un examen IRM. Or l'oxygénation du sang est modifiée lorsqu'une activité cérébrale est présente, ce qui peut être ensuite détectée par IRM. La résolution est moins importante que pour l'imagerie IRM structurale mais permet d'obtenir de précieuses informations sur le fonctionnement du cerveau. Un exemple d'image MRIf est montré à la figure 2.3b. Dans ces travaux nous utilisons que des images de type structurales (T1 et T2) et non fonctionnelle.

Les images IRM sont aujourd'hui principalement des acquisitions 3D. Chaque élément qui la compose est un voxel (pixel en 3D). Même si l'IRM permet d'obtenir des images 3D d'une grande résolution, elle est sujette à un certain nombre d'artefacts qui peuvent rendre son exploitation difficile.

Un bruit est présent dans les images IRM. Celui-ci est dû à l'ensemble de la chaîne d'acquisition et de traitement. Le type d'antenne utilisé, la qualité du champ magnétique \vec{B}_0 ou les paramètres utilisés pour la séquence IRM interviennent dans ce bruit. Ce bruit suit une distribution de type Rice. Il peut être modélisé par un bruit gaussien dans les régions où le rapport signal sur bruit est assez élevé. Le deuxième artefact est l'effet de volume partiel. En effet, même si la résolution de l'image IRM est élevée, le volume de matière représenté par chaque voxel n'est pas suffisamment petit pour ne contenir qu'une seule sorte de tissu. Les voxels ne sont pas ponctuels. Il se peut alors que plusieurs tissus différents se trouvent à l'intérieur du même voxel. Cela se traduit par un effet de mélange dans la réponse IRM mesurée. Cette effet est donc principalement présent à l'interface entre les différents tissus. Les inhomogénéités de champ sont un troisième artefact qui se traduit par de lentes variations d'intensité dans l'image IRM car le champ magnétique B_0 ou le champ radio fréquence ne sont pas forcément spatialement homogènes. Il existe d'autres artefacts tel que les artefacts de mouvement du sujet qui peuvent altérer l'images

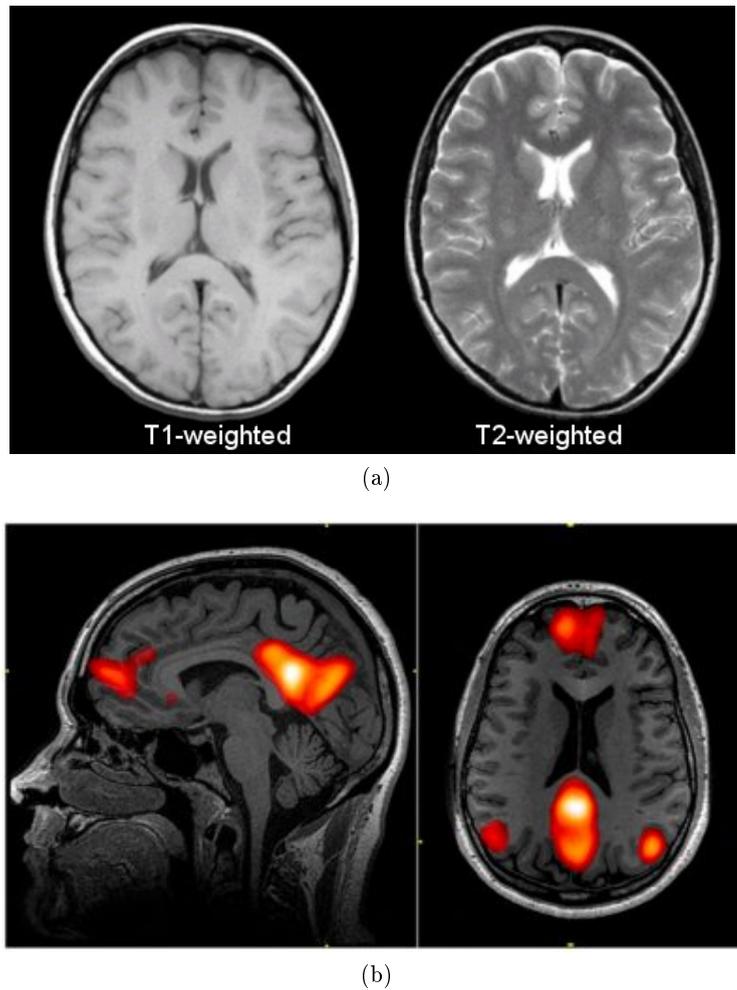


FIGURE 2.3 – L'image (a) illustre une image IRM pondéré T1 et une image IRM pondéré T2. L'image (b) illustre un exemple d'image IRM fonctionnelle (les images sont tirées de [2, 6])

ou l'artefact de susceptibilité des tissus qui entraîne des modifications locales du champ B_0 .

Chapitre 3

Modélisation des informations *a priori*

Les images médicales possèdent en général un contraste assez faible. Beaucoup de structures anatomiques sont représentées par des niveaux de gris similaires, ce qui les rend difficilement distinguables. L'un des moyens de répondre à ce problème est d'utiliser des informations *a priori*, construites à l'aide d'une base d'apprentissage. Les structures anatomiques étant relativement stables en terme de forme et de topologie, il est possible de modéliser les informations de position, de forme et d'intensités des structures de différentes manières.

3.1 Les atlas

Les atlas sont une forme d'information *a priori* spatiale permettant de guider la localisation des structures anatomiques au sein d'images médicales. Dans le cas classique, un atlas A (utilisable pour la segmentation d'images) est constitué de deux ensembles de données. Tout d'abord, un *template* T apporte les informations liées aux intensités classiques des régions à segmenter dans les images acquises. Puis, une carte de labels L ou des cartes de probabilités de présence des labels P_r , associées à l'image template viennent en complément pour apporter l'information de localisation des différentes structures connues *a priori*. Les méthodes utilisant ces atlas pour la segmentation automatique sont devenues très courantes [22, 46]. La création d'atlas nécessite une base d'apprentissage constituée d'images médicales réelles et d'images labellisées. Il existe plusieurs formes d'atlas utilisées pour la segmentation d'images cérébrales que nous allons maintenant décrire rapidement.

Atlas Topologique : les atlas topologiques construits à partir d'un seul individu ont été les premiers atlas produits. Les premiers atlas n'étaient pas digitaux. Le référentiel était défini à partir de points de repère anatomiques. C'est le cas pour l'atlas de Talairach [93] qui a été longtemps utilisé. Il provenait d'un cerveau *post-mortem* segmenté coupe par coupe. Les commissures antérieures et postérieures du cerveau sont ici utilisées pour définir le référentiel. Puis, de nouveaux atlas ont été créés à partir d'acquisitions de différentes

modalités d'imagerie [52, 27]. La labellisation de l'atlas est donc reliée directement à une image réelle associée (ex. IRM ou CT scan). Cette image correspond au template T de l'atlas. Certains visent à décrire des cerveaux atteints de certaines maladies comme la sclérose en plaques dans [1]. Ce type d'atlas fournit des résultats suffisants lors de recalages intra-individuels, c'est à dire lorsque les deux images à recaler proviennent d'un même individu. Néanmoins, ce type d'atlas montre ses limites lorsque les variations anatomiques deviennent importantes entre deux individus différents.

Multi-atlas : les multi-atlas sont composés de plusieurs atlas topologiques (liste d'atlas topologique). L'information provenant de chaque atlas est fusionnée après avoir recalé chacun (totalité ou une partie seulement) des atlas vers l'image que l'on cherche à segmenter. Dans le cas classique, tous les atlas ont été labellisés par des experts et chaque image est considérée de façon similaire. Aujourd'hui, certains multi-atlas sont créés artificiellement, soit en utilisant une base d'images labellisées partiellement segmentée automatiquement [102], soit en déformant synthétiquement des images [50]. Les techniques de fusion nécessaire à l'utilisation de ce type d'atlas seront décrites dans la section 4.4. Ce type d'atlas fournit généralement des résultats de bonne qualité lorsque la base d'images est de taille suffisante. Cependant, ces atlas nécessitent le recalage d'un grand nombre d'images impliquant un temps de calcul important.

Atlas probabiliste : les atlas probabilistes compensent d'une autre manière le manque de diversité qu'un atlas topologique peut représenter. Ces atlas sont construits à partir de plusieurs individus. Ils fusionnent les informations provenant de plusieurs images d'apprentissage pour créer une image moyenne contenant une partie de l'information de chaque image [87, 55]. Plusieurs projets ont été menés pour construire des atlas à partir d'une grande population et de différentes modalités ; 452 individus ont, par exemple, été utilisés pour construire l'atlas cité dans [38].

Avec cette technique, l'image template est créée à partir du recalage de toutes les images d'apprentissage réelles entre elles. Les transformations appliquées à ces images sont ensuite appliquées aux images d'apprentissages étiquetées afin de créer les différentes cartes de probabilité. Chacune de ces cartes de probabilité est un volume de données qui stocke la probabilité de présence de la région à une position spécifique. Un exemple d'atlas est illustré sur la figure 3.1 avec un template IRM et une carte de probabilité de présence pour chaque tissu.

Les atlas probabilistes sont aussi utilisés pour modéliser la croissance des organes (comme le cerveau) au cours du temps. La croissance du cerveau est ainsi caractérisée par l'évolution du template ainsi que des cartes de probabilité associées [42, 56]. Il est alors possible d'interpoler l'atlas à un instant donné quelconque à partir des différents atlas existants.

Ce type d'atlas a aussi permis de décrire le cerveau d'animaux, et en particulier le cerveau de souris [60] et le cerveau de primates [65]. Récemment dans [67, 33], les auteurs ont aussi construit un atlas probabiliste du cerveau ovin. La figure 3.2 montre un exemple

3.1. LES ATLAS

d'atlas probabiliste de ce type de cerveau. Les trois cartes de probabilité représentent la probabilité d'appartenance des voxels respectivement à la matière blanche, la matière grise et au liquide céphalo-rachidien. Le template T1 est l'image moyenne IRM associée à ces cartes de probabilité.

La création de ce type d'atlas nécessite le choix d'une image ou d'un repère de référence vers laquelle les images de la base d'apprentissages seront recalées. Le choix du type de recalage [80] est aussi nécessaire à la construction de ces atlas (cf. section 3.2).

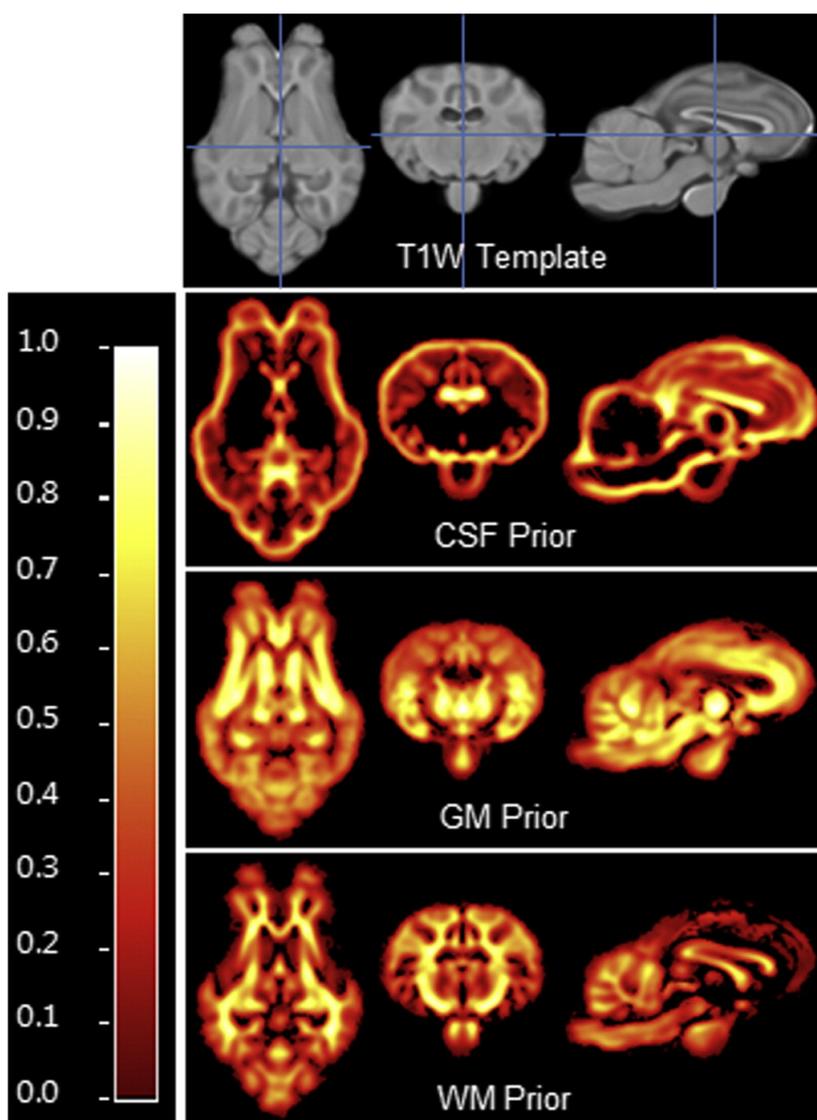


FIGURE 3.2 – Atlas probabiliste du cerveau de la brebis composé d'une image template IRM T1 et de 3 cartes de probabilité de présence de la région (LCF, MB, MG). La probabilité d'appartenance des voxels aux régions est reliée à l'échelle d'intensité située à gauche (l'image est tirée de [33]).

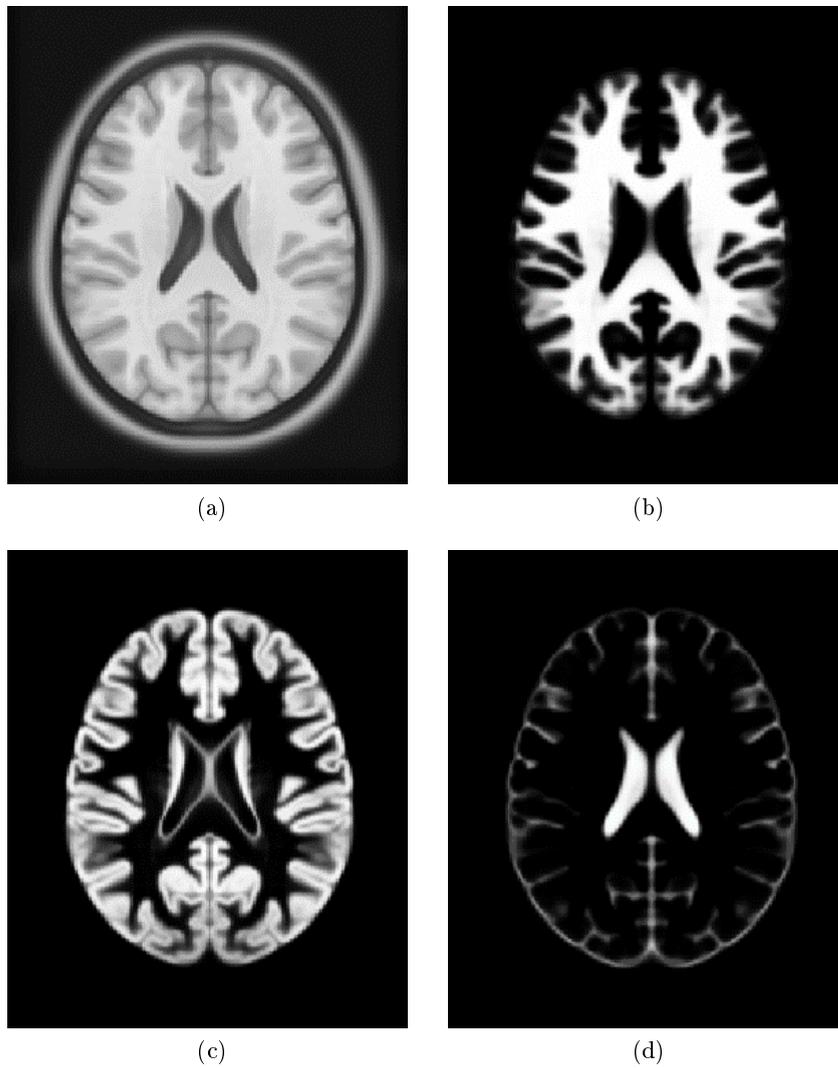


FIGURE 3.1 – Atlas ICBM du cerveau humain avec (a) l'image IRM T1, (b) la carte de probabilité de la matière blanche, (c) la carte de probabilité de présence de la matière grise et (d) la carte de probabilité de présence du liquide céphalo-rachidien [64]. Plus l'intensité est claire pour la probabilité d'appartenance est forte.

Atlas surfacique : Généralement les atlas du cerveau sont des atlas volumiques. Mais, il est parfois difficile de modéliser la forme des régions du cortex cérébral avec ce type de modèles. En effet, les formes des régions et la position des sillons sont très variables. Une des possibilité pour segmenter ce genre de région avec des méthodes basées atlas est d'utiliser des atlas surfacique [37, 34, 82]. Dans ces atlas, la surface du cortex est extraite et déformée pour être projetée sur une sphère. Il est alors possible de recalcr les surfaces dans le référentiel de la sphère et ainsi produire des atlas probabilistes ou segmenter une image. Cependant ce type d'atlas reste très spécifique au cas du cortex cérébral et ne peut être appliqué sur tous les types de structure anatomique.

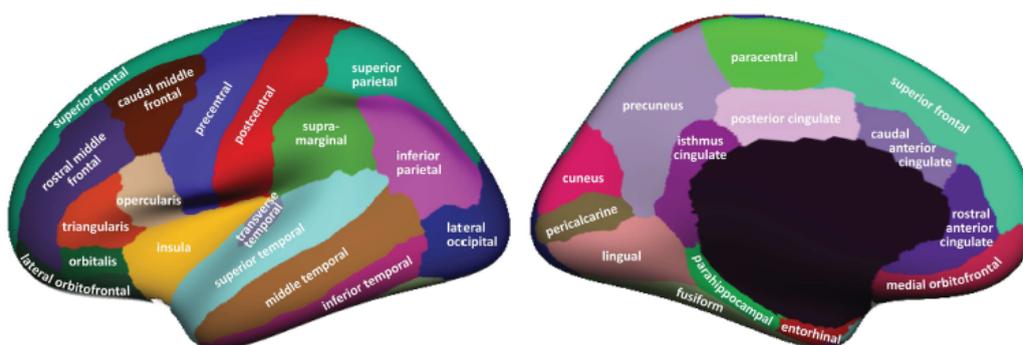


FIGURE 3.3 – Illustration d'un atlas surfacique des régions corticales sur une version dilatée du cortex cérébral. Les régions proviennent de l'atlas de cerveau humain du template FreeSurfer "fsaverage" de l'hémisphère gauche. (l'image est tirée de [54])

Dans la littérature, les atlas sont probablement le type d'information *a priori* le plus utilisé pour la segmentation des images médicales. Pour résumer, les atlas topologiques avec un seul sujet sont historiquement les premiers et peuvent être encore utilisés pour les segmentations intra-individuelles. Les atlas probabilistes et multi-atlas sont plus souples et stockent l'information d'une population. Les multi-atlas sont plus efficaces mais nécessite un long temps d'utilisation. Les atlas probabilistes fournissent des résultats un peu plus faibles mais reste rapides d'utilisation. Les atlas surfaciques sont une représentation efficace des structures corticales mais restent spécifiques à un type de structure anatomique. Dans tous les cas, chaque modélisation des informations *a priori* reste dépendante des méthodes de segmentation qui l'utilisent.

3.2 Recalage d'images

La construction des atlas nécessite parfois de faire des recalages pour faire correspondre spatialement les voxels de deux images. De même, l'utilisation de ces atlas repose généralement sur le recalage de l'atlas vers l'image que l'on veut segmenter. Le recalage se traduit par un transformation d'une image vers l'autre image. Dans [90], les auteurs décrivent et classent les différentes méthodes de recalage d'images qui sont classiquement utilisées dans

le traitement d'images médicales. Une méthode de recalage s'appuie généralement sur trois composantes : un modèle de transformation, une métrique, et une méthode d'optimisation. Chacune de ces étapes est maintenant rapidement présentée. On se place ici dans le cas où l'on souhaite recalibrer l'image I_1 vers une image I_2 .

3.2.1 Transformations

Transformation affine : les transformations rigides ou affines sont des modèles de transformation simples. Ils n'offrent qu'un nombre de liberté (12) relativement faible comparé à d'autres types de transformation. La transformation peut se représenter sous la forme d'une transformation τ du type :

$$\tau = AX + b = \begin{bmatrix} a_{00} & a_{01} & a_{02} & b_0 \\ a_{10} & a_{11} & a_{12} & b_1 \\ a_{20} & a_{21} & a_{22} & b_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix}$$

La matrice A décrit les transformations de type rotation, cisaillement et changement d'échelle tandis que b modélise la translation. Ces méthodes sont rapides mais la qualité n'est pas toujours en accord avec l'application. De plus elle ne permet pas de corriger la variabilité inter-individuelle. Ces recalages peuvent être, dans certains cas utilisés afin d'initialiser des méthodes plus complexes. Il est aussi possible d'appliquer des recalages affines par morceaux afin d'augmenter les possibilités de déformation de l'image [71].

Transformation basée sur des modèles non paramétriques : de nombreux modèles non linéaires ont été proposés pour le recalage. Des modèles inspirés de la physique tel que les modèles élastiques, les modèles visqueux ou les modèles de diffusion permettent d'offrir un très grand nombre de degrés de liberté de transformation [20, 94]. Ils fournissent un champ de déplacement de transformation dense qui décrit le déplacement de chaque voxel. Ces modèles fournissent une très grande flexibilité mais restent lourds en terme de complexité (d'un point de vue calculatoire).

Transformation basée sur des modèles paramétriques : Des méthodes de recalage inspirées de la théorie de l'interpolation ont aussi été proposées. Ces techniques permettent de décrire un grand nombre de transformations tout en utilisant un nombre moindre de paramètre que les modèles déformables inspirés de la physique. Les méthodes basées spline font l'hypothèse que des points de contrôle sont connus dans les deux images. Les splines permettent alors d'interpoler le déplacement nécessaire pour faire correspondre les points de contrôle des deux images. Ces points de contrôles peuvent être des points anatomiques de référence mais il est aussi possible de les définir à partir d'une grille sur l'intensité des voxels. Dans le cas des *Thin Plate Spline* [16], la fonction de transformation est basée sur des *radial basis functions*. Les *free form deformations* sont un autre type de déformation,

elles consistent à représenter les images sous forme de grilles puis d'appliquer une déformation directement sur les nœuds de cette grille. La "tension" appliquée sur les nœuds de la grille est souvent représentée par une fonction B-spline [81]. Le recalage par B-spline à l'avantage de fournir un recalage local et d'être efficace en temps de calcul.

De manière général, il existe un nombre important de types de transformation pour le recalage d'images. Il faut souvent trouver un compromis entre la précision et la complexité du type de transformation. Dans ces travaux nous avons choisies d'utiliser un recalage déformable paramétrique de type B-spline.

3.2.2 Métriques

La métrique, associée à une technique de recalage, est la fonction objectif qui doit être minimisée lors de l'étape d'optimisation. Deux principaux types de métriques existent : les méthodes basées sur des points de contrôle et les méthodes basées sur les intensités de l'image. Pour le premier cas, la métrique fournit une fonction objectif entre des points de contrôle p_i et q_i localisés sur les deux images I_1 et I_2 . Ils peuvent être déterminés de manière automatique ou manuelle. Ces points de contrôle peuvent être aussi des surfaces ou des volumes. Le but est de, si nécessaire, faire la correspondance entre ces points, puis de minimiser une distance entre ces points de contrôle [9]. La similarité *Sim* entre les deux images peut alors s'écrire :

$$Sim = - \sum_i \|\tau(p_i) - q_i\|^2 \quad (3.1)$$

où τ est la transformation permettant de passer des coordonnées de l'images I_1 à celles de l'image I_2 .

La plus simple métrique basée sur les intensités de l'image, est la somme des différences des intensités au carré (SSD). La similarité entre deux images peut alors être définie par :

$$Sim = - \frac{1}{N} \sum_{i=1}^N \|I_1(\tau(v_i)) - I_2(v_i)\|^2 \quad (3.2)$$

où $I_1(v_i)$ et $I_2(v_i)$ sont les intensités de l'image aux voxels v_i pour les images I_1 et I_2 et N le nombre de voxel présents dans la région où les deux images se superposent. Cependant elle peut être de relativement faible qualité car elle laisse place aux minima locaux. De nombreuses améliorations ont été proposées, en particulier pour les approches hiérarchiques. Une autre limitation de la SSD est qu'elle ne permet pas de recalage multi-modal efficace car elle suppose qu'il existe un lien entre les niveaux de gris des deux images. Les méthodes provenant de la théorie de l'information ont permis de pallier à ces problèmes. En particulier, l'information mutuelle (MI) [96, 101, 61] ne suppose plus cette relation entre

les intensités des deux images. Elle est définie telle que :

$$Sim(I_1, I_2) = \mathcal{H}(I_1) + \mathcal{H}(I_2) - \mathcal{H}(I_1, I_2) \quad (3.3)$$

avec $\mathcal{H}(I)$ l'entropie de Shannon Wiener de l'image I et $\mathcal{H}(I_1, I_2)$ est l'entropie conjointe entre l'image I_1 et I_2 . Le ratio de corrélation [77] ou l'information mutuelle régionale [91] sont des métriques basées sur le même principe.

Le choix de la métrique est surtout dépendant des modalités d'images que l'on veut recalculer. Dans ces travaux nous utilisons les métriques de type SSD mais aussi de type information mutuelle.

3.2.3 Optimisations

Enfin, la méthode d'optimisation liée à l'étape de recalage a pour but de minimiser la fonction objectif choisie sachant la transformation utilisée. Les méthodes les plus courantes sont la descente de gradient, la descente de gradient conjugué, ou l'algorithme de Gauss-Newton dans le cas d'optimisation continue. Dans [54], plusieurs de ces méthodes d'optimisation ont été comparées dans le cas du recalage d'images médicales et d'une transformation de type B-spline. Les méthodes d'*iterated conditional modes* (ICP), graph cut et alpha expansion sont, quant à elles, utilisées dans le cas d'optimisations discrètes. Ces dernières sont en général moins dépendantes de l'initialisation mais peuvent être moins précises suite à la discrétisation.

3.3 Modélisation des informations topologiques

Comme nous l'avons vu précédemment, la structure générale d'un organe humain (comme le cerveau) est stable entre les individus. Pour le cerveau, la matière blanche est toujours dans une position interne par rapport à la matière grise. De même, le Noyau caudé est toujours proche du ventricule latéral. Cette stabilité rend possible l'utilisation de relations spatiales afin de guider les méthodes de segmentation.

3.3.1 Relations spatiales

Ces informations structurelles ou topologiques peuvent être modélisées de différentes manières pour ensuite servir à guider la segmentation, la recherche, mais aussi l'interprétation d'images. Ces informations topologiques décrivent les relations entre les différents objets et régions de l'image.

Les relations spatiales peuvent décrire des notions distances entre les régions mais aussi des notions de direction. Certaines modélisations sont plus adaptées à l'interprétation

3.3. MODÉLISATION DES INFORMATIONS TOPOLOGIQUES

d'images par le biais de la compréhension de la topologie. Elles fournissent une description quantitative à propos de la position d'une région par rapport à une autre. Par exemple, les histogrammes d'angles ou les histogrammes de forces modélisent des informations directionnelles entre deux régions. Les histogrammes d'angles représentent les angles existant entre tous les points d'une région A et tous les points d'une région B . Les histogrammes de forces représentent les relations entre les sections longitudinales des deux régions [63, 100]. Ces relations fournissent une représentation générale du positionnement directionnel entre deux structures. Cependant cette modélisation n'est pas adaptée pour le problème opposé qui est de déterminer la position d'un objet par rapport aux autres, car elles ne fournissent pas directement un résultat dans l'espace de l'image.

Pour cela, les relations spatiales floues décrites dans [15] sont mieux adaptées. Elles fournissent une modélisation des relations spatiales du type "*à gauche*" directement dans l'espace de l'image. L'information est fournie sous forme de cartes de probabilités pour prendre en compte l'imprécision des informations. Cette carte de probabilités mesure comment les voxels vérifient une certaine relation par rapport à une certaine région présente dans l'image.

Les relations spatiales peuvent représenter aussi bien les informations de distance telles que "*proche de*" ou "*à telle distance de*", que les informations de direction du type "*à gauche de*".

Les relations de distance entre une région A et une région B peuvent être décrites avec une fonction d'appartenance trapézoïdale f définie à l'aide de 4 distances $0 \leq n_1 \leq n_2 \leq n_3 \leq n_4$ (cf. 3.4). Le noyau de la fonction est inclus dans $[n_2, n_3]$ et le support dans $[n_1, n_4]$. La région B a alors de forte de chance de se trouver à une distance comprise entre n_2 et n_3 de A . La fonction trapézoïdale permettant de modéliser l'imprécision liée à la relation de distance. La carte de probabilité liée à cette relation est alors définie par :

$$\mu_d(P) = f(d_A(P)) \quad (3.4)$$

où P est un point de l'image et d_A est une carte de distance par rapport à l'objet A . La probabilité provenant de la carte μ_d d'un voxel P est donc maximale lorsque P est compris entre n_2 et n_3 puis diminue jusqu'à devenir nulle à l'extérieur de n_1 et n_4 . Les distances modélisées par la fonction f peuvent être apprises sur une base d'apprentissage. Les relations d'adjacence peuvent être modélisées en fixant n_1 et n_2 à 0. Les relations "*loin de*" peuvent être modélisées par n_3 et n_4 à l'infinie. La figure 3.4 décrit un exemple de relations spatiales liées à la distance entre deux régions.

La figure montre des exemples de carte de probabilité provenant des relations "*a gauche*", "*a droite*", "*en haut*" et "*en bas*".

De façon analogue, on peut définir pour les orientations une carte de probabilités d'appartenance à la relation "*l'objet B est dans la direction \vec{u}_α par rapport à A* ". Si P est un point de l'image, Q est un point de la région A et $\beta(P, Q)$ est l'angle entre le vecteur QP

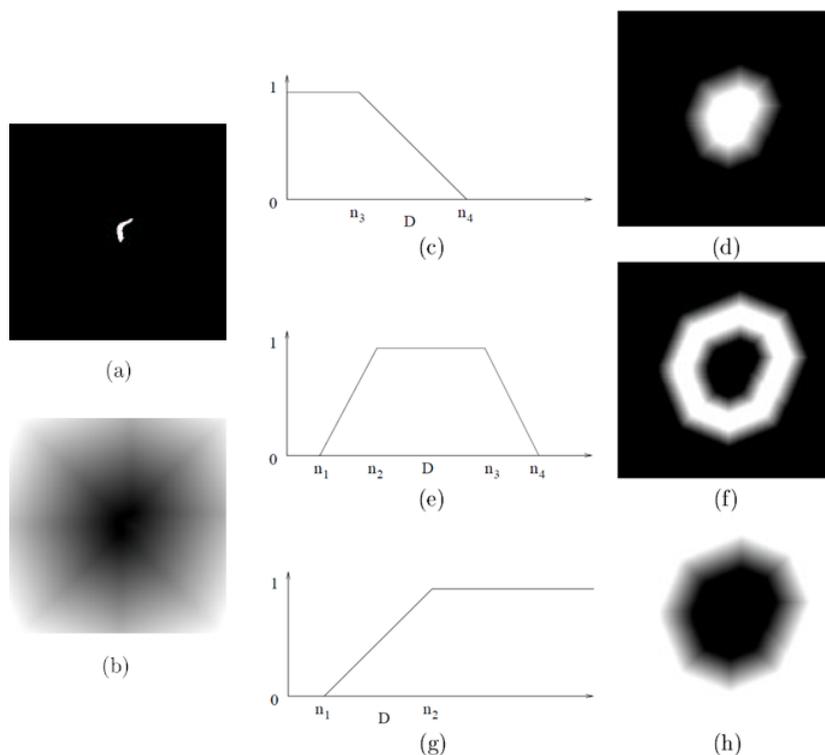


FIGURE 3.4 – Exemple d'une relation spatiale de type "distance". La figure (a) représente la région de référence, (b) la carte de distance, (c) représente la fonction de d'appartenance pour la relation "*distance inférieur à D*" et (d) l'ensemble flou correspondant. (e) représente la fonction d'appartenance pour la relation "*à une distance D*" et (f) l'ensemble flou correspondant. (g) représente la fonction d'appartenance pour la relation "*à une distance supérieur de D*" et (h) l'ensemble flou correspondant. Les zones d'intensité élevée correspondant à une probabilité élevée et les zones d'intensité faible à une probabilité faible. (l'image est tirée de [28])

et le vecteur \vec{u}_α . On peut poser :

$$\beta(P, Q) = \arccos\left[\frac{\vec{QP} \cdot \vec{u}_\alpha}{\|\vec{QP}\|}\right] \quad \text{et} \quad \beta(P, P) = 0 \quad (3.5)$$

Pour chaque point P , on cherche l'angle où β est minimum tel que :

$$\beta_{min}(P) = \min_{Q \in A} \beta(P, Q) \quad (3.6)$$

La carte de probabilité de cette relation est alors définie par :

$$\mu_\alpha^A(P) = \max\left[0, 1 - \frac{2}{\pi}\beta_{min}(P)\right]$$

où $\max[0, 1 - \frac{2}{\pi}x]$ joue le rôle d'une fonction décroissante. $\mu_\alpha^A(P)$ est appelée la cohé-

3.3. MODÉLISATION DES INFORMATIONS TOPOLOGIQUES

rence du voxel P dans la direction α par rapport à la région A . La probabilité provenant de la carte μ_α d'un voxel P est donc maximum lorsque P est dans la direction \vec{u}_α puis diminue lorsque celui-ci s'en écarte.

Les relations spatiales permettent de décrire la position des régions les unes par rapports aux autres. Celles-ci peuvent être couplées à une autre méthode de segmentation afin de réduire l'espace de recherche et guider la suite de la segmentation. Elles peuvent être utilisées pour la segmentation incrémentale lorsque chaque région est recherchée l'une après l'autre. La segmentation de chaque nouvelle région est capable de fournir de nouvelles informations sur la position spatiale des prochaines régions à rechercher. La combinaison de l'ensemble des informations permet de préciser l'espace de recherche d'une région [29, 66].

3.3.2 Autres informations topologiques

La modélisation des informations topologiques a aussi été utilisée pour la segmentation des sillons du cerveau. La segmentation, et particulièrement la labellisation du cortex cérébral ou des sillons, est une tâche toujours difficile car ces régions sont de forme très variables entre les individus. Pour résoudre ce problème, dans [105] les auteurs créent un graphe topologique des sillons. Comme pour la plupart des méthodes dédiées à la segmentation du cortex, celle-ci est appliquée sur une version surfacique du cortex. Chaque sillon est représenté par un nœud dans un graphe. Pour déterminer la notion d'adjacence entre tous les sillons, une dilatation des sillons est effectuée dans l'espace sphérique. Lorsque toutes les dilatations ont été effectuées, un arc est construit entre les sillons qui sont en contact. Le nombre de dilatations n_d nécessaire ainsi que la profondeur moyenne h_a des sillons sont utilisés pour construire les attributs du graphe. On définit h_a par :

$$h_a = \frac{1}{N_p} \sum_{i=1}^{N_p} (r_{12}^i + r_{21}^i) \quad (3.7)$$

avec N_p le nombre de sommets voisins entre le sillon S_1 et le sillon S_2 . r_{12}^i et r_{21}^i sont respectivement la profondeur géodésique du sillon S_1 et S_2 comparée à leur sommet (cf. figure 3.5). Les arcs entre les sillons stockent finalement une valeur numérique provenant d'une combinaison linéaire de n_d et h_a . Les nœuds stockent des attributs classiques de forme. Ensuite, les sillons de l'image de cerveau à segmenter sont détectés puis sur-segmentés en *sous-sillon*, afin que chaque sous-sillon n'appartienne qu'à un seul véritable sillon. Un graphe est construit de telle sorte que les nœuds du graphe représentent les sous-sillons.

Un graphe modèle a aussi été construit préalablement à l'aide d'une base d'apprentissage. Une méthode d'appariement de graphe est appliquée pour labelliser les nœuds du graphe construit sur l'image à segmenter. Le but est de faire correspondre les nœuds du graphe provenant de l'image à segmenter avec les nœuds du graphe modèle afin d'obtenir une labellisation des sous-sillons. Cette modélisation (graphe de sous-sillons) et ces attributs (profondeur des sillons) est efficace pour les structures corticales mais reste spécifique à la segmentation de la surface corticale.

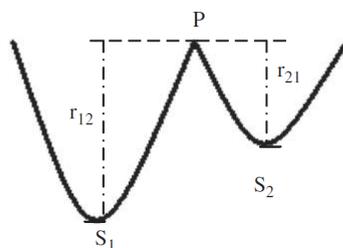


FIGURE 3.5 – Illustration des profondeurs r_{12} et r_{21} liées aux sillons S_1 et S_2 respectivement, qui sont utilisées pour définir l'attribut h_a présent dans l'arc entre le sillon S_1 et S_2 .

3.4 Bilan

Nous avons décrit différentes manières de modéliser les connaissances *a priori* pouvant être exploitées pour segmenter des structures anatomiques. Les atlas sont l'un des outils les plus couramment utilisés permettant de stocker des connaissances *a priori* de forme et d'intensité des régions. Il en existe plusieurs types ayant chacun leurs avantages et leurs défauts. Les données liées aux atlas probabilistes sont, par exemple, plus synthétiques et permettent d'encoder l'information de plusieurs images d'apprentissage dans un seul espace de référence en contrepartie d'une perte d'information. À l'inverse, les multi-atlas conservent l'ensemble des informations mais nécessitent le stockage (et donc l'utilisation) d'une importante quantité de données. Enfin, les atlas surfaciques sont plus dédiés à un type de région particulière comme les atlas surfaciques du cortex cérébral. L'ensemble de ces atlas est fortement dépendant des méthodes de recalage associées que nous avons aussi rapidement présentées dans cette section. Nous verrons par la suite qu'il est aussi possible de conserver les informations sous forme de modèles de classificateurs (cf. deep learning section 4.4) ne nécessitant pas l'utilisation d'atlas au préalable (segmentation *end-to-end*). Enfin nous avons vu aussi que les relations spatiales sont aussi des informations intéressantes pouvant être apprises et conservées afin de guider une segmentation. Les structures anatomiques d'un organe ayant, en général, une topologie stable, ces informations sont particulièrement adaptées.

Chapitre 4

Segmentation guidée

La segmentation des images médicales est très souvent guidée pour compenser le faible contraste à l'intérieur de ces images. La segmentation peut être aussi bien guidée par l'utilisateur que par des informations apprises à l'aide d'une base d'apprentissage. Nous verrons tout d'abord les méthodes interactives utilisables lorsqu'aucune information *a priori* n'est disponible. Nous décrirons ensuite les méthodes de segmentation par deep learning qui ont fait leur apparition récemment. Nous présenterons ensuite les méthodes fonctionnant par classifications de voxels et par fusions de données (classiquement utilisés lors des segmentations basées sur des atlas). Enfin nous décrirons une méthode de correction de segmentation *a posteriori* utilisable pour améliorer la qualité des segmentations initiales.

4.1 Méthodes interactives

Les segmentations interactives sont utilisées lorsque les problèmes de segmentation sont difficiles et que les informations *a priori* ne sont pas disponibles. Ainsi, elles permettent, grâce à des informations fournies par l'utilisateur, d'obtenir des résultats fiables et contrôlés. Ces techniques ont d'abord été développées pour les images 2D avec des outils de type contours actifs. Certaines d'entre elles ont par la suite été adaptées pour être utilisées sur des images 3D. L'utilisateur peut intervenir à plusieurs endroits de la chaîne de segmentation, depuis l'initialisation, en fixant des paramètres ou en plaçant des "graines", jusqu'à la correction manuelle du résultat de segmentation intermédiaire ou final.

4.1.1 Graph-cut

Les méthodes de type *graph cut* sont appliquées dans le cas de segmentations interactives d'images médicales [104, 17] ou naturelles. Elles ont l'avantage de chercher les minima globaux de la fonction objectif contrairement à d'autres méthodes (de type déformable recherchant) les minima locaux.

Dans le cas des segmentations basées graph-cut et pour une segmentation binaire (*objet* et *fond*), l'image est représentée sous la forme d'un graphe de type champ de Markov [49].

Le but est ensuite de déterminer la coupe d'énergie minimale permettant de séparer en deux classes les nœuds du graphe. En général, les nœuds représentent les voxels et les arcs représentent une relation d'adjacence entre ces voxels. Un nœud *source* et un nœud *terminal* sont ajoutés, l'un étant connecté à l'objet alors que le deuxième est connecté au fond. Les arcs liant les nœuds terminaux aux voxels sont nommés les *t-links*, tandis que ceux liant les voxels entre eux sont nommés les *n-links*. Les *t-links* sont des contraintes dites dures, défini par l'utilisateur. Les *n-links* sont des contraintes dites souples dépendant du gradient de l'image. Les arcs entre les voxels de forts gradients possèdent des poids faibles favorisant la coupe. Un poids infini est attribué aux arcs reliant les nœuds terminaux avec les nœuds des voxels. Des poids peuvent aussi être fixés sur les nœuds fonction de l'intensité de la région.

Si $S = (s_1, \dots, s_n)$ est une labellisation de l'image avec s_i le label du voxel v_i , les contraintes, dites souples, liées à l'image peuvent s'écrire :

$$E^{noeud}(S) = \sum_{i \in N} E_i^{noeud}(s_i) \quad (4.1)$$

$$E^{arc}(S) = \sum_{\{i,j\} \in N} E_{\{i,j\}}^{arc} \cdot \delta(s_i, s_j) \quad \text{et} \quad \delta(s_i, s_j) = \begin{cases} 1 & \text{si } s_i \neq s_j \\ 0 & \text{sinon} \end{cases} \quad (4.2)$$

où E_i^{noeud} est une fonction de similarité entre le voxel v_i et l'objet s_i . Il peut se traduire par une similarité entre l'intensité du voxel et un histogramme de l'objet. De même, $E_{\{i,j\}}^{arc}$ est une fonction de coût exprimant la similarité entre le voxel v_i et le voxel v_j . Elle représente une pénalité de discontinuité entre les deux voxels. La fonction est élevée dans le cas où les deux voxels sont similaires, et faible lorsqu'ils sont vraiment différents. Le but est ensuite de trouver la segmentation minimisant la fonction de coût total :

$$E(S) = \lambda \cdot E^{noeud}(S) + E^{arc}(S) \quad (4.3)$$

avec λ un paramètre modulant l'importance du coût "région" comparé au coût des contours. Plusieurs méthodes de coupe minimale de graphe sont utilisables pour déterminer la segmentation optimale du graphe construit à partir d'une image. De manière générale des algorithmes de type *max-flow* [18] sont utilisés.

Concrètement, lors de la segmentation, l'utilisateur va pouvoir fournir des graines pour les éléments recherchées, objet et fond. Ces informations permettront non seulement de fournir les poids aux arcs reliant les deux nœuds terminaux aux voxels de l'image mais aussi de définir un histogramme représentatif de la région. Le poids des voxels dépendra alors du lien entre l'intensité des voxels et l'histogramme. L'intérêt de la méthode réside dans la capacité à fusionner les contraintes dures fournies par l'utilisateur avec les contraintes souples liées à l'image. De plus, il est possible d'ajouter des graines après la visualisation d'un résultat pour ainsi corriger la précédente segmentation. Les nouvelles graines permettront de créer de nouveaux *t-links* et donc changer potentiellement la coupe minimale, de la segmentation.

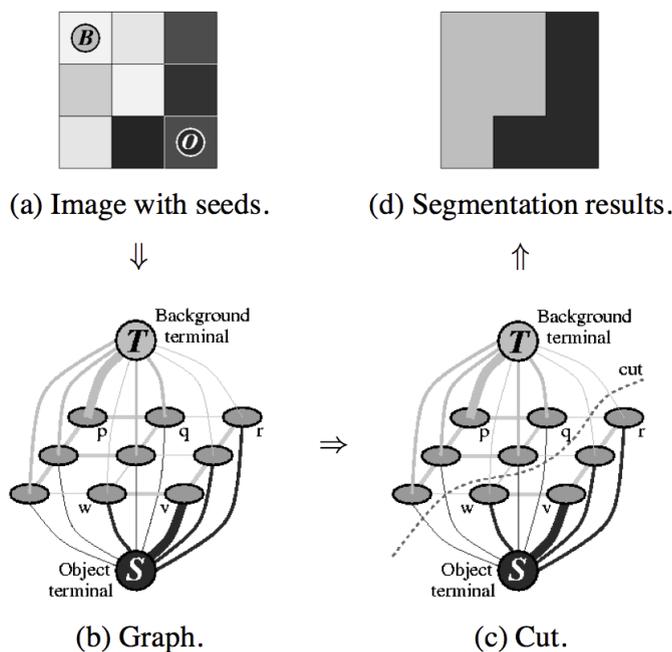


FIGURE 4.1 – Exemple de graph-cut sur une image 3×3 . (a) Deux graines ont été positionné sur l'image (objet O et fond B). (b) Le graphe correspondant est construit puis (c) la coupe de coût minimale est détectée fournissant (d) une segmentation binaire de l'image (l'image est tirée de [104]).

Dans un premier temps, les méthodes de graph-cut ont été développées pour la segmentation d'images naturelles 2D en deux classes, objet et fond. Elles ont ensuite été adaptées pour la segmentation d'images médicales 3D, et pour les problèmes multi-classes [104].

4.1.2 Modèles déformables interactifs

Les méthodes de types contours ont aussi été proposées pour la segmentation interactive en imagerie médicale. Les contours actifs, les *level-set* ou les ciseaux intelligents font parties des méthodes de segmentation basée contours les plus courantes. Des modifications interactives de la fonction d'énergie (via des paramètres) ou dans l'initialisation permet de faire converger le résultat vers celui attendu par l'utilisateur.

Les méthodes de ciseaux intelligents (*live-wire*) sont des méthodes qui permettent de créer itérativement un contour en trouvant le chemin de coût minimal entre deux points positionnés par l'utilisateur [35]. Dans [74], les auteurs décrivent une méthode où l'utilisateur va placer sur l'image un premier point p sur le contour de la région qu'il veut délimiter. Une carte de coût $M(p)$ est calculée pour tous les voxels q de l'image représentant le coût minimal cumulé du chemin entre p et q . Ce coût minimal est aussi associé à son chemin correspondant, qui peut être visualisé par l'utilisateur lorsque celui-ci choisit un deuxième

point sur le contour de la région. L'utilisateur peut appliquer ce principe jusqu'à avoir re-fermé le contour. Les coûts pour créer les cartes M dépendent principalement du gradient d'intensité sur les images mais aussi de la distance pour permettre un lissage de la courbe. Ce type de méthodes a d'abord été utilisé pour les images 2D et a ensuite été étendu au cas des images 3D médicales.

Dans le cas des images 3D, l'utilisateur doit segmenter plusieurs coupes 2D à l'aide d'une méthode *live-wire* sur deux plans orthogonaux. Ensuite la segmentation du volume se fait de manière automatique dans le troisième plan en utilisant comme initialisation les intersections de ce plan avec les contours des régions segmentées sur les deux autres plans. Les *live-wire* positionnées manuellement doivent être réparties sur les deux plans et sur toute la taille de la région pour donner les meilleures informations possibles pour les *live-wire* automatiques. Afin d'ordonner les points provenant des *live-wire* de l'utilisateur, un graphe connecté est créé (cf. figure 4.2). Un point initial est choisi arbitrairement puis on parcourt les points afin de construire la segmentation 3D. S'il y a deux graphes non connectés, c'est qu'il y a deux contours disjoints.

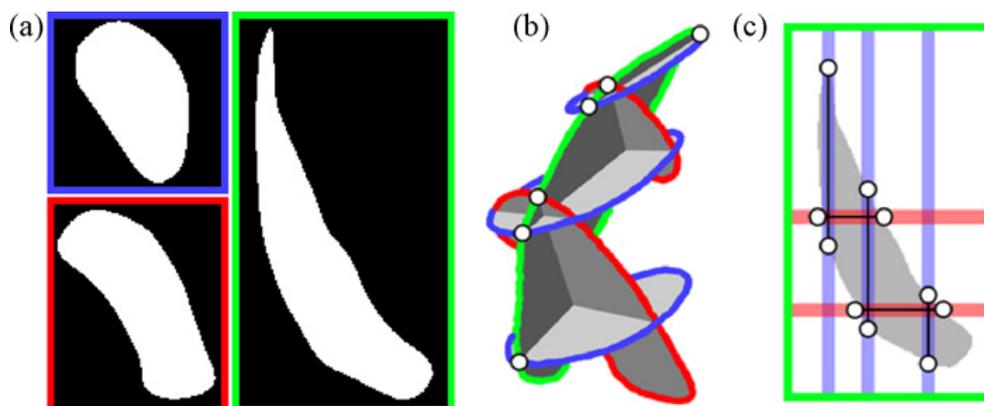


FIGURE 4.2 – Les contours orthogonaux (a) (rouge et bleu) intersectent les coupes dans la troisième dimension (vert). Les points d'intersection sur la troisième dimension sont utilisés comme graines pour obtenir la segmentation en 3 dimensions (l'image est tirée de [74]).

Le *surface dragging* est un autre moyen d'interaction qui peut être associé aux méthodes basées contours. En 2D ou 3D, ce genre de techniques permet de sélectionner un contour d'une segmentation et de le corriger en déplaçant ce contour avec la souris. Dans [51], les auteurs utilisent un VOI (Volume d'intérêt, cf. figure 4.3) autour du curseurs. La surface de la segmentation présente à l'intérieur de l'intersection entre le VOI et la segmentation est la partie qui va pouvoir être déformée. Lorsque l'utilisateur a positionné le VOI, la direction du mouvement de la souris et la taille du déplacement permettent alors de gérer le mouvement de la surface. Un exemple de manipulation est illustré sur la figure 4.3. La surface modélisée après le déplacement du curseur est une courbe spline.

Enfin un dernier moyen pour intervenir sur la segmentation est de modifier les nombreux paramètres qui peuvent être présents dans les techniques de segmentation basées contour.

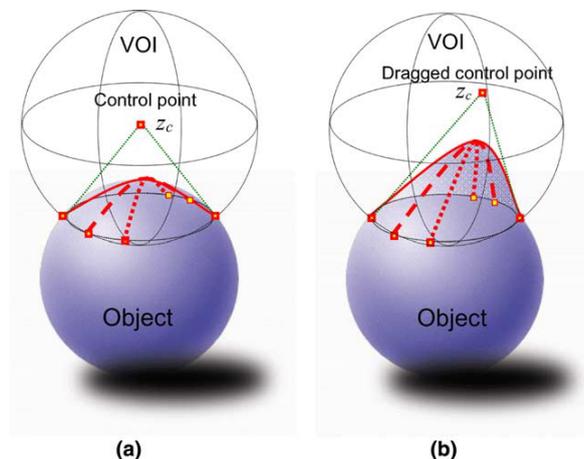


FIGURE 4.3 – Exemple de surface dragging sur une surface sphérique. (a) Représente le point de contrôle avant modification de la forme. (b) Représente la forme de la sphère après déplacement du curseur (l'image est tirée de [51]).

Dans [106] et ITK-SNAP, les auteurs ont créé une interface permettant une segmentation interactive basée sur un *level-set* 3D. L'équation d'évolution de ce *level-set* s'écrit :

$$\frac{\partial}{\partial t}\phi(x;t) = F\Delta\phi \quad (4.4)$$

où ϕ est une fonction de dimension supérieure aux contours ; $\phi(x;t) = 0$ définit une fonction implicite et représente le contour de la région recherchée. La fonction F décrit les forces appliquées sur le contour. Elles se composent d'un terme dépendant du gradient et des intensités de l'image ainsi que d'un terme dépendant de la courbure du contour.

La segmentation se déroule en plusieurs étapes et durant chacune de celles-ci, l'utilisateur peut faire varier certains paramètres afin de guider la segmentation.

- La première étape consiste à construire les cartes de caractéristiques qui vont ensuite guider une segmentation par *level-set*. À partir de l'image réelle, l'utilisateur peut mettre en évidence l'intensité de la région qu'il recherche. À la suite de cette étape, l'image est représentée par une carte de probabilité, la probabilité étant forte là où les voxels ont des intensités proches de celles que l'utilisateur a défini. L'utilisateur peut aussi appliquer des filtres de lissage pour sélectionner les contours intéressants.
- La deuxième étape consiste à positionner les graines formant les contours initiaux. Ces graines sont des volumes sphériques et leurs limites définissent le contour initial du *level-set*.
- Enfin la troisième étape est la gestion de la pondération des paramètres du *level-set*. Ces paramètres peuvent gérer l'importance du gradient de l'image ou de la courbure du contour.

1. <http://www.itksnap.org/pmwiki/pmwiki.php?n=Main.HomePage>

Ces méthodes, peuvent permettre d’obtenir des résultats proches de ce que souhaite l’utilisateur, cependant cela peut nécessiter un temps d’utilisation très important a cause d’un nombre élevé d’interactions potentielles.

4.2 Méthodes basées deep learning

Les méthodes de segmentation par réseaux de neurones convolutionnels (CNN) et par apprentissage profond deviennent de plus en plus courantes et efficaces, y compris pour la segmentation d’images médicales. Les réseaux de type *U-net* apparus il y a quelques années sont actuellement très utilisés pour la segmentation et l’interprétation d’images. D’abord développés dans le cas 2D [24, 78] puis pour le cas 3D [26], les auteurs ont construit une architecture et un *framework* utilisables sur un large type d’images. Cette architecture a la propriété d’être une architecture *end-to-end*, c’est à dire que le réseau construit lui même les caractéristiques nécessaires en plus de produire la décision. L’entrée du réseau est une image et la sortie l’image segmentée correspondante. Une autre propriété de ces méthodes réside dans le fait d’appliquer des mécanisme de génération de données permettant de fonctionner dans le cas où les données d’apprentissage labellisées ne sont pas nombreuses. Des déformations élastiques sont générées sur les images afin d’en créer artificiellement de nouvelles données d’apprentissage.

Récemment, dans [31], un réseau de neurones profonds est utilisé pour une segmentation complète du cerveau. Dans ces travaux, les caractéristiques ne sont pas construites dans le réseau mais sont construites par les auteurs. Celles-ci se composent d’un patch 3D et trois patchs 2D orthogonaux (n’ayant pas la même taille). Ils définissent aussi des caractéristiques globales (cf. figure 4.4) avec trois patchs 2D qui ont été redimensionnés (réduits) d’un certain facteur. S’ajoutent à ceci des informations de positionnement relatif par rapport aux centroides des autres régions. Un premier réseau de neurones est appris sans l’information des relations spatiales par rapport aux centroides permettant d’initialiser une première segmentation. Ensuite, le réseau de neurone possédant les informations spatiales est utilisé plusieurs fois jusqu’à convergence de la segmentation. Même si la base d’apprentissage reste de taille faible et que les réseaux sont sur-appris, les résultats de segmentation sont proches des meilleures méthodes sur les images IRM de cerveaux humains. En effet, l’anatomie du cerveau étant très structurée, la variabilité reste faible. D’autres méthodes ont déjà utilisé les CNNs pour la segmentation d’autres images plus spécifiques comme la segmentation de tumeurs [43] ou la segmentation des tissus sur les images IRM d’enfants [108].

4.3 Autres méthodes de classification des voxels

Pour obtenir une segmentation des structures anatomiques d’un organes et notamment des tissus du cerveau, l’un des moyens les plus courant est d’utiliser des méthodes de classification de voxels. La classification de voxels a pour but d’associer un label choisi parmi

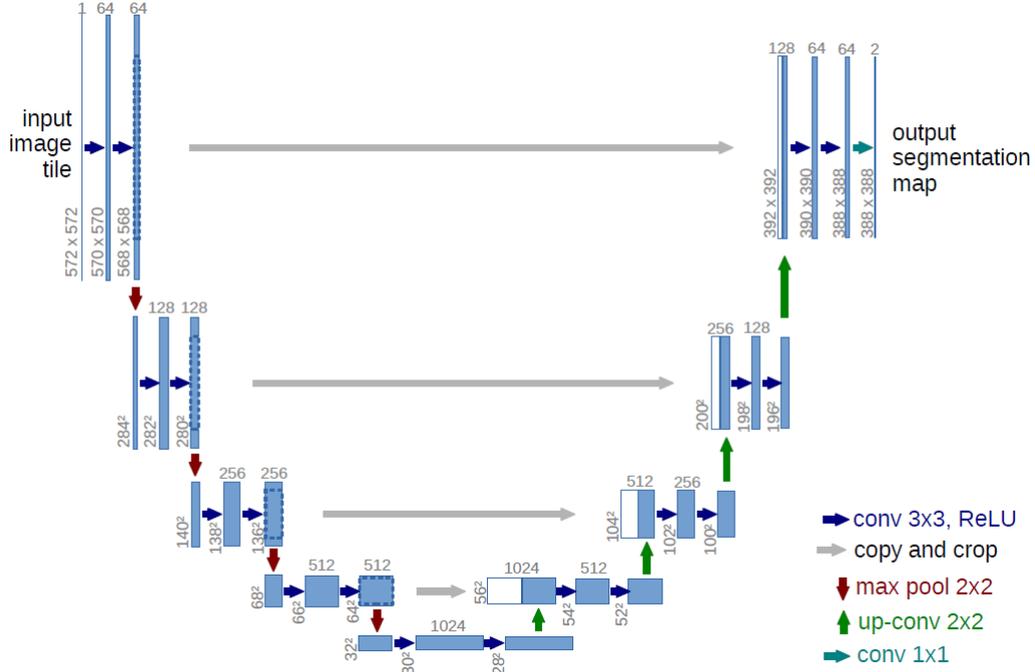


FIGURE 4.4 – Architecture U-net dans le cas 2D pour une image d’entrée 572x572. L’image d’entrée est l’image à segmenter et l’image de sortie est l’image label (*end-to-end*). Le schéma illustre les différentes convolutions et les différents pooling appliqués à l’image (l’image est tirée de [78]).

K classes possibles à chacun des voxels d’une image 3D. La composition et la structure d’un organe étant stables, ce nombre K est généralement connu *a priori*. Il est par exemple égal à 3 lorsque l’on cherche les 3 matières du cerveau (MG, MB et LCR). De plus, il est assez facile d’incorporer dans ces méthodes les informations provenant d’atlas et en particulier d’atlas probabilistes.

4.3.1 K-means

L’algorithme K-means [39] est un algorithme de classification non-supervisée couramment appliqué à la segmentation d’images. Celui-ci vise à partitionner des individus caractérisés par un vecteur de descripteurs en K clusters. L’objectif est de minimiser la distance entre le centroïde des clusters et les voxels appartenant à chacun de ces clusters.

$$\operatorname{argmin}_S \sum_{k=1 \dots K} \sum_{y_i \in S_k} \|\mu_k - y_i\|^2 \quad (4.5)$$

où y_i avec $i = 1, \dots, N$ est l’ensemble des individus à classer, S est la segmentation, S_k avec $k = 1, \dots, K$ sont les K clusters et μ_k est le centroïde du cluster k .

La minimisation de cette fonction est faite de manière itérative en mettant à jours la valeur des centroïdes de chaque cluster puis en assignant chaque individu au centroïde le plus proche. Ces deux étapes sont appliquées jusqu'à convergence lorsque plus aucun individu n'est réaffecté. Le K-means nécessite néanmoins une initialisation qui peut avoir un impact sur le résultat final (minima locaux). Cette initialisation peut être faite en choisissant aléatoirement les centroïdes. Le K-means est régulièrement exploité pour initialiser d'autres méthodes.

4.3.2 C-means

L'algorithme C-means (FCM) découle du K-means et de la logique floue, permettant aux individus d'appartenir à plusieurs clusters selon une certaine probabilité. Ce degré d'appartenance aux différents clusters est attribué à chaque individu en fonction de la distance avec le centroïde. L'objectif est alors de minimiser :

$$\operatorname{argmin}_S \sum_{k=1\dots K} \sum_{y_i \in S_k} \gamma_{ik}^\beta \|\mu_k - y_i\|^2 \quad \text{avec} \quad \sum_{k=1\dots K} \gamma_{ik} = 1 \quad \forall i = 1, \dots, N \quad (4.6)$$

où β est un paramètre modélisant le degré de flou et γ_{ik} est le degré d'appartenance du voxel i à la classe k . Cet algorithme fournit de meilleurs résultats que le simple K-means. De plus de nombreuses méthodes basées sur le FCM ont été développées pour la segmentation du cerveau et des tissus plus particulièrement. Dans [70], les auteurs ont développé une version adaptative du FCM qui incorpore un terme modélisant les inhomogénéités de champ. Dans [23], les auteurs utilisent une régularisation non-local (débruitage non-local [21]) afin d'améliorer la résistance au bruit du FCM. Des informations multimodales pour la segmentation des tissus sont utilisées par [44]. Dans [44, 107], les auteurs utilisent des *Kernel distance* plutôt que la distance euclidienne. Enfin, dans [25], les auteurs utilisent un FCM prenant en compte l'information de voisinage dans la fonction d'énergie.

4.3.3 Mélange de gaussiennes

La segmentation par loi de mélange est une méthode de classification qui a aussi été utilisée dans le cadre des images cérébrales [41]. Dans ce cas, les données sont considérées comme la réalisation d'une variable aléatoire. Si $Z = \{Z_1, \dots, Z_n\}$ est la variable aléatoire cachée correspondant à la segmentation de l'image composée de N voxels, et $Y = \{Y_1, \dots, Y_n\}$ est la variable aléatoire observée correspondant à l'image. Z_i peut prendre les valeurs parmi un nombre fini K d'étiquettes $\{l_1, \dots, l_K\}$, on note $z = \{z_1, \dots, z_n\}$ et $y = \{y_1, \dots, y_n\}$ les réalisations des variables aléatoires Z et Y . La densité de probabilité d'un échantillon i s'écrit alors :

$$p(y_i|\theta) = \sum_{k=1\dots K} \gamma_k p(y_i|z_i = l_k, \theta_k) \quad (4.7)$$

où $p(y_i|z_i = l_k, \theta_k)$ est la densité de probabilité du cluster k pour les caractéristiques du voxel v_i et $\gamma_k = p(z_i = k|\theta_k)$ est le poids du cluster k avec $\sum_{k=1\dots K} \gamma_k = 1$. Si l'on modélise la densité de probabilité de la variable aléatoire par la loi normale, alors on se place dans le cas du mélange de gaussienne et $\theta_k = \{\mu_k, \sigma_k\}$ sont les paramètres de la gaussienne modélisant le k^{eme} cluster. Donc :

$$p(y_i|z_i = l_k, \theta_k) = \frac{1}{(2\pi\sigma_k^2)^{\frac{1}{2}}} \exp\left(-\frac{(y_i - \mu_k)^2}{2\sigma_k^2}\right) \quad (4.8)$$

Dans le cas de la classification des voxels, on cherche à résoudre un problème à données incomplètes et à déterminer le label $z_i \in \{l_1, \dots, l_K\}$ lié au voxel v_i , possédant une intensité y_i . Une fois les paramètres du modèle θ_k estimés, la probabilité *a posteriori* d'appartenance à un cluster peut-être définie par :

$$p(z_i = l_k|y_i, \theta) = \frac{\gamma_k p(y_i|z_i = l_k, \theta_k)}{\sum_{l=1}^K \gamma_l p(y_i|z_i = l, \theta_l)} \quad (4.9)$$

Expectation-Maximization : Pour estimer les paramètres du modèle, on cherche souvent à maximiser la vraisemblance définie telle que :

$$\log(\mathcal{L}(\theta|\mathbf{z}, \mathbf{y})) = \log \prod_{i=1}^N p(y_i|z_i = l_k) p(y_i) = \sum_{i=1}^N \log(\gamma_k p(y_i, z_i = l_k|\theta_k)) \quad (4.10)$$

Une méthode d'optimisation du type Expectation-Maximization (EM) est utilisée pour ajuster les paramètres du modèle [32]. Cette procédure itérative se déroule en deux étapes : une étape d'estimation (E) où l'on calcule l'espérance de la vraisemblance puis une étape de maximisation où l'on calcule les paramètres qui maximisent l'espérance de la vraisemblance. Concrètement, cela se traduit à l'étape q par :

$$\text{(étape E) : } Q(\theta|\theta^{(q)}) = E[\log(p(\mathbf{z}, \mathbf{y}|\theta))|z, \theta^{(q-1)}]$$

$$\text{(étape M) : } \theta^{(q)} = \operatorname{argmax}_{\theta} (Q(\theta, \theta^{(q)}))$$

Lors de l'étape de maximisation, trois paramètres sont mis à jour par :

$$\begin{aligned} \gamma_k^{(q)} &= \frac{1}{N} \sum_{i=1}^N p(z_i = l_k|y_i, \theta^{(q-1)}) \\ \mu_k^{(q)} &= \frac{\sum_{i=1}^N y_i p(z_i = l_k|y_i, \theta^{(q-1)})}{\sum_{i=1}^N p(z_i = l_k|y_i, \theta^{(q-1)})} \\ \sigma_k^{(q)} &= \frac{\sum_{i=1}^N p(z_i = l_k|y_i, \theta^{(q-1)}) (y_i - \mu_k^{(q)})^2}{\sum_{i=1}^N p(z_i = l_k|y_i, \theta^{(q-1)})} \end{aligned}$$

L'algorithme EM est stoppé après un nombre q d'interactions fixées ou lorsque les modifications entre deux itérations sont suffisamment faibles. Le mélange de gaussiennes et l'algorithme sont aussi efficaces dans le cas de données multimodales. Les gaussiennes utilisées sont alors des gaussiennes multivariées.

La classification par mélange de gaussiennes est devenue populaire car elle permet d'incorporer des informations provenant d'atlas probabilistes. Dans [10], les auteurs décrivent une méthode de segmentation utilisant un mélange de gaussiennes pour incorporer différentes informations telles que la carte de probabilités de l'atlas, une régularisation spatiale ou une prise en compte des inhomogénéités de champ. Le moyen utilisé ici pour incorporer les informations de l'atlas est de pondérer l'importance des classes *via* le paramètre γ_k . La probabilité $p(y_i|\theta)$ peut s'écrire :

$$p(y_i|\theta) = \sum_{k=1\dots K} \frac{\gamma_k p_i(k)}{\sum_{l=1}^K \gamma_l p_i(l)} p(y_i|z_i = k, \theta_k) \quad (4.11)$$

avec $p_i(k)$ la probabilité que le voxel v_i appartienne à la classe k d'après l'atlas probabiliste. Un voxel par lequel cette probabilité est élevée d'avoir une certaine classe d'après les informations de l'atlas ($p_i(k) \simeq 1$) obtiendra une densité de probabilité élevée.

Le mélange de gaussiennes (ainsi que le K-means et le FCM) repose sur l'hypothèse que les voxels sont indépendants les uns des autres ce qui n'est pas le cas dans la réalité. Il est possible d'ajouter des termes de régularisation ou alors d'utiliser des modèles Markoviens qui incorporent des informations de voisinage.

4.3.4 Chaines de Markov cachées

Dans [19], les auteurs utilisent des chaînes de Markov cachées pour la segmentation d'images médicales. Une chaîne de Markov est une suite d'état, où l'état à un instant donné ne dépend que de l'état précédent. Une chaîne de Markov est une séquence unidimensionnelle d'état tandis que les images réelles proviennent d'un repère deux ou trois dimensions. Pour décrire l'image sous la forme d'une suite unidimensionnelle, un parcours est défini à l'intérieur de l'image. La distance entre deux voxels dans l'image doit être le plus reliée à la distance de ces deux voxels dans la chaîne de Markov. Le parcours de type Hilbert-Peano, permet de conserver une information de voisinage aussi bonne que possible. Dans le cas de la segmentation d'images, des chaînes de Markov dites cachées sont utilisées. Les états de la chaîne sont alors inconnus et correspondent aux régions que l'on veut détecter et seules les observations provenant de l'image sont disponibles.

De même que dans la section précédente $Z = \{Z_1, \dots, Z_n\}$ et $Y = \{Y_1, \dots, Y_n\}$ sont les variables aléatoires représentant la segmentation et l'image observé. Y est l'ensemble des variables observées et Z l'ensemble des variables cachées que l'on cherche déterminer. Z est une chaîne de Markov alors si :

$$p(Z_{n+1} = l_{n+1} | Z_n = l_n, \dots, Z_1 = l_1) = p(Z_{n+1} = l_{n+1} | Z_n = l_n) \quad (4.12)$$

Ce qui se traduit par, l'état d'un nœud de la chaîne ne dépend que de l'état du nœud précédent. De même que dans les mélanges de gaussiennes la probabilité $p(Y_n = y_n | Z_n = k)$ est modélisée par une gaussienne de moyenne μ_k et de matrice de covariance Σ_k .

Le but étant de trouver Z en déterminant les paramètres de la chaîne de Markov cachée, ces paramètres se définissent par $\pi_k = p(Z_1 = l_k)$ pour la distribution initiale (du premier voxel) et $t_{kl}^n = p(Z_{n+1} = l_l | Z_n = l_k)$ pour la transition de l'état k à l'état l .

L'observation Y_n est considérée conditionnellement indépendante à X et la probabilité conditionnelle de Y_n sachant Z est égale à la probabilité d'avoir Y_n sachant Z_n . Ces deux propriétés mènent à :

$$p(Y|Z) = \prod_{n=1}^N p(Y_n = y_n | Z_n = l_k) \quad (4.13)$$

L'estimation des paramètres de la chaîne de Markov cachée est généralement aussi faite à l'aide d'un algorithme de type EM. Pour cela, il faut pouvoir évaluer la probabilité d'obtenir une certaine séquence d'observations sachant un modèle fixé. Il est difficile d'un point de vue combinatoire d'évaluer cette probabilité de manière brute. L'une des solutions est d'utiliser une méthode de type progressif-rétrogressif (*forward-backward*) [76]. La probabilité *forward* $\alpha_n(k)$ et la probabilité *backward* $\beta_n(k)$ sont définies telles que :

$$\alpha_n(k) = p(z_n, y_{\leq n} = (y_1, \dots, y_n)) = \sum_{l=1}^K \alpha_{n-1}(l) t_{lk} f_k(y_n) \quad (4.14)$$

$$\beta_n(k) = p(y_{>n} = (y_{n+1}, \dots, y_N) | z_n) = \frac{1}{N_{n+1}} \sum_{l=1}^K \beta_{n+1}(l) t_{kl} f_k(y_{n+1}) \quad (4.15)$$

où $f_k(y_n)$ est la probabilité d'obtenir l'observation y_n sachant que l'on se trouve dans l'état k . Il est alors possible de calculer récursivement ces probabilités de manière moins coûteuse. Ces deux probabilités permettent de définir la probabilité marginale *a posteriori* ainsi que la probabilité jointe *a posteriori* qui sont nécessaires lors de la mise à jour des paramètres. la probabilité marginale *a posteriori* $\gamma_n(i)$ définit la probabilité d'être dans l'état l_i à l'étape n . la probabilité jointe *a posteriori* $\xi_n(i, j)$ définit la probabilité d'être dans l'état k à l'étape $n - 1$ et dans l'état l à l'étape n .

$$\gamma_n^{[q]}(k) = p(z_n | y, \phi^{[q]}) = \frac{\alpha_n(k) \beta_n(k)}{\sum_l \alpha_n(l) \beta_n(l)} \quad (4.16)$$

$$\xi_n^{[q]}(k, l) = p(z_{n-1}, z_n | y, \phi^{[q]}) = \alpha_{n-1}(k) t_{kl} f_l(y_n) \beta_n(l) \quad (4.17)$$

Lors de l'étape d'*expectation*, l'estimation de la vraisemblance peut être exprimée en fonction des deux précédentes probabilités. Lors de l'étape de maximisation, on peut mettre à jour les paramètres de la chaîne de Markov cachée ($\pi_k^{[q+1]}$ et $t_{kl}^{[q+1]}$) mais aussi les paramètres des gaussiennes modélisant l'observation liée à chaque état. A l'étape q :

$$\pi_k^{[q+1]} = \gamma_1^{[q]}(k) \quad (4.18)$$

$$t_{kl}^{[q+1]} = \frac{\sum_n \xi_n^{[q]}(k, l)}{\sum_n \gamma_n^{[q]}(k)} \quad (4.19)$$

$$\mu_k^{[q+1]} = \frac{\sum_n \gamma_n^{[q]}(k) y_n}{\sum_n \gamma_n^{[q]}(k)} \quad (4.20)$$

$$\sigma_k^{2[q+1]} = \frac{\sum_n \gamma_n^{[q]}(k) (y_n - \mu_k)^2}{\sum_n \gamma_n^{[q]}(k)} \quad (4.21)$$

Dans le cas où l'on possède les informations provenant d'un atlas probabiliste, les auteurs utilisent une chaîne de Markov "triplet" pour inclure ces nouvelles informations. On cherche à maximiser la probabilité $P(Z_n = l_k | Y, A)$ où A représente la variable aléatoire modélisant l'atlas. La méthode pour optimiser les paramètres de la chaîne de Markov cachée sont modifiés dans le calcul des probabilités *forward* et *backward*. On note $p_n(l)$, la probabilité provenant de l'atlas que le voxel n appartienne à la classe l .

Le calcul des probabilités *forward* et *backward* est modifié tel que :

$$\alpha_n(k) = \sum_{l=1}^K \alpha_{n-1}(l) t_{lk} f_k(y_n) p_n(k) \quad (4.22)$$

et

$$\beta_n(k) = \frac{1}{N_{n+1}} \sum_{l=1}^K \beta_{n+1}(l) t_{kl} f_k(y_{n+1}) p_{n+1}(l) \quad (4.23)$$

Une fois les paramètres optimisés, il faut déterminer la séquence d'états la plus probable sachant les observations et le paramètre du modèle. L'algorithme de Viterbi [97] est aussi un algorithme récursif permettant de résoudre ce problème. Il suffit ensuite de retransformer la séquence 1D vers le format de l'image pour obtenir la segmentation finale.

4.3.5 Champs de Markov Cachés

Les champs de Markov cachés (que l'on nommera par la suite simplement champ de Markov ou HMF) sont des modèles qui respectent la propriété Markovienne sur un graphe non orienté. Ils ont eux aussi été utilisés dans le cadre de la segmentation d'images médicales [109]. Dans un cas classique, le graphe non orienté G est défini tel que les nœuds représentent les voxels et les arcs une informations de voisinage entre ces voxels. Il faut définir une notion de voisinage \mathcal{N} (clique). Dans le cas du champs de Markov, Z est un champ si :

$$\forall z, p(z_i | z_{V \setminus \{i\}}) = p(z_i | z_{\mathcal{N}(i)}) \quad \text{et} \quad p(x) > 0$$

Le terme $V \setminus \{i\}$ représente l'ensemble des nœuds du graphe sans le nœud i . La probabilité d'avoir un nœud i dans l'état z_i ne dépend que des nœuds dans le voisinage $\mathcal{N}(i)$.

Le théorème de Hammersley-Clifford [53] a montré que si Z est un champ de Markov alors la probabilité $p(z)$ est une distribution de Gibbs et peut s'exprimer par :

$$p(z) = W^{-1} \exp(-H(z)) \quad (4.24)$$

où W est un fonction de normalisation et H est une énergie associée aux cliques qui peut s'écrire comme une somme de potentiels :

$$H(z) = \sum_{c \in \mathcal{C}} U_c(z_c) \quad (4.25)$$

Si on conserve les mêmes notations avec $Y = \{Y_1, \dots, Y_N\}$ l'image observée et $Z = \{Z_1, \dots, Z_N\}$ l'image label que l'on recherche. On cherche là encore à trouver la segmentation qui maximise la probabilité $p(z|y, \phi)$ avec ϕ les paramètres du modèle. D'après l'équation 4.24 et que l'on suppose les observations Y conditionnellement sachant Z , alors on peut écrire :

$$H(z|y, \phi) = H(z|\phi_z) - \sum_{i \in V} \log(p(y_i|z_i, \phi_y)) \quad (4.26)$$

avec ϕ_z les paramètres du modèle lié au champ de Markov et ϕ_y les paramètres du modèle d'intensité. Le terme $H(z|\phi_z)$ peut s'exprimer comme une somme de potentiels sur l'ensemble du volume tel que :

$$H(z|\phi_z) = - \sum_{i \in V} (U_i(z_i) + \frac{\eta}{2} \sum_{j \in \mathcal{N}(i)} U_{ij}(z_i, z_j)) \quad (4.27)$$

avec $U_i(z_i)$ le potentiel du voxel v_i . Ce terme permet d'incorporer le champs externe apportant les informations de l'atlas. Si $p_i(z_i)$ est la probabilité d'appartenance à la classe x_i provenant de l'atlas probabiliste, alors on peut fixer $U_i(z_i) = f(p_i(z_i))$. Le terme $U_{ij}(z_i, z_j)$ représente la corrélation spatiale entre le voxel i et son voisinage. η permet de moduler l'importance relative de cette information. Le modèle de Potts simple décrit $U_{ij}(z_i, z_j)$ tel que :

$$U_{ij}(z_i, z_j) = \delta(z_i, z_j) \quad (4.28)$$

où $\delta(z_i, z_j)$ est la fonction qui vaut 1 si z_i et z_j sont égaux, et 0 sinon. La modélisation de l'intensité $p(y_i|z_i, \phi_y)$ peut quant à elle toujours se décrire sous forme d'une gaussienne potentiellement multi-variée, comme pour le mélange de gaussiennes et les chaînes de Markov cachée. La segmentation peut aussi se faire par maximum *a posteriori* et ainsi déterminer les paramètres ϕ maximisant la probabilité $p(x|y, \phi)$. Un algorithme de type EM ou ICM (Iterated Conditional Modes [12]) peut être utilisé par cette optimisation. La modélisation par champ de Markov permet d'obtenir une régularisation spatiale par l'intermédiaire de l'influence des voxels du voisinages.

4.4 Fusion de données

Il est courant en imagerie médicale d'avoir recours à des méthodes de fusion de données afin de pouvoir prendre une décision en utilisant de manière efficace toutes les informations

disponibles. Cette situation peut se produire dans le cas d'images multi-modales où il est nécessaire d'utiliser l'ensemble des images à disposition ; mais aussi dans le cas de la segmentation multi-atlas, où la qualité de la segmentation dépend de la qualité de la fusion des données provenant de chaque atlas. Nous allons décrire les méthodes les plus importantes de fusion d'information.

4.4.1 Méthode de vote

De manière générale, les segmentations basées sur des atlas passent par une étape de propagation de label. Si I est l'image que l'on veut segmenter ; A est un atlas topologique simple composé d'un template T (une image réelle) et L une image labellisée correspondante ; si τ est une transformation qui permet de relier les voxels v de l'espace de l'atlas A vers l'espace de l'image I ; alors la propagation de label peut s'exprimer par :

$$S(v) = L(\tau(v)) \quad (4.29)$$

avec S la segmentation de l'image I . Les paramètres qui peuvent influencer cette propagation de label proviennent de la transformation τ et de la méthode de recalage permettant de l'obtenir (cf. 3.2).

Cette propagation de label est utilisée dans le cas d'atlas avec un seul sujet (un seule couple d'image) mais peut aussi être utilisé dans le cas d'une méthode multi-atlas. La méthode la plus simple pour obtenir une segmentation à partir de plusieurs atlas est la méthode de vote. Si $A = \{(T_i, L_i) | i \in (1, \dots, N)\}$ est un ensemble d'atlas composée de N couples d'image template T_i et d'image labellisée correspondante L_i , alors la méthode de vote s'exprime par :

$$S(v) = \operatorname{argmax}_{l \in L} \sum_{a=1}^N \delta(L_a(\tau_a(v), l)) \quad (4.30)$$

où τ_i est la transformation de l'atlas $A_i = (T_i, L_i)$ vers l'image I ; $\delta(x, y)$ est la fonction qui vaut 1 si $x = y$ et 0 sinon ; L est le nombre de labels possibles. Il peut être intéressant de sélectionner une sous partie $N' \leq N$ des atlas disponibles pour la méthode de vote. Cette sélection a tendance à améliorer les résultats de segmentation en ne prenant pas en compte les atlas très différents de l'image à segmenter. Elle permet aussi de réduire le temps de segmentation *via* une réduction du temps de recalage. Cette sélection peut être faite *via* le calcul d'une similarité entre l'image à segmenter et les différents atlas [8, 59]. Par exemple, un recalage rapide (rigide, affine, etc.) est utilisé sur tous les atlas et les N' atlas où la similarité est la plus forte sont sélectionnés. Un recalage non linéaire plus précis (et plus long) leur est appliqués avant de procéder à la méthode de vote.

Une extension de la méthode de vote est la méthode de vote pondéré. Un poids w_i peut être affecté à chaque atlas dépendant de la similarité entre l'atlas A_i recalé et l'image I . Les poids des atlas sont normalisés tel que $\sum_{i=1}^N w_i = 1$. La pondération peut être faite de manière globale [103], dans ce cas le poids w_i est constant pour tous les voxels v de

l'atlas A_i . Si $Sim(I_1, I_2)$ est une mesure de similarité entre l'image I_1 et l'image I_2 , alors la segmentation de l'image I peut s'exprimer par :

$$w_i = \frac{1}{Sim(T_i, I)} \quad (4.31)$$

$$S(v) = \operatorname{argmax}_{l \in L} \sum_{i=1}^N w_i \delta(L_i(\tau_i(v), l)) \quad (4.32)$$

La pondération peut aussi être locale, dans ce cas w_i dépend du voxel v_i [48] (ou potentiellement d'un patch autour de v_i). Cette pondération est fonction de la différence d'intensité entre le template T_i et l'image I sur un patch autour de voxel v . Cette pondération est généralement inversement proportionnelle à la métrique choisie tel que :

$$w_i(v) = \frac{1}{Sim(T_i(v), I(v))} \quad (4.33)$$

$$S(v) = \operatorname{argmax}_{l \in L} \frac{1}{\sum w_i(v)} \sum_{i=1}^N w_i(v) \delta(L_i(\tau_a(v), l)) \quad (4.34)$$

Dans le cadre des segmentations multi-atlas, dans [83], les auteurs proposent un modèle probabiliste général de fusion de labels. Les méthodes de segmentation multi-atlas avec méthode de vote (non-pondéré, pondéré localement ou globalement) peuvent s'exprimer comme des cas particuliers de ce modèle. Le modèle considère que l'image à segmenter (I, S) est générée à partir d'une ou plusieurs images de la base d'apprentissage. La figure 4.5 illustre le graphiquement le modèle utilisé. La segmentation a pour but d'estimer l'image label S qui maximise le maximum *a posteriori* défini tel que :

$$\mathcal{L} = \operatorname{argmax}_L p(L|I; T_1, \dots, T_N, L_1, \dots, L_N)$$

De plus, les auteurs dans [83] utilisent un champ aléatoire M qui associe à chaque voxel, l'index de l'atlas à partir duquel il a été généré. Si on suppose que $I(v)$ et $L(v)$ sont conditionnellement indépendants, alors il est possible d'écrire :

$$\mathcal{L} = \operatorname{argmax}_L \sum_M P(M) \prod_{v \in X} p(I(v)|M(v); I_{M(v)}) \times p(L(v)|M(v); L_{M(v)}) \quad (4.35)$$

Chaque terme de l'équation peut ensuite être instancié pour obtenir un certain modèle de segmentation. Le terme lié à la vraisemblance de l'image peut être représenté par une distribution gaussienne telle que :

$$p(I(v)|M(v) = a; I_a) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2}(I(v) - T_a(v))^2\right] \quad (4.36)$$

4.4. FUSION DE DONNÉES

Le terme lié aux labels peut être exprimé à l'aide d'une distance signée aux labels définie dans [73]. Elle peut s'écrire telle que :

$$p(L(v) = l | M(v) = a; L_a) = \frac{1}{Z_{a\rho}(v)} \exp(\rho \mathcal{D}_a^l) \quad (4.37)$$

où \mathcal{D}_a^l est la distance signée au label l dans l'atlas a , ρ est un paramètre qui module la distance signée, et Z est la constante de normalisation.

Enfin un *a priori* de type champ de Markov caché est utilisé sur le champ M . La probabilité $p(M)$ est définie telle que :

$$p(M) = \frac{1}{Z_\beta} \prod_v \exp(\beta \sum_{v' \in \mathcal{N}_v} \delta(M(v), M(v'))) \quad (4.38)$$

Les paramètres σ et β permettent de moduler la méthode de fusion. Si $\beta = 0$, alors le champ $M(v)$ est le même pour tous les labels et pour tous les voxels v . La segmentation de L correspond à une méthode de vote pondérée localement par la vraisemblance liée à l'image. Si β tend vers l'infini, $p(M)$ est différent de 0 seulement si le champ $M(v)$ est la même pour tous les voxels v . Cela mène à une méthode de vote à pondération globale. Si σ tend vers l'infini alors la vraisemblance liée à l'image est une constante et la même pour toutes les atlas. Cela mène à une méthode de vote majoritaire où la pondération de chaque atlas est la même. La résolution de ce problème se fait à l'aide d'un algorithme de type EM. Ce modèle peut aussi être étendu au cas multi-modal ou aux méthodes basées patches [47].

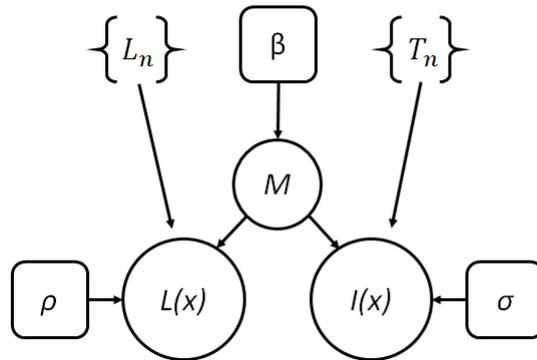


FIGURE 4.5 – Schéma du modèle génératif. I_n est l'image template de l'atlas n° n , L_n est l'image label correspondante. I est l'image à segmenter et L est l'image label que l'on cherche à déterminer. M est un champ aléatoire qui associe à chaque voxel de l'image L , l'index de l'atlas à partir duquel il a été généré. σ est un paramètre qui module l'importance de la ressemblance entre le template T_n et l'image I . β est un paramètre qui module le choix du champs M (variation entre une pondération totalement globale et totalement locale). ρ est un paramètre qui modélise l'importance de la distance entre les labels des atlas et de l'image à segmenter I (l'image est tirée de[83]).

4.4.2 Joint Label Fusion

La méthode de *Joint Label Fusion* (JLF) décrite dans [99] est la méthode de segmentation multi-atlas qui a fourni les meilleurs résultats lors du challenge MICCAI'12 [57] (segmentation multi-atlas complète du cerveau). Elle se base sur la méthode de vote pondérée localement en utilisant une pondération prenant en compte la corrélation entre les différents atlas. Si un groupe d'atlas fournit le même type d'information alors le poids donné aura tendance à agir comme s'il n'y avait qu'un seul atlas. Pour cela, les poids locaux sont définis de manière à minimiser l'espérance de l'erreur de labellisation avec la méthode de vote pondéré et la segmentation de l'image I .

$$M_v(a, b) = p(\delta_{err}^i(v)\delta_{err}^j(v))1|I, T_a, T_b) \quad (4.39)$$

où $M_v(a, b)$ est une matrice de dépendance exprimant la probabilité que l'atlas a et l'atlas b fournissent la même erreur de label au voxel v . Les auteurs montrent ainsi que si on cherche à minimiser l'espérance de la différence entre le label provenant de la méthode de vote pondéré et la segmentation de l'image I , alors on peut exprimer la pondération des atlas en fonction de la matrice M_v tel que :

$$\mathbf{w}_v = \frac{M_v^{-1}\mathbf{1}_n}{\mathbf{1}_n^t M_v^{-1} \mathbf{1}_n} \quad (4.40)$$

avec $\mathbf{1}_n$ un vecteur unitaire de taille n . \mathbf{w}_v est le vecteur de pondération des atlas au voxel v . L'estimation de la matrice M_v peut être faite à l'aide d'un modèle similarité locale entre les atlas et l'image à segmenter. Cette mesure de similarité est calculée à partir de différentes métriques. Dans le cas de la différence absolue en intensité, M_v est décrit telle que :

$$M_v(a, b) = |T_a(\mathcal{N}(v)) - I(\mathcal{N}(v))| \times |T_b(\mathcal{N}(v)) - I(\mathcal{N}(v))|^\beta \quad (4.41)$$

où \mathcal{N}_v est un patch local centré en v et β est un paramètre permettant de moduler l'aspect uniforme (ou non uniforme) des poids. Cette méthode permet d'augmenter significativement la qualité des résultats de segmentation par rapport aux méthodes de vote simple vues précédemment.

Les méthodes multi-atlas obtiennent de très bons résultats pour un grand nombre de problème de segmentation d'images cérébrales. Elles sont générales et peuvent s'appliquer à de nombreux problèmes différents. Cependant celles-ci nécessitent le recalage d'un grand nombre d'images pouvant être couteux en temps. Elles restent aussi entièrement automatiques et il est difficile pour l'utilisateur d'y incorporer des informations.

4.4.3 Fusion floue

Les relations spatiales floues, les atlas probabilistes ou la modélisation de l'intensité des images peuvent fournir des informations de types probabilistes. La fusion de ce type d'infor-

mations est donc pertinente dans le cadre de la segmentation des images médicales. Suivant le contexte, il existe un nombre important de moyens pour fusionner ce type d'information [14].

On peut répartir les opérateurs de fusion en trois classes : les opérateurs conjonctifs, les opérateurs disjonctifs et les compromis. Les opérateurs conjonctifs permettent de réduire l'espace et sont définis tel que $g(y_1, y_2) \leq \min(y_1, y_2)$, avec y_1 et y_2 deux informations floues. Il faut que les deux conditions soient satisfaites pour que l'opérateur fournisse des informations positives. Les opérateurs disjonctifs font l'opposé et sont définis tels que $g(y_1, y_2) \geq \max(y_1, y_2)$. Ils ont tendance à augmenter la certitude finale. Les opérateurs de type compromis sont des opérateurs qui fournissent des résultats intermédiaires.

Dans le cas de l'utilisation des relations spatiales floues, les fusions de types conjonctives sont le plus utilisées car l'espace flou défini pour une région est souvent plus grand que l'espace de la région réelle. La fusion de toutes les relations spatiales permettent alors de préciser l'espace de recherche.

4.4.4 Théorie des croyances

La théorie de Dempster Shafer (ou théorie des croyances) est une autre méthode de fusion d'information [86]. Celle-ci permet de modéliser l'incertitude et mais aussi l'imprécision liée aux connaissances.

Il est possible d'allouer une croyance à un ensemble ou un sous ensemble de réponse, contrairement au cadre bayésien qui ne peut agir que sur des singletons. La théorie des croyances est appliquée sur un ensemble de parties $\Omega = \{\omega_1, \dots, \omega_N\}$ appelées cadre de discernement. Une fonction de masse est une fonction qui assigne à un sous ensemble de Ω une valeur de croyance telle que :

$$m : 2^\Omega \rightarrow [0, 1] \quad \text{et} \quad \sum_{A \subseteq \Omega} m(A) = 1 \quad (4.42)$$

A est un ensemble et $m(A)$ est la valeur de confiance que l'on a strictement dans A . L'ignorance totale est caractérisée par la fonction de masse $m(\Omega)$. On peut définir ensuite la croyance $Bel(A)$ qui représente la croyance totale que l'on a pour A . Et la plausibilité $Pl(A)$ liée à A qui représente la croyance maximale que l'on peut avoir en A

$$Bel(A) = \sum_{B_i \subset A} m(B_i) \quad \text{et} \quad Pl(A) = \sum_{B_i \cap A} m(B_i) \quad (4.43)$$

Il est possible de combiner les croyances définies sur un même cadre de discernement. La combinaison de ces croyances permet de fusionner des informations provenant de différentes sources. Dans le cas de la combinaison conjonctive la fonction de masse associée à un élément $A \subset \Omega$ est définie telle que :

$$m_{1 \cap 2}(A) = \sum_{B \cap C = A} m_1(B)m_2(C) \quad \forall A \subseteq \Omega \quad (4.44)$$

4.5. MÉTHODES BASÉES SUR DES ATLAS EXPLOITANT DES INFORMATIONS LOCALES

La prise de décision, sachant l'espace de discernement et les différentes sources d'information, peut se faire par maximum de croyance *Bel*. Il peut être réduit aux hypothèses simples (singletons) ou non. Cette méthode a, par exemple, été utilisée dans [14] pour la segmentation des tissus sur des images IRM multimodales. Dans ce cas présent, les fonctions de masse sont construites à partir des histogrammes de l'image. Les opérations de combinaison permettent de fusionner les informations provenant de deux modalités différentes.

L'avantage de la théorie des croyances, c'est qu'elle permet de moduler l'importance de chaque source à l'aide d'une ignorance globale. De plus, la prise de décision en faveur d'éléments composés, tels que l'union de classes, permet la prise en compte des effets de volume partiel présent dans les images. Cette méthode de fusion de données apporte un cadre plus riche et plus flexible que le cadre probabiliste de base. Il peut cependant devenir complexe (en terme de calcul) lorsque le nombre de classes devient important.

4.5 Méthodes basées sur des atlas exploitant des informations locales

Certains travaux ont utilisé des méthodes basées atlas tout en y ajoutant des information de localité. Ces méthodes définissent des volumes d'intérêt dans l'image qui peuvent avoir ou non un sens anatomique. Ces méthodes agissent différemment sur ces sous-volumes dans le but de sélectionner des informations plus pertinentes ou de les utiliser plus rapidement.

Dans [88] les auteurs construisent des atlas probabilistes sur des sous parties du cerveau. Le but est ici de segmenter les tissus du cerveau à partir d'une base d'apprentissage de N images IRM $T_{i \in (1, \dots, N)}$ et images labellisés correspondantes $L_{i \in (1, \dots, N)}$. Un algorithme de partage des eaux est appliqué sur le cerveau afin de parcelliser le cerveau en Z différentes zones. Un cerveau moyen ainsi que sa segmentation associée sont aussi créés. Ensuite, chaque zone est extraite de chacune des images de la base d'apprentissage. Pour chaque zone $z \in [1, \dots, Z]$, N images IRM et N images labellisés sont alors disponibles. Un clustering est effectué sur les sous-images extraites pour chaque zone. Quatre clusters sont alors détectés et l'image *exemple* du cluster est conservée E_{zk} avec $k \in (1, 2, 3, 4)$.

La figure 4.6a illustre, pour une région, les quatre clusters et l'exemple E_{zk} associé à chaque cluster. Une carte de probabilité de présence des tissus est créée pour chaque exemple de chaque zone noté L_{zk} . Celle-ci est créée en moyennant les images du même cluster (avec une pondération représentant sa similarité avec l'image exemple du cluster). L'ensemble des données d'apprentissage est appelé *multi region multi reference atlas*. Un exemple pour une région est illustré sur la figure 4.6b.

La segmentation s'opère donc aussi par zone. L'image que l'on veut segmenter est parcellisée par recalage puis une propagation de label du cerveau moyen est appliquée. Sur chaque zone, l'exemple ressemblant le plus à la zone est sélectionné et assemblé pour faire un atlas complet du cerveau. Enfin, la segmentation de l'image est effectuée par une méthode de type *joint registration-segmentation* [72] (la déformation et la segmentation de

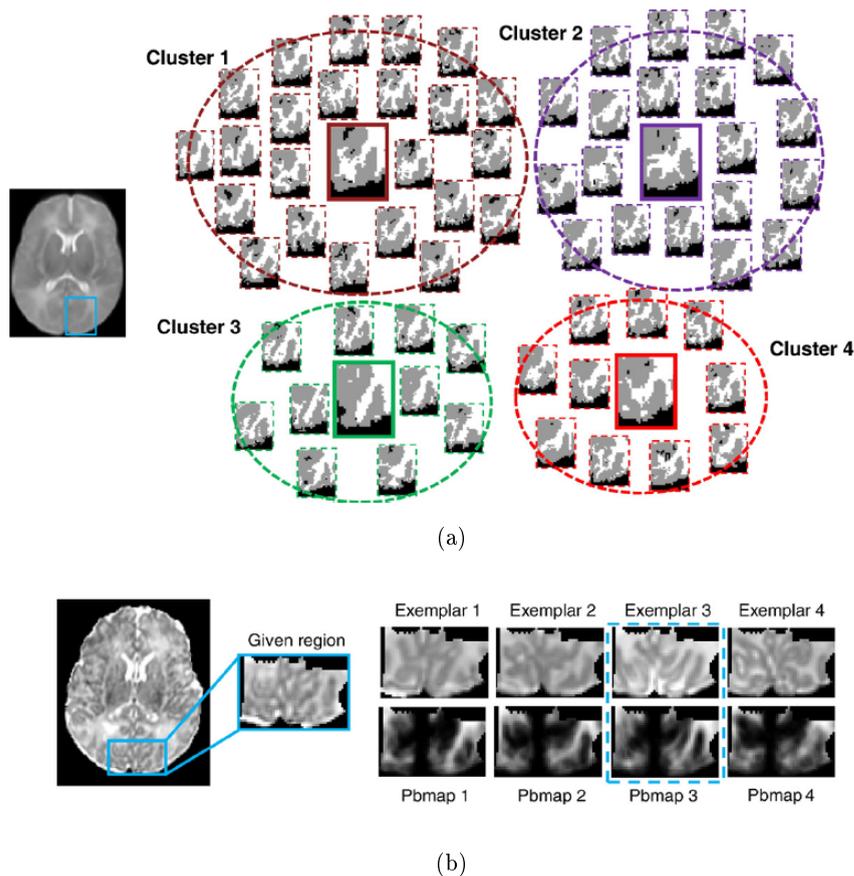


FIGURE 4.6 – Creation du multi region multi reference atlas. La figure (a) montre un exemple de 4 clusters utilisés pour une des région du cerveau. La figure (b) montre un exemple des 4 templates et cartes de probabilités pour un région donnée (l’image est tirée de [88]).

l’image sont faites conjointement de manière itérative au sein d’un algorithme EM).

Dans [95], des images de cœurs ont aussi été segmentées à l’aide d’une méthode multi-atlas appliquée par morceaux. Tous les atlas A_i sont d’abord recalés rapidement (affine) vers l’image à segmenter I . L’atlas le plus proche de l’image à segmenter est choisi comme atlas de référence noté A_r . Cet atlas de référence est divisé en z parties et chaque partie est notée A_{rj} . Ces différentes parties sont définies de manière arbitraire. Elles sont ensuite propagées sur tous les autres atlas ainsi que sur l’image à segmenter notée I_j . Chaque partie de l’image va alors être mise à jour itérativement. L’image label est notée S et les sous parties sont nommées S_j . Dans un premier temps, la partie de l’image j , où A_{rj} est la moins ressemblante à I_j est choisie comme la première partie à mettre à jour. Par la suite, les parties où le nombre potentiel de voxels qui peuvent être modifiés est le plus fort sont sélectionnées. En effet, lors de la segmentation multi-atlas, il est possible que suffisamment d’atlas aient déjà propagé un label pour que celui soit modifiable lors de l’ajout d’un nouvel

4.5. MÉTHODES BASÉES SUR DES ATLAS EXPLOITANT DES INFORMATIONS LOCALES

atlas. Ensuite, l'atlas qui met à jour la partie j est celui dont la différence voxel à voxel est la plus faible. Un recalage plus précis est ensuite effectué sur la sous partie j et les labels sont propagés sur la segmentation S_j . Un critère d'arrêt est fixé lorsqu'il n'y a plus aucune partie où le nombre potentiel de voxels qui peut changer de label est inférieur à un certain seuil θ exprimé en pourcentage. Cette méthode a été testée pour la segmentation des noyaux caudés. Les auteurs montrent que les résultats sont semblables à la méthode multi-atlas non local mais reste moins coûteuse en terme de temps de calcul.

Enfin une autre méthode utilisant un caractère local par morceaux est présentée dans [85]. Dans cette méthode, les auteurs utilisent de multiples champs de Markov coopérant entre eux pour fournir une segmentation. L'image est d'abord subdivisée en z sous parties et chaque sous partie est décrite par un champ de Markov local M_c^T segmentant les trois tissus. Ensuite, la coopération entre les modèles de Markov se traduit par une phase de correction de modèle lors de la phase de maximisation de l'algorithme EM. Pour cela un modèle moyen \tilde{M}_c^T est créé pour chaque tissu T et pour chaque sous partie z comme une moyenne des modèle provenant des sous parties voisines de z . La moyenne est pondérée en fonction de la distance entre la sous partie z et ses voisines. La figure 4.7 décrit la division du cerveau en plusieurs champs de Markov locaux.

Les paramètres des modèles sont alors corrigés comme une moyenne pondérée par un paramètre κ compris entre 0 et 1. Ce paramètre κ est dépendant de la distance entre les paramètres de M_c^T et de \tilde{M}_c^T . Si la différence est trop importante alors κ est fixé à 1 pour corriger complètement les paramètres du modèle M_c^T par \tilde{M}_c^T . Si la différence est faible alors les paramètres du modèle sont inchangés. Enfin, si la différence est raisonnable alors le paramètre κ prend une valeur appartenant à $]0, 1[$. Pour finir, à l'aide d'une interpolation de types "splines" on peut définir un modèle spécifique pour chaque voxel à l'aide de tous les modèles M_c^T corrigés. Chaque voxel et chaque tissu est alors décrit par une gaussienne.

En plus de la segmentation des tissus, des champs de Markov sont ajoutés pour la segmentation des structures. Chaque structure est définie par un champ de Markov binaire. Le champ de Markov des structures est influencé par le champ de Markov des tissus par l'intermédiaire d'un modèle d'intensité. La probabilité d'avoir tel intensité sachant la structure est la probabilité d'avoir cette intensité sachant le tissu. Inversement le champ de Markov des tissus est influencé par le champ de Markov des structure par l'intermédiaire d'un champ externe (comme pour un atlas).

Comme le montre la figure 4.7, la segmentation est faite en mettant à jour itérativement tous les paramètres du modèle. Lorsque le champ de Markov lié à une structure est optimisé, il active les champs de Markov liés aux tissus et inversement jusqu'à convergence. On peut aussi remarquer que dans ces travaux [85], la segmentation des structures est aussi guidée par des informations *a priori* de type relation spatiale floues.

Ces méthodes de segmentation plus local ont montré plusieurs avantages. Elle peut permettre d'obtenir de meilleurs résultats que les méthodes où l'atlas est construit de manière globale [88]. Elles peuvent aussi permettre de rendre la segmentation plus rapide par le biais de recalage de plus petites parties de l'image [95]. Enfin elles sont aussi capable de gérer correctement les inhomogénéités de champs présents sur les images IRM [85]. Il

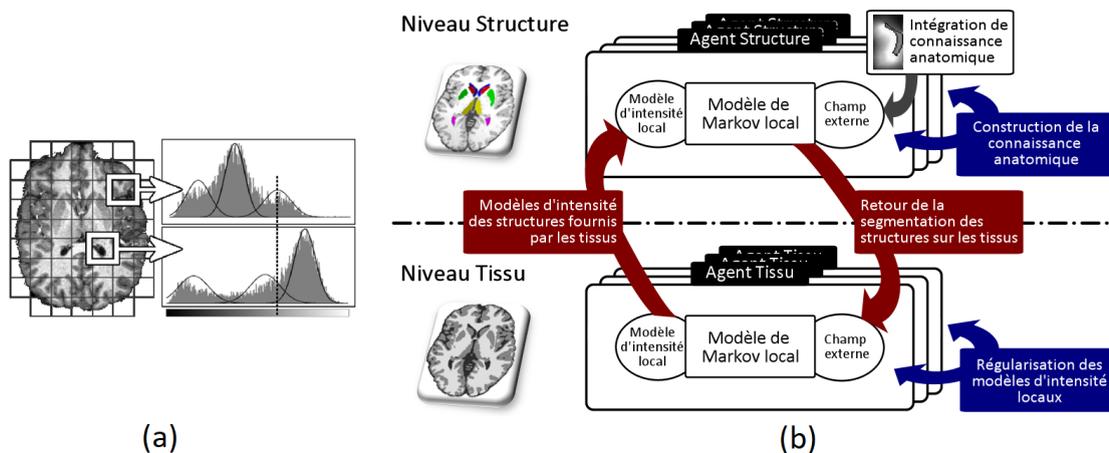


FIGURE 4.7 – (a) Division de l’image en N sous-volumes cubiques, chacun segmenté par champ de Markov (en tissus). La figure (b) représente les interactions entre les différents champs de Markov (tissus et structures) (l’image est tirée de [84]).

faut cependant remarquer que ces méthodes restent entièrement automatiques et nécessite la segmentation de tout le cerveau.

4.6 Correction *a posteriori*

Pour finir ce panorama des techniques de segmentation des images 3D, il est intéressant de noter une méthode de correction *a posteriori* proposée par [98]. Les auteurs décrivent une méthode qui peut s’appliquer à la suite d’une segmentation pour corriger les erreurs systématiques de la méthode. La segmentation peut provenir d’une méthode quelconque, notée méthode *hôte*. Cette post-correction permet d’apprendre quelles sont ces erreurs au moyen de classifieurs dédiés pour ensuite les corriger. La figure 4.8 décrit le schéma de fonctionnement de la méthode. L’apprentissage des classifieurs nécessite une base de N images d’apprentissage avec la vérité terrain associée ainsi que la segmentation produite par une méthode *hôte*. Pour chaque région r que l’on veut apprendre, un classifieur C_r est appris à l’aide des voxels présents à l’intérieur d’un volume dilaté de la région r . Un autre classifieur C_b est appris pour les voxels extérieurs n’appartenant à aucune région. Celui-ci est appris à l’aide des voxels présents dans l’intersection du volume de toutes les régions dilatées et du volume extérieur dilaté. Ce classifieur C_b est appelé classifieur du fond.

Les classifieurs utilisés sont des classifieurs *Adaboost* [40]. Ils sont construits à partir de l’association de plusieurs classifieurs faibles h . Le principe est de construire itérativement ces classifieurs faibles tout en donnant plus d’importance aux éléments de la base d’apprentissage qui ont été mal classifiés par les classifieurs faibles précédents. Le but est de créer et d’assembler des classifieurs faibles dont le but est de corriger les voxels que les précédents classifieurs n’arrivaient pas à corriger. Les classifieurs utilisés dans *Adaboost* sont de simples seuillages sur une des caractéristiques du voxel. L’apprentissage des classifieurs suit le schéma classique d’apprentissage d’*Adaboost*. L’utilisation d’un classifieur *Adaboost*

se fait ensuite par combinaison et pondération de tous les seuillages appris lors de la phase d'apprentissage défini par :

$$H(x) = \text{sign}\left(\sum_{t=1}^T \alpha_t h_t(x)\right) \quad (4.45)$$

où α est la pondération du classifieur faible h_t . Si on note $D = \sum_{t=1}^T \alpha_t h_t(x)$ alors on peut définir une probabilité d'appartenance telle que :

$$P = \frac{\exp(D)}{\exp(D) + \exp(-D)} \quad (4.46)$$

Il est possible ensuite d'utiliser cette probabilité pour comparer ou combiner différents classifieurs Adaboost.

Le classifieur C_r de la région r est appris à l'aide des caractéristiques des voxels présents dans un volume construit à l'aide d'une dilatation de la région r segmentée avec la méthode *hôte*. Les caractéristiques utilisées sont apprises sur un patch autour du voxel. On note $v_{\Delta x \Delta y \Delta z}$ le voxel du patch autour de v à la position relative (x_v, y_v, z_v) .

Ces caractéristiques contiennent les intensités relatives de chaque voxel du patch $v_{\Delta x \Delta y \Delta z}$ comparées à l'intensité moyenne \bar{I} du volume.

Les caractéristiques sont aussi composées de la valeur de la segmentation de chaque voxel du patch $S(v_{\Delta x \Delta y \Delta z})$. Cette caractéristique provient de la segmentation faite par la méthode *hôte* que l'on veut corriger.

Enfin les informations spatiales au sein du patch sont aussi prises en compte en combinant la position relative d'un voxel comparées au centroïde $(\bar{x}, \bar{y}, \bar{z})$ de la région r avec les deux précédentes caractéristiques. Si $P_x(v) = x_v - \bar{x}$, $P_y(v) = y_v - \bar{y}$ et $P_z(v) = z_v - \bar{z}$ sont les positions relatives du voxel du patch. Alors la combinaison des caractéristiques donne $P_x(v) \cdot S(v_{\Delta x \Delta y \Delta z})$, $P_y(v) \cdot S(v_{\Delta x \Delta y \Delta z})$ et $P_z(v) \cdot S(v_{\Delta x \Delta y \Delta z})$. A chaque étape itérative du classifieurs Adaboost, un seuillage est fait sur l'une de ces caractéristiques. Les principaux paramètres de la méthode sont le nombre de voxels utilisé lors de la dilatation pour obtenir le volume, ainsi que la taille du patch sur lequel les caractéristiques sont calculées.

Après la phase d'apprentissage, chaque région possède un classifieur C_r et un classifieur du fond C_b . On peut alors les utiliser pour corriger une nouvelle segmentation faite par la méthode. Les données d'entrée nécessaires sont l'image réelle ainsi que la segmentation de l'image par la méthode *hôte*. De la même manière une dilatation du volume des régions et de l'extérieur est faite pour chaque région afin de définir les volumes. Les caractéristiques sont extraites pour chaque voxel. Chaque classifieur est utilisé pour reclassifier les voxels présents dans le volume. À la fin de cette étape chaque classifieur, dont le classifieur C_b , fournit une probabilité d'appartenance à la région. La décision finale se fait alors en sélection la région (ou le fond) possédant la probabilité maximale.

Cependant, ces classifieurs ne sont utilisables que dans le cas où la segmentation des régions est la même que pendant l'apprentissage. En effet les seuillages sont appris sur

l'image segmentée par la méthode S . Elle n'est donc pas directement applicable sur une segmentation incrémentale où seulement une partie des régions est présente dans l'image devant être segmentée.

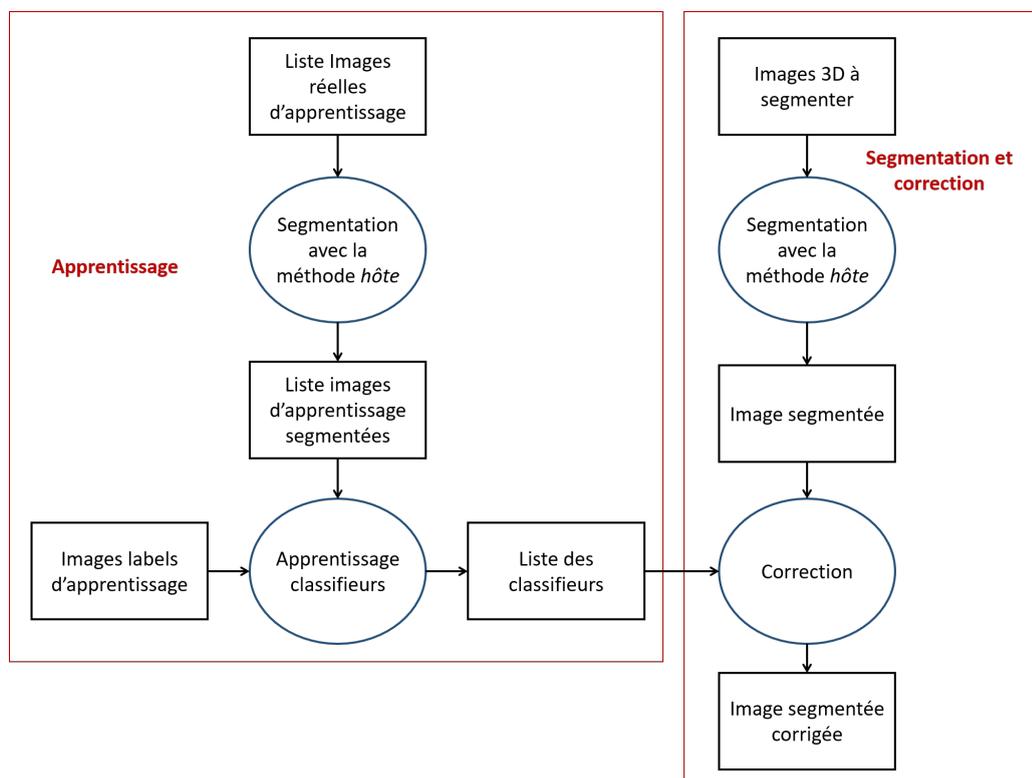


FIGURE 4.8 – Schéma de la construction et l'utilisation de la post-correction [99]

4.7 Bilan

Aujourd'hui la grande majorité des méthodes de segmentation d'images développées sont des méthodes automatiques, en particulier dans le cas des images médicales. Ces outils permettent de faire gagner du temps aux praticiens hospitaliers. Cependant, ils sont souvent dédiés à un problème particulier utilisant des informations *a priori* très spécifiques. Les quelques méthodes interactives existantes dans le cas des images médicales demandent un nombre d'interactions important. Le temps nécessaire demandé à l'utilisateur peut devenir rapidement important. Il n'existe pas de méthodes intermédiaires suffisamment automatiques pour ne pas solliciter trop intensément l'utilisateur mais suffisamment interactif pour rester suffisamment.

La quasi-totalité des outils de segmentation automatique utilisent des informations *a priori* pour guider les segmentations disponibles, le plus souvent sous forme d'atlas anatomiques. Il en existe plusieurs types chacun ayant ses avantages et inconvénients. Le stockage des informations sous forme de multi-atlas ou d'atlas probabilistes fournit aujourd'hui les informations les plus complètes et efficaces. Les méthodes de type multi-atlas fournissent souvent de meilleurs résultats que l'utilisation d'atlas probabilistes. Cependant les atlas probabilistes ne nécessitent le recalage que d'une seule image et sont donc plus rapides. C'est pourquoi ces derniers sont probablement plus adéquats dans le cas des méthodes interactives.

De même, l'ensemble des outils de segmentation exploitant les informations provenant des atlas fournissent de manière générale des résultats de bonne qualité (et similaire cf. [57]). Les champs de Markov sont néanmoins l'un des moyens les plus couramment exploités pour tenir compte des informations probabilistes d'un atlas [85, 58]. Ils ont l'avantage de pouvoir gérer la multi-modalité, les informations probabilistes de l'atlas et de fournir une régularisation spatiale.

Les informations topologiques et les relations spatiales ont montré leur efficacité pour guider certaines segmentations [29], la qualité d'une segmentation pouvant être améliorée lorsque celle-ci est complétée par ce type de données.

Le caractère multi-modal des images médicales est maintenant géré par une grande partie des algorithmes. Au niveau des atlas, il est possible de recalibrer des images de différentes modalités entre elles [77, 61]. Au niveau de la segmentation, les méthodes de classification de voxels sont capables de travailler lorsque les images sont composées de plusieurs modalités et lorsque chaque voxel est décrit par une liste d'intensité.

Lors de nos travaux, nous avons essayé de développer des méthodes de segmentation reprennent les points forts des méthodes de l'état de l'art identifiés dans ce chapitre. Le tableau 4.1 résume les différents avantages et inconvénients de ces grandes familles de mé-

thodes et les comparées avec l'approche qui sera proposée. L'idée principale est d'inclure une forme d'interactivité dans l'utilisation d'atlas de type probabiliste ; les atlas probabilistes étant assez performants, génériques et rapides d'utilisation. De plus, nous avons choisi d'utiliser des atlas définis plus localement dans le but de répondre au problème du temps de recalage des atlas. Les atlas étant de taille plus restreintes, le recalage appliqué se déroule plus rapidement. Le positionnement de cet atlas pourra, de plus, être guidé par des relations spatiales apprises au préalable. La segmentation à proprement dites sera effectué à l'aide d'un champ de Markov caché.

4.7. BILAN

Méthode de segmentation	Avantages	Inconvénients
Méthode contours actifs et interactifs (ex : ITK-SNAP)	<ul style="list-style-type: none"> • Interactivité (nécessaire) • Bonnes performances (si temps nécessaire) • Généricité 	<ul style="list-style-type: none"> • Temps utilisateur potentiellement très long
Machine learning (ex : deep learning)	<ul style="list-style-type: none"> • Bonnes performances [26, 31] • Segmentation rapide [43] 	<ul style="list-style-type: none"> • Pas d'interactivité • Apprentissage difficile (besoin de beaucoup de donnée d'apprentissage) ou potentiellement long (déformation d'images à effectuer)
FreeSurfer [36]	<ul style="list-style-type: none"> • Segmentation complète du cerveau humain (aussi cortical [36]) 	<ul style="list-style-type: none"> • Très spécifique (cerveau humain) • Segmentation longue • Pas d'interactivité
Segmentation à base d'atlas probabilistes	<ul style="list-style-type: none"> • Performances correctes • Gère la multi-modalité • Générique (applicable à différents types d'organes/structures anatomiques) [19, 69] 	<ul style="list-style-type: none"> • Apprentissage potentiellement long • Dépendant de la méthode segmentations utilisant l'atlas • Pas d'interactivité
Segmentation à base de multi-atlas	<ul style="list-style-type: none"> • Bonnes performances [99] • Gère la multi-modalité [47] • Générique [99, 111] 	<ul style="list-style-type: none"> • Segmentation longue • Pas d'interactivité • Dépendant de la méthode de fusion
Notre méthode	<ul style="list-style-type: none"> • Bonnes performances • Gère la multi-modalité • Générique • Interactivité possible • Rapide dans le cas de segmentation partielle / traitement local 	<ul style="list-style-type: none"> • Apprentissage potentiellement long (si bcp de régions) • Dépendant de la méthode segmentations utilisant l'atlas

TABLE 4.1 – Qualités et défauts des différentes catégories de méthodes de segmentation d'images médicales.

4.7. BILAN

Troisième partie

Segmentation incrémentale et interactive d'images médicales 3D

Chapitre 5

Modélisation structurelle des données anatomiques

Comme nous l'avons vu, des connaissances *a priori* concernant les structures anatomiques à analyser sont souvent exploitées pour guider pour la segmentation des images médicales. Celles-ci peuvent apporter des informations sur la forme des structures anatomiques, sur leur intensité (colorimétrique) dans une certaine modalité d'image ou sur leur position à l'intérieur d'un organe. La manière d'encoder cette connaissance *a priori* sur les images et les éléments anatomiques à localiser nous semble un point fondamental de la chaîne de traitement à développer. Ce chapitre présente la manière dont nous proposons de modéliser ces informations *a priori* afin de simplifier leur exploitation durant la phase de segmentation mais aussi la mise à jour (incrémentale) durant la phase d'apprentissage. Nous proposons pour cela une modélisation des informations *a priori* sous forme de graphe.

Les graphes constituent des outils très puissants capables de représenter et modéliser des systèmes de manière flexible et précise. Un graphe $G = (E, V)$ se compose d'un ensemble de sommets $V = \{n_{r_1}, \dots, n_{r_M}\}$ représentant les différents objets. Il se compose aussi d'un certain ensemble d'arcs $E \subseteq M \times M$ reliant différentes parties d'une scène d'un système. On peut ensuite noter $a_{r_i r_j}$ l'arc reliant le nœud n_{r_i} au nœud n_{r_j} . Les arcs peuvent alors servir à représenter les relations existantes entre les différentes parties du système à modéliser. Si le graphe est orienté alors l'arc $a_{r_i r_j}$ est différent de l'arc $a_{r_j r_i}$. Les nœuds et les arcs peuvent encapsuler des informations sur les objets et les relations qu'ils représentent au travers de labels qui leur sont associés. Dans ce cas, le graphe est dit attribué et les informations stockées sont les attributs. Les attributs liés à un nœuds n_r sont notés ℓ_{n_r} et les attributs liés à l'arc $a_{r_j r_i}$ sont notés $\ell_{a_{r_j r_i}}$.

Nous proposons d'utiliser un graphe attribué pour modéliser et stocker toutes les informations *a priori* utiles à une segmentation. Dans notre proposition, les nœuds n_{r_i} du graphe représentent les structures anatomiques r_i apprises et les arcs représentent les relations spatiales existantes (également apprises) entre ces structures. Les attributs liés aux nœuds devront permettre d'encoder des informations de forme et d'intensité associé à une région. Les attributs liés aux arcs serviront quant à eux à encoder des informations de

distance permettant de définir le positionnement relatif des atlas les uns par rapport aux autres. Ces informations permettront de positionner une région à localiser à partir des régions déjà localisées.

Plus concrètement, nous proposons d'encoder les informations de forme et d'intensité associées à chaque région anatomique sous la forme d'un atlas probabiliste local. Ainsi, chaque nœud du graphe possède les attributs suivants :

$$\ell_{n_r} = \{Id_{n_r}, T_{n_r}, P_{n_r}, N_{n_r}\} \quad (5.1)$$

Avec Id_{n_r} l'identifiant de la région r associé au nœud n_r (nom ou numéro de la région) ; T_{n_r} est le template local de la région r ; P_{n_r} la carte de probabilité locale de la région r et N_{n_r} le nombre d'images utilisées pour apprendre cette région. On désignera par r la région anatomique représentée par le nœud n_r . L'ensemble des attributs associées aux nœuds sont décrits en détails dans la section 5.1.1).

Concernant les relations spatiales représentées par les arcs du graphe G , nous proposons d'encoder les informations de relations spatiales entre régions par :

$$\ell_{a_{r_i r_j}} = \{Dist_{r_i r_j}\} \quad (5.2)$$

Avec $Dist_{r_i r_j}$ l'ensemble des informations de distance entre la position de la région r_i représentée par nœud n_{r_i} et la position de la région r_j représentée par le nœud n_{r_j} . De la même manière que pour les nœuds, la section 5.1 (p. 80) décrit en détail la manière donc sont construits les relations topologiques stockées à l'aide des arcs du graphe G .

5.1 Modélisation locale de la forme

Pour modéliser la forme et les intensités locales des différentes structures anatomiques, nous proposons d'exploiter le mécanisme d'atlas probabiliste mais de manière locale plutôt que globale (comme habituellement). Cette originalité nous paraît fondamentale et nous verrons par la suite comment elle permet une segmentation partielle, incrémentale et interactive plutôt qu'un traitement complètement automatique et peu contrôlable. Plutôt que d'avoir un seul atlas global modélisant à la fois la forme, l'intensité et la position des régions, notre approche propose d'utiliser un atlas local par structure anatomique à modéliser. Ce choix apporte plus de souplesse aussi bien pour l'apprentissage (création de l'atlas local indépendant des autres) que durant la segmentation (liés de manière moins rigide aux autres segmentation).

5.1.1 Construction des atlas probabilistes locaux

Chaque atlas local encode des connaissances *a priori* sur la forme de la région et sur les intensités des voxels composant cette région. Pour cela, nous utilisons des atlas probabilistes non classiques dans le sens où chacun se limite à une région spécifique. Pour une

région donnée, l’atlas local est formé d’un template pour chaque modalité (ex. T1 IRM) et d’une carte de probabilités d’appartenance des voxels à la région. La création d’un atlas local nécessite donc plusieurs recalages réalisés sur des volumes de données moins importants correspondants aux boîtes englobantes de la région concernée extraites des images d’apprentissage disponibles. La création des atlas locaux se déroule, en plusieurs étapes, à partir d’une base d’apprentissage, composée de N couples d’images réelles (ex. IRM) et d’images labellisées correspondantes notées I^n et L^n respectivement avec $n = \{1, \dots, N\}$. L’ensemble des régions labellisées dans l’image L^n est notée R_{L^n} . Cet ensemble n’est pas forcément identique à R_{L^m} avec $m \neq n$. En effet, il n’est pas nécessaire pour l’apprentissage des informations *a priori*, que les mêmes régions soient labellisées sur chacune des images de la base d’apprentissage. L’ensemble de toutes les régions présentes dans les différentes images L_n est noté R .

$$\bigcup_{n=1}^N R_n = R \quad (5.3)$$

La figure 5.1 décrit les étapes importantes de la construction des atlas locaux. Cette construction est produite indépendamment pour chaque région r . Les sections suivantes décrivent les différentes étapes d’apprentissage d’un atlas local.

5.1.1.1 Extraction des sous régions

La première étape du processus de création est l’extraction des boîtes englobantes de chaque région de la base d’apprentissage. Pour chaque région $r \in R$, les images label L^n sont sélectionnées si r est contenue dans R_{L^n} . Pour chacune des images sélectionnées, le volume présent à l’intérieur de la boîte englobante de r est défini et noté L_r^n . Le volume correspondant à cette boîte englobante dans l’image 3D I^n est notée I_r^n . En pratique, une marge est prise autour des régions pour deux raisons :

- D’une part cela permet d’éviter des variations abruptes dans la future carte de probabilité.
- D’autres part cela vise à compenser de futurs mauvais positionnements de l’atlas (qui auraient un trop fort impact si aucune marge n’était prise).

La boîte englobante finalement sélectionnée est une boîte englobante étendue. La marge ajoutée est définie par un paramètre correspondant à un pourcentage de la taille initiale de la boîte englobante (généralement fixé à 10 %). Dans la suite, I_r^n et L_r^n correspondent aux volumes présents à l’intérieur de la boîte englobante étendue. Ainsi, chaque région r est définie à l’aide de N_r couples de sous-images notés $\{L_r^n, I_r^n\}$. Pour chaque région r , la valeur de l’attribut N_r est déjà connu et provient du nombre d’images label L_r^n :

$$N_r = \text{card}(R_{L^n}) \quad (5.4)$$

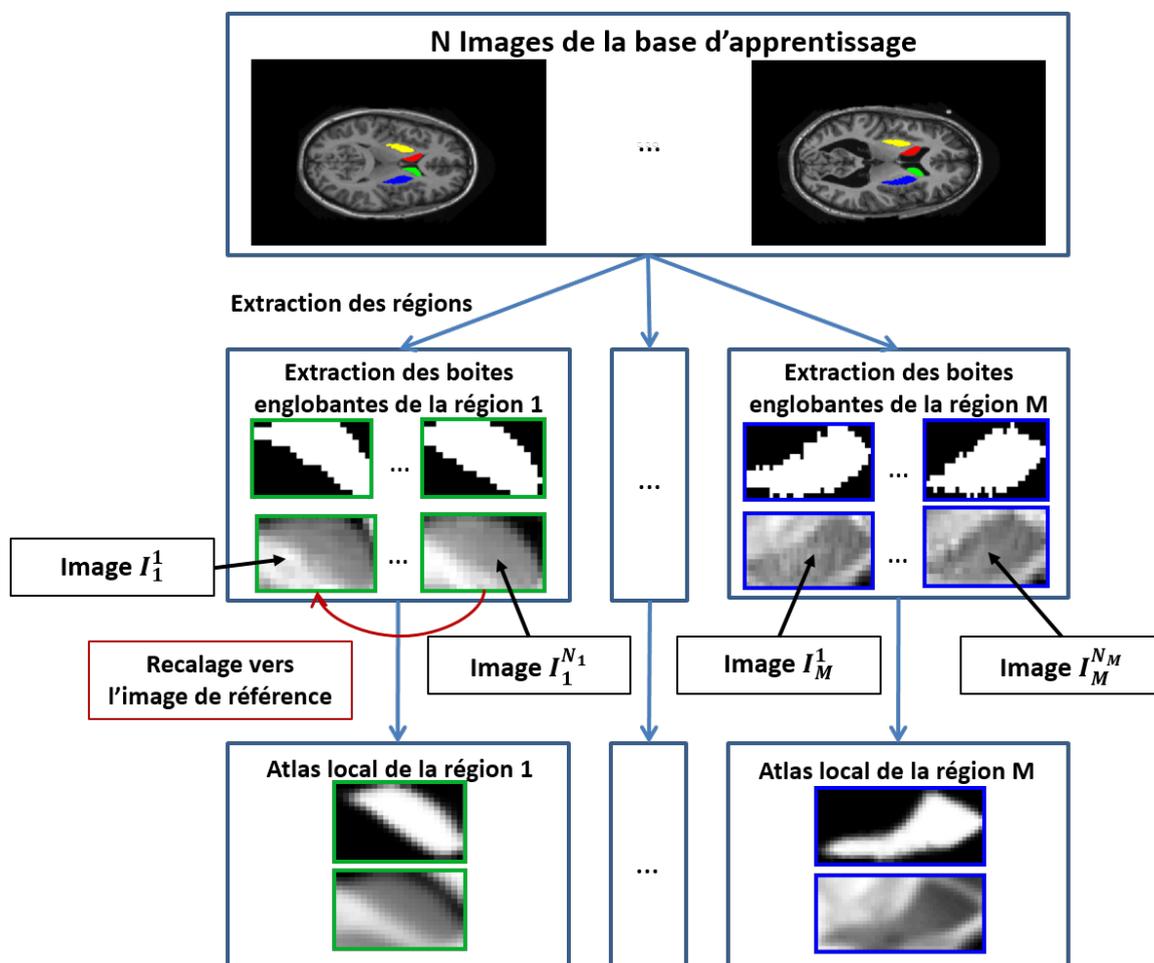


FIGURE 5.1 – Schéma de la construction des différents atlas probabilistes locaux à partir d'une base d'images d'apprentissage. Exemple d'application sur une base d'images IRM T1 de cerveaux, la région 1 représente un Noyau caudé et la région M un Putamen.

5.1.1.2 Normalisation des intensités

Pour chaque région r , une normalisation des intensités des images réelles est effectuée sur les N_r images I_r^n . La méthode utilisée est celle présentée dans [68]. Les histogrammes des intensités des images sont décrits à l'aide de 5 points de contrôle, deux valeurs extrêmes, notées e_r et s_r , permettant de supprimer les outliers (si on le souhaite) et les 3 quartiles 25%, 50% et 75% notées q_r^1 , q_r^2 et q_r^3 . Contrairement à l'utilisation usuelle [68], nous appliquons la méthode sur les sous-images I_r^n et non pas sur des images entières. Dans notre cas, les deux valeurs extrêmes sont fixées à 0 et à la valeur maximale (correspond à 0 ou 100%) (les outliers ne sont en général pas présents dans nos sous-images). La méthode se déroule en deux étapes : la construction de points de contrôle de référence à partir des N_r images, et la transformation des histogrammes de chaque image I_r^n afin que les points de contrôle soient identiques aux points de contrôle de référence. La figure 5.2 illustre le fonctionnement de cette normalisation.

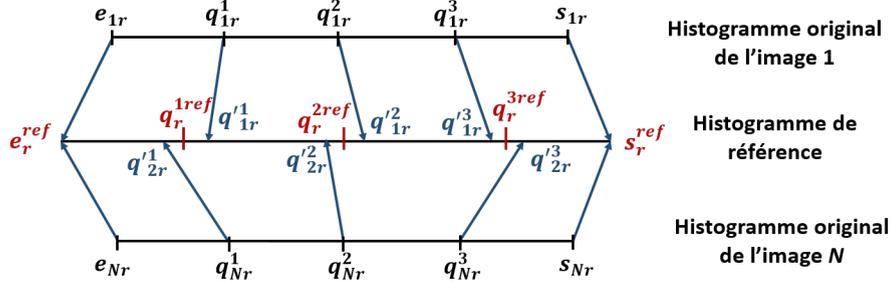


FIGURE 5.2 – Construction des points de contrôle de l’histogramme de référence. e_r^{ref} et s_r^{ref} sont fixés par l’utilisateur tandis que les q_r^{ref} sont calculés comme la moyenne des quartiles des histogrammes disponibles.

Ainsi, pour une région r , les histogrammes \mathcal{H}_r^n associées aux sous-images I_r^n sont calculés, puis, sur chaque histogramme les trois quartiles sont détectés ($q_{nr}^1, q_{nr}^2, q_{nr}^3$). Les valeurs extrêmes de l’histogramme de référence e_r^{ref} et s_r^{ref} sont fixées à 0 et à la valeur maximale. Les quartiles de référence sont ensuite construits comme la moyenne des quartiles précédents (moyenne des N_r images). Pour chaque région trois quartiles de référence sont obtenus et notés $q_r^{i,ref}$ avec $i = \{1, 2, 3\}$.

Ensuite, les histogrammes \mathcal{H}_{nr} et les points de contrôle associés sont sélectionnés. Une transformation des histogrammes est appliquée pour faire correspondre les quartiles q_{nr}^i avec les quartiles $q_r^{i,ref}$. Une transformation linéaire est faite entre chaque point de contrôle pour faire correspondre une intensité de l’histogramme d’origine vers l’histogramme de référence.

Cette normalisation des intensités est locale à la région r et il y a autant de normalisations d’intensité que d’occurrences de la région présentes pour l’atlas que l’on veut créer. À la fin de cette étape, les N_r images d’apprentissage I_r^n possèdent donc des niveaux d’intensités normalisés.

5.1.1.3 Choix de l’image de référence

Parmi les N_r images I_r^n , l’une d’elle doit être sélectionnée pour être l’image de référence de l’atlas local de r . Cette référence constitue le premier template local de r noté $T_r^{(1)}$. L’image sélectionnée est celle qui minimise la somme des distances euclidiennes au carré (SSD) avec toutes les autres images de la base d’apprentissage associée à la région r . L’image label L_r^n associée au premier template $T_r^{(1)}$ permet de produire la première carte de probabilités $P_r^{(1)}$ (si le voxel est labellisé *région* la probabilité associée est mise à 1, 0 étant la valeur par défaut).

5.1.1.4 Recalage itératif des images

L'étape suivante consiste à faire évoluer T et P en tenant compte des autres images de la base d'apprentissage. Pour une certaine région r , les sous-images vont être recalées sur le template les unes après les autres. À l'étape (q) , l'un des N_r couples de sous-images L_r^n et I_r^n qui n'a pas encore été utilisé, noté $L_r^{(q)}$ et $I_r^{(q)}$, est sélectionné.

La sous-image sélectionnée $I_r^{(q)}$ est recalée vers le template actuel (T_R^1 pour la 1^{re} image) en deux temps. Dans un premier temps, les dimensions de $I_r^{(q)}$ sont ajustées à celles de $T_r^{(q-1)}$ au moyen d'une transformation τ_1 : un redimensionnement obtenu à l'aide d'une interpolation trilineaire. Puis, un recalage non linéaire basé sur le recalage B-spline, tel que présenté dans [81], est appliqué. Cette transformation est notée τ_2 .

Le template T et la carte de probabilité P sont alors mis à jour en moyennant les valeurs actuelles avec les informations recalées selon :

$$T_r^{(q)}(v) = \frac{(T_r^{(q-1)}(v) * (q-1) + I_r^{(q)}(\tau_2(\tau_1(v))))}{q} \quad (5.5)$$

$$P_r^{(q)}(v) = \frac{(P_r^{(q-1)}(v) * (q-1) + L_r^{(q)}(\tau_2(\tau_1(v))))}{q} \quad (5.6)$$

Lors d'un recalage d'image certains voxels d'une image (proche des bords de l'image) peuvent n'être reliés à aucun autres voxels dans l'autre image. Cependant, il est courant dans le cas des images médicales (ex. cerveau ou cœur) que l'extérieur soit composé d'air. Les voxels présents à cet emplacement possèdent une valeur bien souvent nulle ou au moins uniforme. Il est donc assez facile de gérer ces problèmes en fixant une valeur par défaut aux voxels qui ne serait relié à aucun autre voxel. Dans notre cas, les régions à recalcer sont des parallélépipèdes pour lesquels l'intensité des voxels sur les bords est souvent non nulle. Il est possible qu'un voxel v de l'image cible T_R^{q-1} ne soit lié à aucun voxel de l'image à recalcer I_R^q . Dans ce cas, pour le template, le voxel v conserve la valeur qu'il avait précédemment et la carte de probabilité est mise à jour en considérant la probabilité d'appartenance de ce voxel à la région comme étant nulle. Les voxels v se trouvant dans ce cas de figure sont mis à jour selon :

$$T_r^{(q)}(v) = T_r^{(q-1)}(v) \quad (5.7)$$

$$P_r^{(q)}(v) = \frac{(P_r^{(q-1)}(v) * (q-1))}{(q)} \quad (5.8)$$

En fin de processus et lorsque toutes les itérations sont effectuées, le couple $\{T_r, P_r\}$ forme l'atlas local de la région r et sera stocké dans le nœud du graphe correspondant à la région r . Le label lié au nœud r est défini par :

$$\ell_{n_r} = \begin{cases} Id_r & \text{Identifiant de la région } r \\ N_r & \text{Nombre d'image ayant servi à la construction de cet atlas local} \\ T_r & \text{Image template liée à l'atlas local} \\ P_r & \text{Carte de probabilité liée à l'atlas local} \end{cases} \quad (5.9)$$

5.1.2 Multi-atlas locaux

Notre approche n'est pas limitée à l'utilisation d'atlas probabiliste locaux. Il est possible de mettre en place notre approche locale avec d'autres formes d'atlas. Il est par exemple, envisageable d'exploiter des multi-atlas locaux. Dans ce cas, chaque région est décrite par N_r couples d'images I_r^n et P_r^n . P_r^n est la version binarisée de L_r^n (pour chaque voxel de l'image L_r^n valant r , la valeur associée de P_r^n est 1 (0 sinon). L'ensemble des attributs lié au nœud r peut se définir par :

$$\ell_{n_r} = \begin{cases} Id_r & \text{Identifiant de la région} \\ N_r & \text{Nombre d'images présentes dans le multi-atlas local} \\ I_r^1 & \text{1^{re} image template liée au multi-atlas local de la région } r \\ L_r^1 & \text{1^{re} carte binaire liée au multi-atlas local de la région } r \\ \dots & \\ I_r^{N_r} & \text{N^{re} image template liée au multi-atlas local de la région } r \\ L_r^{N_r} & \text{N^{re} carte binaire liée au multi-atlas local de la région } r \end{cases} \quad (5.10)$$

5.1.3 Apprentissage incrémental

Pour ces deux types d'atlas locaux, il est facile de définir un mécanisme d'apprentissage incrémental des connaissances a priori (mise à jour des labels associés aux nœuds). Les données étiquetées (vérité terrain fournie par l'expert) n'étant pas toujours facile à acquérir, cette capacité nous a paru aussi très intéressant à proposer dans notre chaine de traitement. Les images segmentées jour après jour peuvent aussi servir à alimenter la base d'apprentissage associée à chaque région.

Dans le cas des multi-atlas locaux, l'ajout d'une nouvelle image est fait simplement en ajoutant les nouvelles informations (couple d'atlas locaux). Si I^{N_r+1} est une nouvelle image réelle 3D et L^{N_r+1} l'image labellisée correspondante, alors on extrait les sous-images $I_r^{N_r+1}$ et $L_r^{N_r+1}$ de la manière habituelle pour chaque région r disponible. Ensuite, tous ces couples d'images sont ajoutés aux couples déjà présents dans la base d'apprentissage du nœud du graphe associé à la région r . Le nombre d'images N_r dans la base d'apprentissage est incrémenté.

Dans le cas des atlas probabilistes locaux, l'ajout d'une nouvelle image ou d'un nouveau groupe d'images nécessite de nouvelles étapes de recalage. Après l'extraction des sous-images $I_r^{N_r+1}$ et $L_r^{N_r+1}$, celles-ci vont être recalées vers les atlas locaux (T_r et P_r) déjà

5.2. APPRENTISSAGE ET MODÉLISATION DES INFORMATIONS TOPOLOGIQUES

existants du graphe de connaissances *a priori*. Le processus est similaire à celle présentée en section 5.1.1.4. Une normalisation des intensités est effectuée entre les sous-images $B_r^{N_r+1}$ et les templates T_r . Les sous-images $I_r^{N_r+1}$ et $L_r^{N_r+1}$ sont recalées telles que :

$$T_r^{new}(v) = \frac{(T_r * N_r(v) + I_r^{N_r+1}(\tau_2(\tau_1(v))))}{(N_r + 1)}$$

$$P_r^{new}(v) = \frac{(P_r * N_r(v) + L_r^{N_r+1}(\tau_2(\tau_1(v))))}{(N_r + 1)}$$

T_r^{new} et P_r^{new} est le nouvel atlas local de la région r dans le cas où une seule image est ajoutée à la région r . Si un ensemble d'images est ajouté, les images sont ajoutées les unes après les autres. Si dans l'ensemble d'images, certaines régions n'étaient pas présentes dans le graphe de connaissances *a priori* existant, alors on construit leurs atlas locaux selon le processus décrit dans la section 5.1.1. La construction incrémentale des atlas probabilistes locaux ne fournit pas un résultat identique à une construction avec toutes les images disponibles aux même instant. En effet, l'image de référence n'est pas forcément la même dans les deux cas. Si le premier graphe de connaissances *a priori* est appris avec une seule image, alors ce sera forcément cette image qui jouera le rôle d'image de référence dans la suite de la segmentation. Dans tous les cas, l'image qui va jouer le rôle de référence est une image présente dans le premier ensemble d'images d'apprentissage. Enfin le nombre N_r est augmenté du nombre de nouvelles sous-images incorporées à l'atlas local de la région r .

5.2 Apprentissage et modélisation des informations topologiques

Les atlas locaux n'incluent pas l'information de position et de taille de la région à l'intérieur de la scène à modéliser. Avant d'utiliser une atlas local, il faut positionner la boîte englobante de ce dernier dans la scène. Pour apprendre et modéliser cette information, les relations spatiales entre chacune des structures doivent être apprises à partir d'images d'apprentissage et des segmentations associées. Nous proposons de les stocker à l'intérieur des arcs du graphe de connaissances *a priori*. Ces relations ont pour but de lier les régions les unes par rapport aux autres (en terme de distance et de taille relative). L'objectif est de placer une région cible à partir d'une ou de plusieurs régions sources connues.

5.2.1 Apprentissage des relations spatiales

Comme nous l'avons mentionné dans l'état de l'art, l'exploitation de relations spatiales entre régions pour la segmentation d'images n'est pas une nouveauté. Par exemple, les relations floues [15] peuvent permettre de définir un domaine de présence probable pour un objet donné dans une scène à partir de relations spatiales le liant à d'autres objets. La sortie provenant directement de ces informations est une carte de probabilité de présence de la région dans le domaine spatial de l'image. Notre problématique diffère et vise plus

5.2. APPRENTISSAGE ET MODÉLISATION DES INFORMATIONS TOPOLOGIQUES

simplement, à dimensionner et à placer de la manière la plus adéquate possible la région à rechercher en fonction de la position des régions préalablement localisées dans l'image. Les boîtes englobantes sont des parallélépipèdes qui doivent être positionnés au sein de l'image entière à segmenter, pour que ensuite les atlas locaux soient recalés afin d'obtenir une probabilité d'appartenance à la région recherchée pour chaque voxel à l'intérieur du parallélépipède positionné.

Les bords des boîtes englobantes de chaque région peuvent être positionnés à partir des bords des boîtes englobantes des autres régions préalablement segmentées. Dans le cas d'images 3D, douze distances entre les bords de chaque structure sont alors apprises et stockées comme information *a priori* reliant les plans comparables selon un repère donné (cf. figure 5.3). Les valeurs de distance sont calculées relativement à la taille de la région source, ce qui permet de rendre les relations indépendantes aux dimensions des images utilisées (et donc de la résolution ou de la taille de la région à modéliser).

Comme défini dans la section 5.1, la région source est notée r_1 et la région cible est notée r_2 dans le cas d'une image n . Le but est de décrire la position de r_2 par rapport à celle de r_1 . On considère un repère orthonormal $(0, \vec{x}, \vec{y}, \vec{z})$. Pour la suite de la méthodologie, on se place dans le cas de direction \vec{x} . La dimension de r_1 dans la direction \vec{x} est notée dim_x . Les quatre valeurs $d_{r_1 r_2}^{x11}$, $d_{r_1 r_2}^{x12}$, $d_{r_1 r_2}^{x21}$ et $d_{r_1 r_2}^{x22}$, représentent les distances entre le premier et le deuxième bord de r_1 vers le premier et le deuxième bords de r_2 dans la direction \vec{x} . On calcule ensuite les quatre distances relatives d_r suivante :

$$d_{r_1 r_2}^{xij} = \frac{d_{r_1 r_2}^{xij}}{dim_x} \quad i = (1, 2), \quad j = (1, 2) \quad (5.11)$$

Cette distance relative est calculée pour toutes les images de la base d'apprentissage comportant le couple r_1 et r_2 . On ne conserve qu'un intervalle de distance composé de la distance minimum et maximum $[Min_{d_{r_1 r_2}^{xij}}, Max_{d_{r_1 r_2}^{xij}}]$ observées dans la base d'apprentissage, d'après les relations :

$$\begin{cases} Max_{d_{r_1 r_2}^{xij}} &= \max_{I_n, n=1, \dots, N} (d_{r_1 r_2}^{xij}) \\ Min_{d_{r_1 r_2}^{xij}} &= \min_{I_n, n=1, \dots, N} (d_{r_1 r_2}^{xij}) \end{cases} \quad (5.12)$$

On obtient alors quatre intervalles selon \vec{x} pour décrire le positionnement de r_2 par rapport à r_1 . En appliquant le même schéma dans les directions \vec{y} et \vec{z} , on obtient 12 intervalles. Les relations spatiales apprises étant relatives à la région source, la distance de r_1 par rapport à r_2 n'est pas la même que la distance de r_2 par rapport à r_1 . Il est donc possible d'obtenir 24 intervalles pour décrire l'ensemble des relations entre 2 régions (r_1 et r_2) que l'on va stocker comme label de l'arc correspondant dans notre graphe de connaissances *a priori* afin de pouvoir les utiliser lors des segmentations.

Le label lié à l'arc reliant la région r_1 à la r_2 est donc composé de 12 intervalles définis par :

$$l_{a_{r_1 r_2}} = \begin{cases} [Min_{dr_{r_1 r_2}^{xij}}, Max_{dr_{r_1 r_2}^{xij}}] & (i, j) \in (1, 2)^2 \\ [Min_{dr_{r_1 r_2}^{yij}}, Max_{dr_{r_1 r_2}^{yij}}] & (i, j) \in (1, 2)^2 \\ [Min_{dr_{r_1 r_2}^{zij}}, Max_{dr_{r_1 r_2}^{zij}}] & (i, j) \in (1, 2)^2 \end{cases} \quad (5.13)$$

D'autres formes de modélisation de relations spatiales ont aussi été testées. Nous avons, par exemple, essayé de rendre les relations spatiales relatives par rapport à la valeur moyenne de dim_x ou alors à la taille moyenne du cerveau. Nous avons également essayé de stocker les distances sous une forme de gaussienne. Au lieu de conserver le minimum et le maximum des distances relatives, la moyenne et l'écart type relatives sont calculés et conservés. Quatre valeurs moyennes relatives $\mu_{dr_{r_1 r_2}^{xij}}$ et quatre écarts types relatifs $\sigma_{dr_{r_1 r_2}^{xij}}$ dans chaque dimension doivent alors être stockés dans chaque arc du graphe. Dans le cas de la direction \vec{x} :

$$\mu_{dr_{r_1 r_2}^{xij}} = \sum_{I_n, n=1, \dots, N} \frac{dr_{r_1 r_2}^{xij}}{N}$$

$$\sigma_{dr_{r_1 r_2}^{xij}} = \sqrt{\sum_{I_n, n=1, \dots, N} (\mu_{dr_{r_1 r_2}^{xij}} - dr_{r_1 r_2}^{xij})^2}$$

Le label lié à l'arc entre la région r_1 vers la r_2 est donc composé de 12 intervalles et définis par :

$$l_{a_{r_1 r_2}} = \begin{cases} [\mu_{dr_{r_1 r_2}^{xij}}, \sigma_{dr_{r_1 r_2}^{xij}}] & (i, j) \in (1, 2)^2 \\ [\mu_{dr_{r_1 r_2}^{yij}}, \sigma_{dr_{r_1 r_2}^{yij}}] & (i, j) \in (1, 2)^2 \\ [\mu_{dr_{r_1 r_2}^{zij}}, \sigma_{dr_{r_1 r_2}^{zij}}] & (i, j) \in (1, 2)^2 \end{cases} \quad (5.14)$$

Nous avons choisi de conserver les intervalles de distance minimum et maximum expliqués ci-dessus car les résultats étaient similaires (cf. section 7.2.2) pour chacune de ces deux modélisations sur nos bases de test. Cependant cette modélisation autorise directement un apprentissage incrémental des relations spatiales. Il suffit alors de modifier l'intervalle si une image supplémentaire dans la base d'apprentissage ne respecte pas l'ancien intervalle.

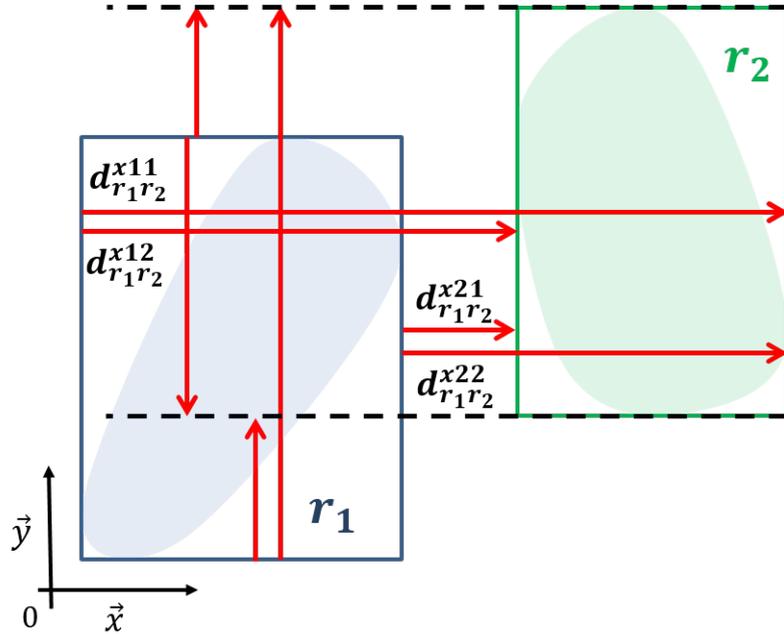


FIGURE 5.3 – Illustration dans le plan $(0, \vec{x}, \vec{y})$ des relations spatiales de la région r_1 vers la région r_2 (8 distances dans le cas 2D). Quatre distances dans chaque direction font ici un total de 8 directions.

5.2.2 Graphe de connaissances *a priori*

À la fin de la phase d'apprentissage des connaissances *a priori*, toutes les informations nécessaires pour réaliser une segmentation guidée sont disponibles dans un graphe G construit à partir des données d'apprentissage disponibles. Ces informations sont stockées sous la forme d'un graphe que l'on appelle graphe de connaissances *a priori*. Les nœuds représentent les régions anatomiques recherchées et caractérisent leur forme et leur intensité colorimétrique (éventuellement pour différentes modalités d'acquisition). Ces informations sont stockées sous forme d'atlas locaux. Les arcs représentent les relations spatiales et topologiques entre structures anatomiques. Ces connaissances sont stockées sous forme intervalles de distances relatives entre bords. Ces arcs sont orientés, il y a donc deux arcs entre chaque nœud représentant des régions présentes dans la base d'apprentissage.

Dans le cas où chaque couple de région possible est présent dans au moins une image de la base d'apprentissage, le graphe G obtenu est un graphe complet. La figure 5.4 illustre la structure d'un graphe de connaissances *a priori* composé de 3 régions dans le cas d'images IRM T1 du cerveau humain.

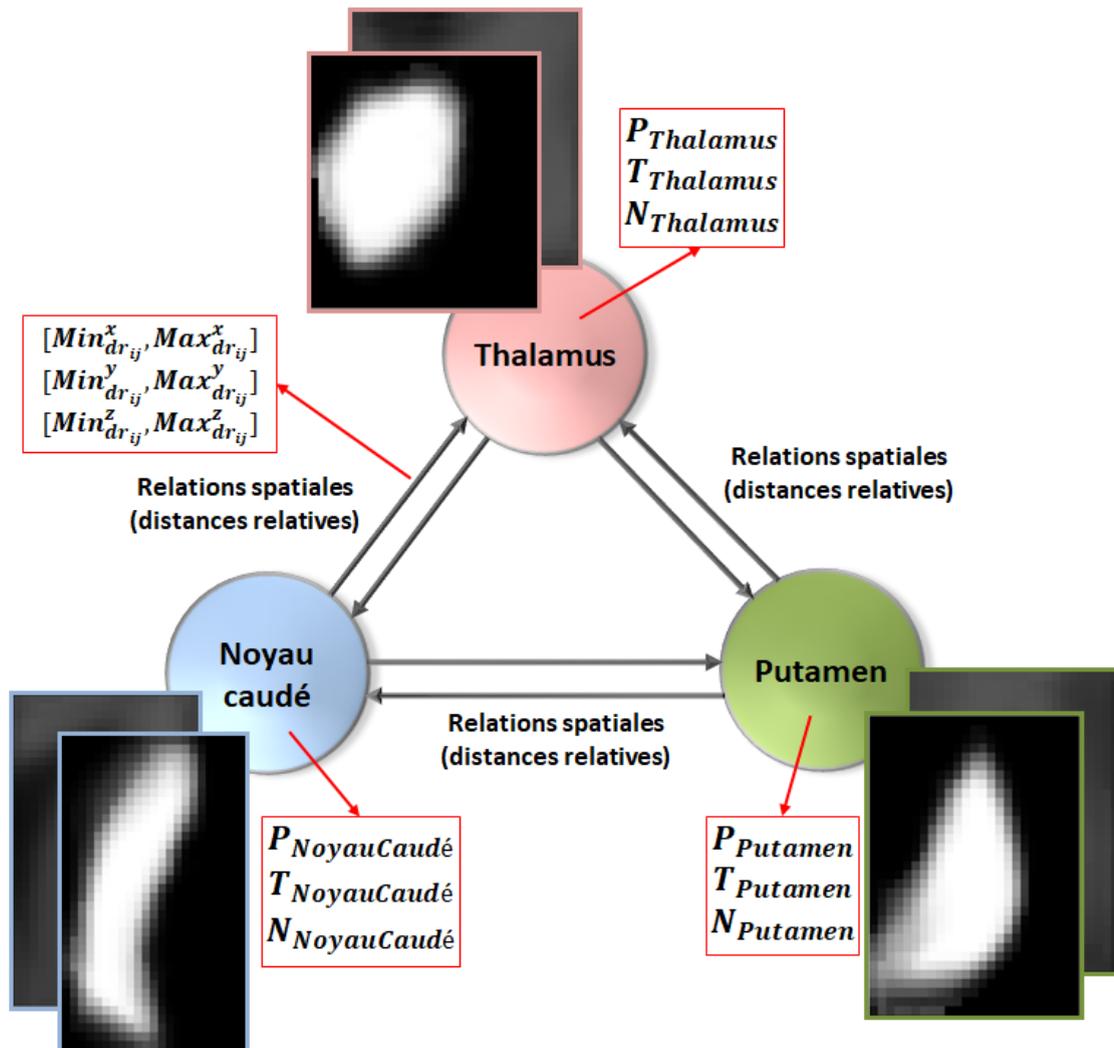


FIGURE 5.4 – Exemple de graphe *a priori* dans le cas de 3 structures sous-corticales du cerveau humain (Putamen, Noyau caudé et Thalamus).

5.3 Complément de l'apprentissage pour une correction a posteriori

En plus des informations *a priori* classiquement exploitées en segmentation d'images guidée par les données, il nous a semblé intéressant d'ajouter un type de connaissance beaucoup plus marginal qui concerne des connaissances sur les erreurs classiquement commises par la segmentation lorsque celle-ci tentent de labelliser les données soumises. Il s'agit alors d'apprendre des "méta connaissances" permettant d'effectuer un post traitement après chaque segmentation locale et visant à améliorer incrémentalement les résultats préalablement obtenus. Cette idée provient des travaux présentés dans [98] décrite dans

la section 4.6 et proposant une méthode efficace permettant d'améliorer la qualité d'une segmentation. Celle-ci a pour objectif de corriger les erreurs systématiques qu'une méthode de segmentation peut avoir tendance à produire.

Cette méthode n'est pas directement utilisable dans le cadre d'une segmentation locale et incrémentale nécessitant d'appliquer cette correction après chaque nouvelle segmentation. En effet, la correction utilise un classifieur du fond unique qui est appris et utilisé dans le cas où toutes les régions sont segmentées simultanément. Dans notre cas, nous proposons d'apprendre un classifieur du fond pour chaque région pour définir et rester fidèle à notre approche locale. Nous proposons, de plus, une seconde amélioration par rapport à la méthode originale [98] consistant à prendre en entrée de cette étape de post-classification, non plus les labels fournis par la méthode de segmentation mais plutôt les cartes de probabilité locales d'appartenance à la région fournies par notre méthode de segmentation locale.

5.3.1 Exploitation d'informations probabilistes en remplacement des labels de régions

La post correction proposée dans [98] se base sur l'apprentissage de multiples classifieurs *Adaboost*. La méthode de correction ainsi que le principe d'utilisation des classifieurs *Adaboost* est décrit dans la section 4.6. Il y a un classifieur *Adaboost* C_r par région. Chaque classifieur est construit à l'aide d'une combinaison de classifieurs faibles. Ces classifieurs ont pour but d'apprendre à reclassifier les voxels mal classifiés par la segmentation initiale. Pour cela, il faut définir les caractéristiques qui seront fournis aux classifieurs *Adaboost*. Dans [98], les auteurs utilisent des caractéristiques basées sur une combinaison des intensités des voxels, de leur positionnement spatial à l'intérieur de la région et du label qu'a fourni la méthode initiale de segmentation.

La plupart des méthodes de segmentation fonctionnant par classification de voxels peuvent fournir des probabilités d'appartenance à une région en plus du label finalement attribué au voxel. Il est alors plus intéressant d'utiliser ces valeurs comme caractéristiques plutôt que les décisions finales. En effet les valeurs de probabilité fournissent une information plus riche que uniquement le label. Dans ce cas, pour chaque classifieur C_r , les caractéristiques correspondant aux labels de la segmentation sont remplacées par les probabilités d'appartenance des voxels à la région provenant du classificateur de voxels.

Les caractéristiques du voxels v , présent aux coordonnées (x_v, y_v, z_v) , se composent de l'ensemble de caractéristiques provenant des voxels $v_{\Delta x \Delta y \Delta z}$ présents dans un patch autour du voxel. Le voxel $v_{\Delta x \Delta y \Delta z}$ est le voxel présents aux coordonnées $(x_v + \Delta x, y_v + \Delta y, z_v + \Delta z)$.

Chaque voxel $v_{\Delta x \Delta y \Delta z}$ du patch est décrit par :

- son intensité relative $\mathcal{I}_{rel}(v_{\Delta x \Delta y \Delta z})$ par rapport à la moyenne de l'intensité de la région.
- la probabilité $P_r^{classif}(v_{\Delta x \Delta y \Delta z})$ de d'appartenance du voxel à la région provenant de la méthode de segmentation initiale.

5.3. COMPLÉMENT DE L'APPRENTISSAGE POUR UNE CORRECTION A POSTERIORI

- une combinaison (produit) de la position spatiale du voxel v par rapport au centroïde de la région d'une part, et des probabilités d'appartenance des voxels du patch d'autre part.

L'ensemble des caractéristiques de chaque voxel du patch autour du voxel v est défini tel que :

$$carac_v(\Delta x \Delta y \Delta z) = \begin{cases} \mathcal{I}_{rel}(v_{\Delta x \Delta y \Delta z}) \\ P_r^{classif}(v_{\Delta x \Delta y \Delta z}) \\ P_x(v) \cdot P_r^{classif}(v_{\Delta x \Delta y \Delta z}) \\ P_x(v) \cdot \mathcal{I}_{rel}(v_{\Delta x \Delta y \Delta z}) \end{cases} \quad \begin{matrix} P_y(v) \cdot P_r^{classif}(v_{\Delta x \Delta y \Delta z}) \\ P_y(v) \cdot \mathcal{I}_{rel}(v_{\Delta x \Delta y \Delta z}) \end{matrix} \quad \begin{matrix} P_z(v) \cdot P_r^{classif}(v_{\Delta x \Delta y \Delta z}) \\ P_z(v) \cdot \mathcal{I}_{rel}(v_{\Delta x \Delta y \Delta z}) \end{matrix} \quad (5.15)$$

où :

- $\mathcal{I}_{rel}(v_{\Delta x \Delta y \Delta z}) = \mathcal{I}(v_{\Delta x \Delta y \Delta z}) - \bar{\mathcal{I}}$ avec $\bar{\mathcal{I}}$ l'intensité moyenne de la région.
- $P_x(v) = x_v - \bar{x}$, $P_y(v) = y_v - \bar{y}$ et $P_z(v) = z_v - \bar{z}$ sont les positions relatives du voxel v par rapport au centroïde de la région à la position $(\bar{x}, \bar{y}, \bar{z})$.

L'ensemble de ces caractéristiques sont illustrées sur la figure 5.5.

Au delà de fournir des informations plus riches que la segmentation en label, l'utilisation de la probabilité $P_r^{classif}(v)$ provenant du champ de Markov permet d'avoir des informations indépendantes des autres régions. En effet, la segmentation finale n'est plus utilisées pour apprendre le classifieur. La correction *a posteriori* peut donc être utilisée indépendamment des régions segmentées autour.

Les voxels utilisés pour l'apprentissage de ces classifieurs sont normalement [98] ceux présents dans un volume dilaté de la région r . Pour définir ce volume à partir de la carte de probabilité, on applique simplement un seuillage à la probabilité 0.5 puis on applique la dilatation. De plus, les voxels disponibles ne sont pas tous utilisés pour des problèmes de temps de calcul et de mémoire (dans notre situation, 5 % des voxels sont sélectionnés aléatoirement).

5.3.2 Ajout de classifieurs non-région

En plus des questions de performances, l'utilisation d'un seul classifieur pour le fond commun pour toutes les régions pose problème lorsque l'on veut appliquer la correction *a posteriori* de manière incrémentale et indépendamment pour chaque région. Lors de l'apprentissage d'un unique classifieur, les labels de toutes les régions sont susceptibles d'apparaître alors que cela est impossible durant la phase de segmentation incrémentale d'une région. Pour fonctionner de façon incrémentale, il est nécessaire de modifier ce fonctionnement pour faire en sorte de n'exploiter que des données locales lors de l'apprentissage

5.3. COMPLÉMENT DE L'APPRENTISSAGE POUR UNE CORRECTION A POSTERIORI

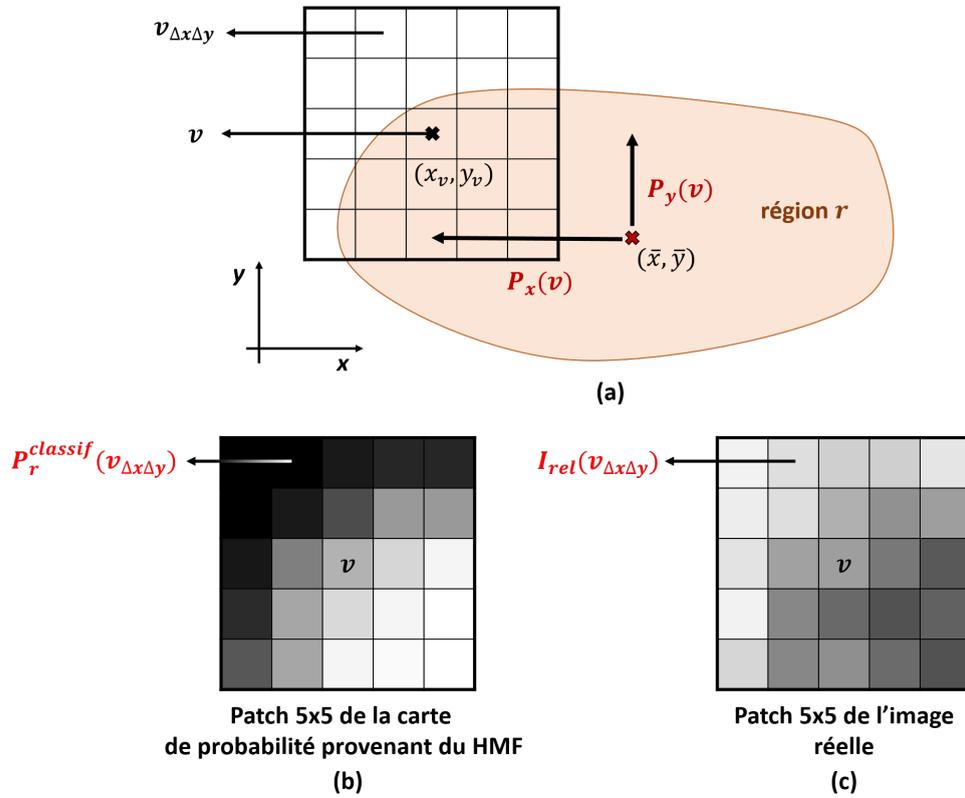


FIGURE 5.5 – Illustration des caractéristiques utilisées pour définir les classifieurs Adaboost (dans le cas 2D). La région r est illustrée ici est le seuillage de la carte de probabilité provenant de la segmentation à corriger. (a) illustre un exemple pour un voxel v et une région r . La grille représente le voisinage autour du voxel v permettant de calculer les caractéristiques. (b) représente les valeurs de probabilité provenant du HMF sur le patch autour du voxel v . (c) représente les valeurs d'image réelle sur le patch autour du voxel v . Les données illustrées en rouge font référence aux données décrites dans l'équation 5.5.

de tous les classifieurs sans dépendre d'aucune information provenant des autres régions.

La méthode la plus simple pour résoudre ce problème, déjà utilisée dans [98], est de ne pas apprendre de classifieur pour le fond. Les décisions des classifieurs sont combinés et le classifieur qui fournit la plus forte probabilité d'appartenance détermine le label final de la région. Sans classifieur de fond, il est possible de décider que si aucun des classifieurs ne fournit une probabilité supérieure à 0.5 alors le voxel est classifié comme appartenant au fond.

Une autre solution est de créer un classifieur *fond* pour chaque région que l'on va noter $C_{\bar{r}}$. Dans la même idée que le classifieur du fond C_b (cf. section 4.6 p. 70), le classifieur $C_{\bar{r}}$ est appris sur un VOI représenté par l'intersection entre une dilatation de la région r et une dilatation de l'extérieur de la région r . Le VOI pour le classifieur $C_{\bar{r}}$ est aussi définie par le seuillage à 0.5 de la carte de probabilité provenant du champ de Markov. Chaque

région est caractérisée par un couple de classifieurs $\{C_r, C_{\bar{r}}\}$.

Tous ces classifieurs sont appris indépendamment les uns des autres et peuvent donc aussi être appliqués indépendamment les uns des autres de manière incrémentale. En pratique, le stockage de ces informations, toujours comme attributs associés aux nœuds du graphe, se compose de deux types de données :

- les paramètres liés à l'extraction des caractéristiques (nombre de voxel pour la dilatation, taille du patch, etc.)
- les paramètres définissant chaque classifieur faible composant le classifieur Adaboost, telles que les caractéristiques sélectionnées, les seuillages et les pondérations associées.

5.4 Bilan

Nous avons présenté dans ce chapitre une modélisation originale sous forme de graphe des connaissances *a priori* pouvant servir à guider une segmentation. Les informations de chaque région sont indépendantes les unes des autres et reliées entre elles par des relations spatiales représentées par les arcs du graphe. Les informations sont modélisées localement à l'aide d'un graphe de connaissances *a priori* encodant les informations de formes, d'intensité, des relations spatiales, et de post-traitement de correction. Nous avons ainsi décrit comment apprendre (incrémentalement), à partir d'une base d'apprentissage, des connaissances concernant la forme et l'intensité d'une région par l'intermédiaire d'atlas locaux qui seront stockées dans les nœuds du graphe. D'autre part, l'apprentissage (incrémental) des relations spatiales à partir d'une base d'apprentissage sous forme d'intervalles de distance qui seront stockées dans les arcs du graphe. Enfin nous avons vu comment apprendre les paramètres d'un mécanisme de post-correction sous forme de différents classifieurs Adaboost ayant pour but de corriger les erreurs systématiques d'une méthode de segmentation. La section suivante décrit les modes d'exploitation que nous proposons pour ces différents types de connaissances *a priori* apprises.

Chapitre 6

Exploitation d'informations locales pour la segmentations d'images

Ce chapitre décrit le mode opératoire que nous proposons pour exploiter la modélisation des connaissances *a priori* apprises précédemment pour segmenter des images médicales 3D. Nous considérons qu'un graphe G modélisant les informations *a priori* a été construit avec l'aide d'une base d'apprentissage décrivant les différentes structures à localiser dans les images. Nous décrirons d'abord le processus global de segmentation et ensuite chaque étape sera détaillée en insistant sur la manière dont sont utilisées les connaissances *a priori* apprises préalablement sous forme d'un graphe (atlas locaux et relations spatiales).

6.1 Schéma de segmentation

La chaîne de segmentation proposé vise à introduire de l'interactivité dans le processus de segmentation, ce qui se traduit par un traitement séquentiel des régions (les unes après les autres). Le même processus est appliqué de manière incrémental lors de chaque segmentation d'une nouvelle région. Ce processus peut nécessiter certaines informations provenant de l'utilisateur. La segmentation d'une région r se déroule en plusieurs étapes :

- choix de la région à segmenter (parmi celles disponibles dans le graphe de connaissances *a priori*)
- positionnement d'une boîte englobante autour de la région à segmenter au sein de l'image par l'utilisateur (obligatoirement pour la première région) ou automatiquement.
- classification des voxels afin d'obtenir un premier résultat de segmentation
- affinement du résultat par application de la correction *a posteriori*

6.1.1 Sélection de la région

Différentes possibilités sont envisageables pour définir l'ordre de segmentation des régions. Pour chaque étape, l'utilisateur peut choisir la région r qu'il souhaite segmenter, ce qui implique qu'il définit lui-même l'ordre de la segmentation. Ces choix peuvent être faits dans le but d'améliorer les résultats de segmentation ou simplement car seule une partie des régions l'intéresse. La sélection se fait parmi la liste des régions apprises et présentes dans le graphe de connaissances *a priori*.

Il est aussi tout à fait envisageable que l'ordre d'extraction des régions soit défini automatiquement. Différents critères et algorithmes de sélection de la prochaine région à segmenter peuvent être imaginés comme par exemple, la proximité avec les régions déjà segmentées, la précision ou stabilité des relations spatiales définissant la position potentielle d'une région. Un ordre arbitrairement prédéterminé peut également être envisagé. L'analyse précise de ces différentes possibilités restent une perspectives de ce travail car faute de temps, nous n'avons pas étudié toutes ces possibilités. Seuls un ordonnancement automatique a été évalué durant ces travaux. Cet ordre automatique est basé sur la complémentarité des relations spatiales provenant des structures déjà localisées (cf section 6.3.3).

6.1.2 Positionnement de la boîte englobante de la région

L'atlas local $\{Tr, Pr\}$ correspondant à la région r choisie est sélectionné et doit être positionné sur l'image Y que l'on cherche à segmenter. En effet, l'atlas local n'est défini que sur une sous-partie de l'image et nécessite donc d'être positionné à l'intérieur de celle-ci. Pour cela, il est nécessaire de positionner les bords de la boîte englobante associé à la région r sur l'image Y . Ces bords définissent une sous-image correspondant à la zone spatiale représentant l'intérieur de l'atlas local. Ce positionnement de la boîte englobante peut être appliqué différemment selon la liberté que l'utilisateur souhaite accorder au logiciel. Dans certains cas l'utilisateur veut imposer le placement de la zone dans laquelle il faut rechercher la région considérée. Dans d'autres, l'utilisateur laissera l'algorithme utiliser les relations spatiales apprises (cf. 5.2.1 p.86) pour déterminer automatiquement la position de la boîte englobante d'une région. Les positionnements proposés automatiquement pour les régions les plus difficiles à segmenter pourront être corrigées par l'opérateur expert, avant le lancement de la classification des voxels. Les dérives de positionnement automatique sont ainsi évitées. Dans tous les cas, l'utilisateur conserve le contrôle du processus de segmentation et peut exploiter le logiciel même sur des images plus difficiles.

6.1.3 Classification des voxels

Le positionnement de la boîte englobante de la région que l'on cherche à segmenter, permet de définir un volume de voxels à étiqueter (classifier). La classification des voxels qui suit n'agit que sur ce volume. De la même manière que lors de la phase d'apprentissage, une dilatation de la boîte englobante est effectuée avant l'extraction du volume (cf. 5.1.1.1). Le pourcentage utilisé pour la marge est le même que celui utilisé lors de la phase d'apprentissage. Cette marge va permettre, d'atténuer l'impact d'éventuelles erreurs de

6.2. CLASSIFICATION DES VOXELS (À PARTIR DE L'EXPLOITATION DES ATLAS LOCAUX)

positionnement automatique ou manuel. Le sous-volume dilaté de la région r extrait finalement est noté Y_r . L'atlas local de la région à rechercher est recalé sur le volume présent à l'intérieur de l'image Y_r à segmenter. Une normalisation des intensités peut être appliquée similairement à celle appliquée lors de la création des atlas probabilistes locaux. Lors de l'utilisation des atlas probabilistes locaux, le template T_r est recalé vers la sous-image Y_r , puis la même transformation est appliquée à la carte de probabilité P_r correspondante. Le type de recalage appliqué est le même que celui utilisé lors de l'apprentissage. Il se compose d'un recalage linéaire τ_1 permettant le redimensionnement des deux sous-images, puis d'un recalage non linéaire τ_2 par B-spline. Le résultat de cette transformation permet d'initialiser le processus de classification des voxels inclus dans Y_r . Dans une approche probabiliste, la classification des voxels peut se baser sur un champ de Markov qui détermine une probabilité d'appartenance pour chaque voxel en tenant compte de l'atlas local appris préalablement. Ce processus de classification est décrit en 6.2.2 (p. 99). Lors de l'utilisation de multi-atlas locaux, chaque couple d'images $\{I_r, L_r\}$ est recalé de la même manière vers Y_r . Puis une classification des voxels appliquée à l'aide d'une méthode de vote (cf. section 6.2.3 p. 101) pour déterminer le label de chaque voxel.

6.1.4 Correction *a posteriori*

Comme nous l'avons mentionné, nous proposons d'inclure dans notre chaîne de segmentation, une étape de post-traitement permettant de corriger le premier résultat de segmentation obtenu par classification des voxels. Cette correction utilise les classifieurs *Adaboost* (C_r et $C_{\bar{r}}$) qui ont été appris lors de la phase de création du graphe de connaissances *a priori* (apprentissage des connaissances *a priori*, cf. section 5.3). La probabilité d'appartenance des voxels à la régions provenant du classifieur initial est utilisée comme entrée de ces deux classifieurs. Les classifieurs *Adaboost* fournissent finalement deux cartes de probabilité *a posteriori*, une pour l'appartenance à la région P_r^{corr} et une pour la non-région $P_{\bar{r}}^{corr}$. L'information provenant de ces cartes de probabilité, peuvent être combinées à celles provenant des régions préalablement segmentées pour prendre une décision finale pour chacun des voxels.

6.2 Classification des voxels (à partir de l'exploitation des atlas locaux)

Dans cette section, on suppose que le volume présent à l'intérieur de la boîte englobante Y_r de la région r a été défini. L'atlas local $\{T_r, P_r\}$ a été positionné et recalé sur ce volume. Un champ de Markov aléatoire caché (HMF) est utilisé pour classifier les voxels situés à l'intérieur de la boîte englobante Y_r en 2 classes : *région* et *non-région*. Des voxels présent à l'intérieur du volume Y_r ont déjà pu être classés lors de segmentations de régions précédentes. Néanmoins, leur label est ignoré lors du processus de classification de voxels jusqu'au moment de la décision finale.

6.2. CLASSIFICATION DES VOXELS (À PARTIR DE L'EXPLOITATION DES ATLAS LOCAUX)

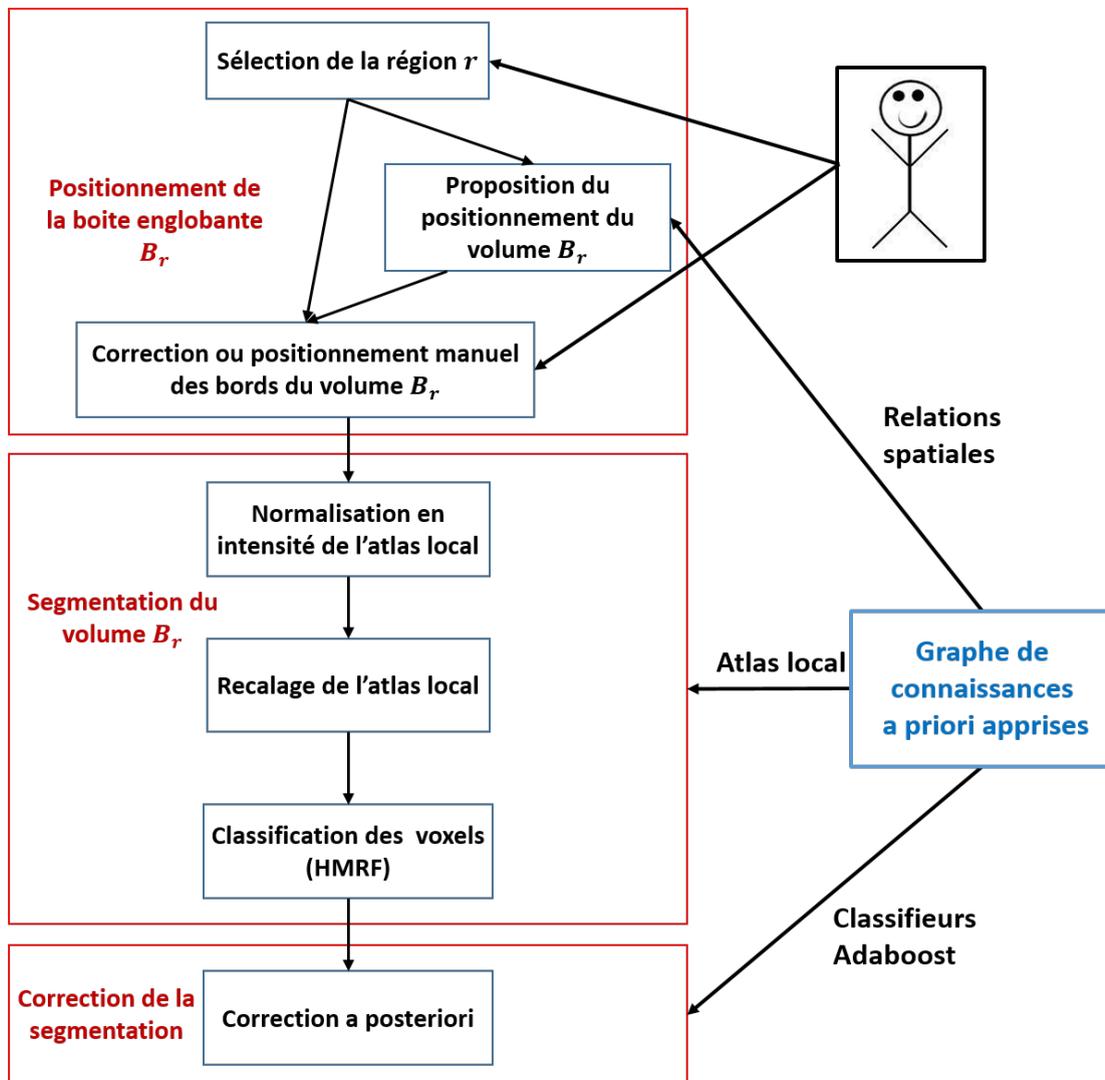


FIGURE 6.1 – Schéma représentant les différentes étapes nécessaires à la segmentation d'une région r .

6.2.1 Initialisation

Le champ de Markov utilisé pour la classification permet de classer les voxels en K classes distinctes. Bien que la décision finale ne comporte que 2 classes (*région* et *non-région*) ; il arrive souvent que la complexité des images médicales ne puisse pas être représentée par 2 classes seulement. On modélise alors l'intensité des voxels par K classes plutôt que 2. Ces K classes vont représenter le nombre de gaussiennes qui vont être nécessaires pour modéliser l'intensité des voxels à l'intérieur du volume Y_r à traiter. Dans la plupart des cas, les régions étudiées présentent des intensités assez homogènes. Une seule gaussienne suffit alors à modéliser l'intensité de la région. Les $K - 1$ autres gaussiennes permettent de modéliser l'extérieur de la région. Dans la pratique, nous verrons (cf. section 7.2.1 p. 119)

6.2. CLASSIFICATION DES VOXELS (À PARTIR DE L'EXPLOITATION DES ATLAS LOCAUX)

que le nombre de classes K nécessaires pour modéliser correctement le contenu à classifier peut varier entre 3 et 4 (une classe pour modéliser l'intensité de la région et deux ou trois autres pour modéliser l'extérieur de la région). Ces gaussiennes peuvent potentiellement être multivariées dans le cas où les images à segmenter sont des images multimodales.

Le champ de Markov est initialisé par un algorithme de type K-means où K est le nombre de classes recherchées. En sortie du K-means, la classe qui possède le plus grand nombre de voxels i dont la probabilité d'appartenance à la région r recherchée est *importante*, est étiquetée "région". La probabilité est donnée par la carte de probabilité de l'atlas correspondant à la région r qui a été recalée sur l'image à segmenter. La notion d'importance est définie actuellement par un seuil plancher, fixé empiriquement à 0.5. Toutes les autres classes sont alors étiquetées "non-région".

Pour le voxel v_i , on attribue à la classe "région" un champ externe u_i , apportant les informations de l'atlas, égal à $-\log(P_r(\tau(v_i)))$, et on attribue aux autres classes un champ externe égal à $-\log(1 - P_r(\tau(v_i)))$. $P_r(\tau(v_i))$ est la probabilité d'appartenance à la région r provenant de la carte de probabilité recalée.

Le champ de Markov est chargé de classer les voxels. Les algorithmes utilisés sont tirés de ceux décrits dans [84] et présentés dans la partie suivante.

6.2.2 Atlas probabiliste et champs de Markov cachés

Les champs de Markov cachés permettent d'introduire des notions d'interactions spatiales entre les labels. Ces relations sont liées à une notion de voisinage. Soit $\mathcal{N}(i)$ l'ensemble des voxels voisins du voxel v_i , K le nombre de classes et N le nombre de voxels de B_r . La sous-image observée est notée $Y_r = \{Y_{r_1}, \dots, Y_{r_N}\}$, la sous-image label que l'on cherche à obtenir est notée $Z_r = \{Z_{r_1}, \dots, Z_{r_N}\}$. Y_r et Z_r peuvent être considérés comme des champs aléatoires. Chaque Z_{r_i} peut prendre une valeur parmi les $\{1, \dots, K\}$ classes. Et \mathbf{z}_r et \mathbf{y}_r sont des réalisations de ces variables aléatoires.

Un champ de Markov peut être défini sous la forme d'une distribution de Gibbs par :

$$p(\mathbf{z}_r|\phi) = W^{-1} \exp(-H(\mathbf{z}_r|\phi)) \quad (6.1)$$

avec W une constante de normalisation, ϕ les paramètres du modèle et H une énergie locale restreinte à un certain voisinage autour de chaque voxel v_i . On se place ici dans la situation d'un champ de Markov à données incomplètes avec Y les observations et Z les données cachées liées aux observations Y . En utilisant la règle de Bayes et le fait que les observations Y sont indépendantes conditionnellement sachant Z , on peut définir la probabilité *a posteriori* d'avoir une labellisation z_r sachant l'image y_r et les paramètres ϕ par :

$$p(\mathbf{z}_r|\mathbf{y}_r, \phi) = W^{-1} \exp(-H(\mathbf{z}_r|\mathbf{y}_r, \phi)) \quad (6.2)$$

6.2. CLASSIFICATION DES VOXELS (À PARTIR DE L'EXPLOITATION DES ATLAS LOCAUX)

avec H la fonction d'énergie locale définie comme une somme de potentielle telle que :

$$H(\mathbf{z}_r | \mathbf{y}_r, \phi) = - \sum_{i \in V} [\alpha_i u_i + \beta \sum_{j \in N(i)} z_{r_i} \cdot z_{r_j} + \log p(y_{r_i} | z_{r_i}, \phi)] \quad (6.3)$$

Le paramètre α_i permet de régler l'influence du champ externe u_i apportant les informations de l'atlas. Le paramètre β va régler l'importance des labels voisins pour la segmentation d'une région. $p(y_{r_i} | z_{r_i}, \phi)$ décrit la probabilité *a priori* modélisant l'intensité des voxels sachant la classe à laquelle celui-ci appartient.

Des expériences ont permis de définir empiriquement les paramètres permettant d'optimiser les résultats Dice des segmentations. Ces expériences seront décrites à la section 7.2.1. Finalement, les paramètres du champ de Markov sont fixés à 0.05 pour le paramètre pondérant l'importance des voisins et à $0.75 + 2 * \mathcal{H}_i$ pour le paramètre pondérant l'importance de l'atlas. \mathcal{H}_i est ici l'entropie liée à la distribution de probabilité *a posteriori* : $\mathcal{H}_i = \sum_{k=1, \dots, K} p(z_{r_i} | y_{r_i}, \phi) \log(p(z_{r_i} | y_{r_i}, \phi))$.

La modélisation *a priori* de l'intensité des voxels se traduit par une gaussienne potentiellement multivariée telle que :

$$p(y_i | z_i, \phi) = \frac{1}{\sqrt{2\pi} |\Sigma_k^{1/2}|} \exp\left(-\frac{1}{2} (y_i - \mu_i)^T \Sigma^{-1} (y_i - \mu_i)\right) \quad (6.4)$$

Les paramètres des gaussiennes μ_k et Σ_k sont estimés à l'aide d'un algorithme EM (Expectation Maximization [32]). Celui-ci permet de trouver les paramètres qui maximisent la vraisemblance des observations. L'algorithme est stoppé lorsque un nombre d'itération fixé est atteint. Une approximation en champ modal est utilisée pour éviter l'explosion combinatoire lors de l'étape d'estimation. A la fin des itérations, une probabilité d'appartenance à chaque classe est disponible. Les voxels sont étiquetés vers le classe pour laquelle la probabilité d'appartenance est la plus forte ("région" pour la classe région, "non région" pour l'une des classes "non région"). L'attribution des labels se fait par la relation qui s'exprime telle que :

$$z_{r_i} = \operatorname{argmax}_{k \in [1, \dots, K]} p(z_{r_i} = k | y_i, \phi) \quad (6.5)$$

Les voxels possédant le label "région" sont définitivement attribués à la région r . Les voxels classés vers le label "non-région" pourront être reclassés lors d'une future segmentation.

La segmentation de la sous-image est transférée dans l'image globale. Si R' est l'ensemble des régions parmi R déjà segmentées et que la $S_{R'}$ est la segmentation à l'étape précédente. S_{new} est la nouvelle segmentation de l'image, décrite par :

$$S_{new_i} = \begin{cases} S_{R'_i} & \text{si } S_{R'_i} \neq 0 \\ 0 & \text{si } S_{R'_i} = 0 \text{ et } z_{r_i} = \text{"non-region"} \\ r & \text{si } S_{R'_i} = 0 \text{ et } z_{r_i} = \text{"region"} \end{cases} \quad (6.6)$$

Les voxels qui étaient déjà classés comme appartenant à une région lors d'une précédente segmentation ne sont pas modifiés. Ceci implique que la segmentation d'une région est définitive et que la segmentation d'une nouvelle région (différente) ne modifiera pas la segmentation d'une précédente région.

Notons qu'il est tout à fait possible de conserver les différentes probabilités d'appartenir à chaque classe *région* renvoyées par les HMF. Ces informations peuvent être plus riches que la segmentation en labels. Dans la suite du document, la probabilité $p(\mathbf{z}_r = \text{region} | \mathbf{y}_r, \phi)$ sera notée $P_r^{classif}$.

6.2.3 Multi-Atlas et méthode de vote

Le schéma global de segmentation reste le même dans le cas de la segmentation avec des multi-atlas locaux. Seul le champ de Markov est remplacé par une méthode de vote. La boîte englobante est définie et le volume Y_r est extrait comme pour les atlas probabilistes locaux. Une normalisation des intensités peut être appliquée aux différents templates $(I_r^n, n = \{1, \dots, N_r\})$ de chaque atlas local. Puis, ils sont tous recalés vers le volume Y_r et la même transformation τ_n^r (linéaire + non linéaire) est appliquée à la vérité terrain binaire associée L_r^n .

Une méthode de vote est ensuite effectuée sur la sous-image Y_r de la même manière qu'elle pourrait être effectuée sur l'image complète. Si plus de la moitié des atlas votent pour un même label (cf. section 4.4.1) à un voxel alors ce label "région" est attribué au voxel. Cette attribution des labels se traduit par la relation suivante :

$$z_{r_i} = \begin{cases} \text{"region"} & \text{si } \sum_{n=1}^{N_r} \delta(L_n(\tau_n^r(v_i), r)) \geq \frac{N}{2} \\ \text{"non-region"} & \text{si } \sum_{n=1}^{N_r} \delta(L_n(\tau_n^r(v_i), r)) < \frac{N}{2} \end{cases} \quad (6.7)$$

La fonction $\delta(L_n(\tau_n^r(v_i), r))$ est égale à 1 si l'image label L_n recalée associe la région r au voxel v_i .

La méthode de vote sans pondération est le processus le plus simple pour utiliser les multi-atlas. Cependant, il est possible d'appliquer facilement une méthode de vote avec pondération globale afin d'améliorer les résultats de segmentation. Dans cette situation, le vote de chaque atlas A_n est pondéré par un terme w_n représentant sa ressemblance avec l'image que l'on souhaite segmenter. La similarité de A_n avec B_r est décrite par la somme des distances au carré entre T_n et B_r après la normalisation. Si S_n décrit cette similarité alors on définit simplement w_n tel que :

$$w_n = \frac{1}{S_n} \quad (6.8)$$

Et la segmentation finale est fournie par :

$$z_{r_i} = \begin{cases} \text{"region"} & \text{si } \sum_{n=1}^{N_r} w_n \delta(L_n(\tau_n^r(v_i), r)) \geq \frac{N}{2} \\ \text{"non-region"} & \text{si } \sum_{n=1}^{N_r} w_n \delta(L_n(\tau_n^r(v_i), r)) < \frac{N}{2} \end{cases} \quad (6.9)$$

Que ce soit pour la méthode de vote avec pondération ou sans pondération, la segmentation de l'image globale est obtenue de la même manière que lors de l'utilisation des atlas probabilistes (cf. eq. 6.6).

6.3 Positionnement automatique des régions d'intérêt

Dans notre chaîne de traitement, nous proposons un mécanisme de positionnement automatique des boîtes englobantes, tout en laissant à l'utilisateur expert le choix de le corriger en cas de besoin. On suppose pour cela qu'au moins une région a déjà été segmentée. Cette "base" permet de placer les autres régions. L'utilisateur guide alors la segmentation simplement en indiquant quelle région (présente dans la graphe G) l'algorithme doit rechercher. Cette région est notée r .

6.3.1 Sélection des informations disponibles

La première étape du positionnement automatique consiste à sélectionner les informations disponibles dans le graphe appris G . Si R' est l'ensemble des régions parmi R déjà segmentées alors chacune de ces régions peut fournir des informations. Les régions servent alors de référence pour le placement de la boîte englobante de la région Y_r . On note ici B_r^j , avec $j \in \{1, \dots, 6\}$, les 6 bords de la boîte englobante du volume Y_r au sein de l'image Y . Dans le graphe G , les arcs provenant des nœuds associés aux régions de R' et pointant vers le nœud associé à r contiennent les intervalles de distance qui peuvent être utilisés.

Le positionnement se fait indépendamment pour chaque bord du volume Y_r . La suite présente les différents calculs effectués dans le cas de la direction \vec{x} . Le même processus est appliqué sur les autres directions et les autres bords.

Chaque région r' de R' fournit deux intervalles sur le positionnement du bord recherché. L'intervalle $[Min_{dr_{r'}^{x11}}, Max_{dr_{r'}^{x11}}]$ présenté dans la section 5.2.1 décrit la distance relative entre le bord n° 1 de la région r' et le bord n° 1 de la région r dans la direction \vec{x} (Idem du bord n° 2 vers n° 1 avec $[Min_{dr_{r'}^{x21}}, Max_{dr_{r'}^{x21}}]$). Ensuite, chacun de ces intervalles doit être transformé en position absolue dans l'image à segmenter. Si $x_{r'}^i$ est la position du bord i (1 ou 2) et que $dim_{r'}$ est la dimension de r' dans la direction \vec{x} alors il est possible de déterminer un intervalle de position $[Xmin_{r'}^{ij}, Xmax_{r'}^{ij}]$ défini tel que :

$$\begin{aligned}
 Xmin_{r'}^{1j} &= x_{r'}^i + dim_{r'} * Min_{dr_{r'}^{x1j}} & j \in (1, 2) \\
 Xmax_{r'}^{1j} &= x_{r'}^i + dim_{r'} * Max_{dr_{r'}^{x1j}} & j \in (1, 2)
 \end{aligned}
 \tag{6.10}$$

avec $Xmin_{r'}^{1j}$ la position minimale du bord j (de r) provenant du bord n° 1 dans la direction \vec{x} de r' . Le bord n° 2 de r' fournit lui aussi un intervalle $[Xmin_{r'}^{2j}, Xmax_{r'}^{2j}]$ de position absolue dans l'image provenant de la position $x_{r'}^2$.

Lorsque toutes les informations relatives aux régions de R' ont été exploitées et replacées dans le repère de l'image, on peut les combiner afin d'obtenir une position finale des bords de la région r .

6.3.2 Fusion des intervalles

Ici, la fusion des informations fait référence au fait de combiner les différents intervalles que l'on a à disposition et qui proviennent de sources (régions) différentes afin d'obtenir une décision finale de positionnement des bords. Pour combiner les différents intervalles, les intervalles sont représentés à l'aide de la fonction *porte* Π définie par :

$$\Pi_{r'}^{ij}(x) = \begin{cases} 1 & \text{si } x \in [Xmin_{r'}^{ij}, Xmax_{r'}^{ij}] & (i, j) \in (1, 2)^2 \\ 0 & \text{sinon.} & (i, j) \in (1, 2)^2 \end{cases}
 \tag{6.11}$$

Ces intervalles pourront être sommés afin d'avoir une fonction de positionnement représentant l'ensemble des informations disponibles. Pour cela, une amplitude (ou une pondération) doit être définie pour chaque intervalle. Ainsi, plus une relation est précise plus elle aura une pondération importante par rapport aux autres. La précision d'une relation est ici évaluée par la taille de l'intervalle. Un intervalle est de faible taille si la disparité des distances relatives observées dans la base d'apprentissage est faible également. La pondération attribuée à chaque intervalle est donc inversement proportionnelle à sa longueur et définie par :

$$\mathcal{W}_{r'}^{ij} = \frac{1}{Xmax_{r'}^{ij} - Xmin_{r'}^{ij} + \gamma}
 \tag{6.12}$$

où γ est un paramètre strictement positif, permettant aux poids de garder une valeur finie. Cette situation peut se produire lorsqu'une seule image est apprise ou lorsque toutes les images de la base d'apprentissage fournissent la même distance relative pour une certaine relation.

L'espérance de la somme des fonctions portes pondérées est utilisée pour prendre la décision de positionnement. La décision finale du bord n° j de la région r est alors calculée telle que :

$$B_r^j = E\left(\sum_{r' \in R'} \mathcal{W}_{\Pi_{r'}^{1j}} \Pi_{r'}^{1j}(x) + \sum_{r' \in R'} \mathcal{W}_{\Pi_{r'}^{2j}} \Pi_{r'}^{2j}(x)\right)
 \tag{6.13}$$

avec E l'espérance. Ce processus est répété six fois pour chaque région pour définir le volume B_r . Le positionnement est ensuite affiché à l'utilisateur qui pourra valider ou corriger le résultat.

Gaussiennes de distances relatives :

Si les distances relatives sont stockées sous forme de gaussiennes, le même processus est appliqué. Les gaussiennes de distances relatives sont transformées en gaussiennes de position dans l'image à segmenter.

$$\begin{aligned} X\mu_{dr_{r_1r_2}^{x_{ij}}} &= dim_x * \mu_{dr_{r_1r_2}^{x_{ij}}} \\ X\sigma_{dr_{r_1r_2}^{x_{ij}}} &= dim_x * \sigma_{dr_{r_1r_2}^{x_{ij}}} \end{aligned} \quad (6.14)$$

Dans le cas des gaussiennes, une amplitude est déjà associée à chaque fonction (elles sont normalisées). La pondération est déjà existante par l'intermédiaire de l'écart type de la gaussienne. L'ensemble des gaussiennes de position réelle est donc sommé de la même manière que pour les intervalles. Puis, l'espérance de la fonction est utilisée pour obtenir une décision finale sur la position du bord.

$$B_r^j = E\left(\sum_{r' \in R'} \mathcal{N}(X\mu_{dr_{r_1r_2}^{x_{1j}}}, X\sigma_{dr_{r_1r_2}^{x_{1j}}}) + \sum_{r' \in R'} \mathcal{N}(X\mu_{dr_{r_1r_2}^{x_{2j}}}, X\sigma_{dr_{r_1r_2}^{x_{2j}}})\right) \quad (6.15)$$

avec $\mathcal{N}(\mu, \sigma)$ la loi normale de moyenne μ et d'écart type σ .

6.3.3 Ordonnement automatique de la segmentation

Les informations provenant des relations spatiales peuvent aussi être un critère pour choisir l'ordre de segmentation des régions. En effet, l'ordre de segmentation joue un rôle important lorsque le positionnement automatique est utilisé. Les relations spatiales étant dépendantes des segmentations précédentes, le choix de la région et la qualité de sa segmentation future guideront les prochaines segmentations. C'est pourquoi, dans un premier temps, il peut être intéressant de traiter en priorité les régions facilement détectables. La qualité de la segmentation sera probablement bonne et les relations spatiales qui vont provenir de cette région seront, elles aussi, probablement de bonne qualité. La qualité d'une segmentation, dans notre situation, dépend tout particulièrement de la qualité du positionnement du volume Y_r . Donc l'objectif est de détecter les régions de forme simple et visible dans un premier temps (en utilisant le positionnement automatique). L'un des critères pour permettre le choix de la prochaine région à localiser peut se baser sur la fiabilité des relations spatiales.

Les régions potentiellement sélectionnables font partie des régions r non segmentées de $R \setminus R'$ et où il existe au moins une relation spatiale entre r et l'une des régions de R' .

Les informations spatiales disponibles sont considérées et les intervalles sont utilisés pour déterminer les positions dans le repère de l'image (cf. section précédente). Indépendamment pour chaque bord et chaque région $r \in R \setminus R'$, la somme pondérée des intervalles que l'on a à disposition est calculée. Le calcul est similaire à l'équation 6.13. L'écart type est ensuite calculée pour décrire le caractère certain (la fiabilité, stabilité) de la position des bords. L'écart type des 6 bords est sommée ce qui fournit une valeur de certitude sur la position générale de chaque région. La région qui possède la somme des écarts types la plus faible est sélectionnée comme prochaine région à segmenter lorsque l'ordre est calculé automatiquement. Comme nous l'avons déjà indiqué, d'autres stratégies de sélection sont envisageables mais n'ont pas pu être étudiées jusqu'à présent.

6.4 Détails sur l'étape de post-correction

Une fois la segmentation basée atlas d'une région terminée (après avoir positionné et classifié les voxels), il est possible d'appliquer une étape de correction *a posteriori* du résultat obtenu conformément à l'idée initiale proposés dans cite() et que nous avons adaptée à notre méthode opératoire (local et incrémental).

La correction est effectuée en deux étapes principales. La figure 6.2 décrit le schéma général de la correction *a posteriori* que nous proposons. Dans un premier temps, des classifieurs sont utilisés pour fournir deux nouvelles cartes de probabilité *a posteriori* *région* et *non-région*. Puis, dans un deuxième temps, une décision est prise en fusionnant l'ensemble des cartes de probabilité *a posteriori* disponibles.

6.4.1 Mode d'utilisation des classifieurs

Les classifieurs *Adaboost* ont été appris lors de la phase d'apprentissage (cf. section 5.3). La probabilité d'appartenir à la classe *région* en sortie du champ de Markov ($P_r^{classif}$) est conservée pour être fournie en entrée du classifieur *Adaboost* C_r ainsi que le classifieur *Adaboost* $C_{\bar{r}}$. Les voxels qui vont être utilisés lors de cette classification sont sélectionnés de la même façon que lors de l'apprentissage. Les voxels utilisés par le classifieur C_r sont les voxels présents dans une dilatation du volume provenant du seuillage à 0.5 de la carte de probabilité provenant du HMF. Les voxels utilisées par le classifieur $C_{\bar{r}}$ sont les voxels présents dans l'intersection entre la dilatation ce même volume et une dilatation de l'extérieur du volume. Les voxels sont ensuite classifiés par les deux classifieurs C_r et $C_{\bar{r}}$. Chaque classifieur fournit une probabilité *a posteriori* notée $P_r^{corr}(v)$ et $P_{\bar{r}}^{corr}(v)$ pour le voxel v . Les cartes de probabilité ainsi générées sont conservées pour pouvoir être utilisées lors de la segmentation d'autres régions.

6.4.2 Fusion des cartes de probabilité

La fusion incrémentale des cartes de probabilité obtenues est réalisée comme suit :

6.4. DÉTAILS SUR L'ÉTAPE DE POST-CORRECTION

D'abord, pour chaque voxel de la région r en cours de traitement, la région avec la probabilité *a posteriori* la plus élevée est sélectionnée et notée r_{max} .

$$r_{max}(v) = \arg \max_{r \in R'} P_r^{corr}(v)$$

avec R' l'ensemble des régions déjà segmentées (incluant la nouvelle région que l'utilisateur cherche à segmenter).

Ensuite, la probabilité d'appartenance à la région r_{max} est comparée à la probabilité d'appartenir à la *non-region* $P_{r_{max}}^{corr}$. La segmentation finale est calculée telle que :

$$S_{new}(v) = \begin{cases} r_{max} & \text{si } P_{r_{max}}^{corr}(v) > P_{r_{max}}^{corr}(v) \\ 0 & \text{si } P_{r_{max}}^{corr}(v) \leq P_{r_{max}}^{corr}(v) \end{cases} \quad (6.16)$$

Avec S la segmentation de l'image qui va être montré à l'utilisateur.

Le fait de conserver les cartes de probabilités *a posteriori* des précédentes régions permet de corriger la segmentation de ces régions lors des segmentations futures. Si des voxels ont été étiquetés vers une certaine région r' avec une probabilité p' et qu'une nouvelle région r a une probabilité p supérieure à p' alors ces voxels seront reclassifiés vers r .

6.4.3 Correction de l'adjacence

Le processus incrémental de segmentation local que nous proposons peut impliquer quelques problèmes d'adjacence à l'intérieur des résultats de segmentation locaux obtenues. De plus, la correction fonctionne lorsque deux régions se superposent mais fonctionne moins bien lorsque deux régions semblent éloignées alors que celles-ci devraient être adjacentes. Cela se traduit par de fréquents voxels non-classés à la frontière entre les régions.

Pour réduire le nombre et voxels non classés entre les régions adjacentes, nous proposons de tirer partie de cette étape de post correction afin de réduire les chances qu'un voxel présent entre deux régions adjacentes appartienne au fond. Pour cela, la probabilité d'appartenir au fond est diminuée lors de la décision définie par l'équation (6.16). L'idée est de favoriser les voxels à être classés comme région afin de corriger les espaces vides entre les régions adjacentes. Pour chaque voxel qui n'est pas labellisé et qui possède plus de deux régions différentes dans son voisinage, les coefficients suivants sont calculés :

$$\begin{cases} K1 & = \frac{Nb_{class}}{Nb_{patch}} \\ K2 & = (1 - (|Nb_{R1} - \frac{Nb_{class}}{2}| + (|Nb_{R2} - \frac{Nb_{class}}{2}|))) \end{cases} \quad (6.17)$$

avec r_1 et r_2 les deux régions les plus représentées à l'intérieur du patch. Nb_{class} est le nombre de voxel qui ont été classés comme région dans le patch. Nb_{patch} est le nombre de voxels total présents dans le patch. Nb_{R1} et Nb_{R2} est le nombre de voxels à l'intérieur

du patch qui appartiennent à r_1 et r_2 respectivement. Le coefficient K_1 est important lorsque le voxel est entouré de beaucoup de voxels affectées à des régions, quelque soit le label de ces régions. Le coefficient K_2 est lui plutôt lié à l'indécision entre deux régions. Si le nombre de voxel de r_1 et r_2 est important et similaire à la fois alors ce coefficient est important. Lorsque l'un de ces deux coefficients est calculé, la probabilité *a posteriori* d'appartenir au fond $P_{r_{max}}$ lié à la région r_{max} est pondérée par le coefficient pour obtenir une segmentation telle que :

$$S_{new}(v) = \begin{cases} r_{max} & \text{si } P_{r_{max}}^{corr}(v) > (P_{r_{max}}^{corr}(v) * (1 - K_1)) \\ 0 & \text{si } P_{r_{max}}^{corr}(v) \leq (P_{r_{max}}^{corr}(v) * (1 - K_1)) \end{cases} \quad (6.18)$$

La même méthode peut être utilisée pour le coefficient K_2 . Nous verrons que ces deux coefficients fournissent des résultats de qualité similaire.

La figure 6.3, illustre un résultat qui pourrait se produire sur une partie de l'image. Dans cet exemple, deux régions sont segmentées l'une après l'autre. À chaque étape, la post-correction est appliquée afin d'améliorer le résultat de segmentation. Ces images représentent le résultat qui serait montré à l'utilisateur à chaque instant, en pratique les résultats intermédiaires ne sont pas présentés aux utilisateurs.

À noter que l'utilisation de cette post-correction permet à la segmentation de ne plus être définitive comme c'était le cas précédemment (sans la post-correction). La segmentation d'une nouvelle région peut alors modifier les précédentes.

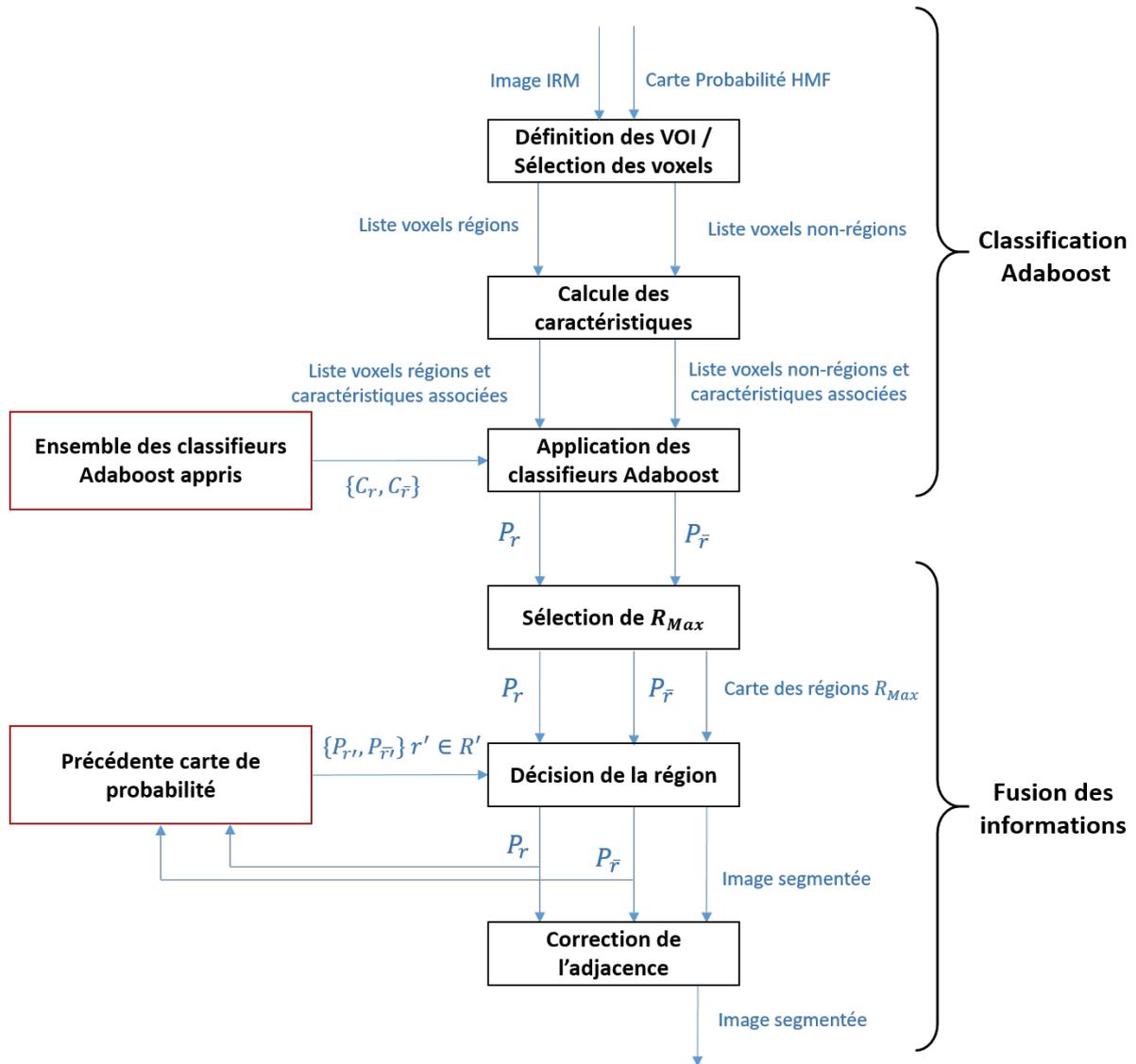


FIGURE 6.2 – Étapes de correction *a posteriori*

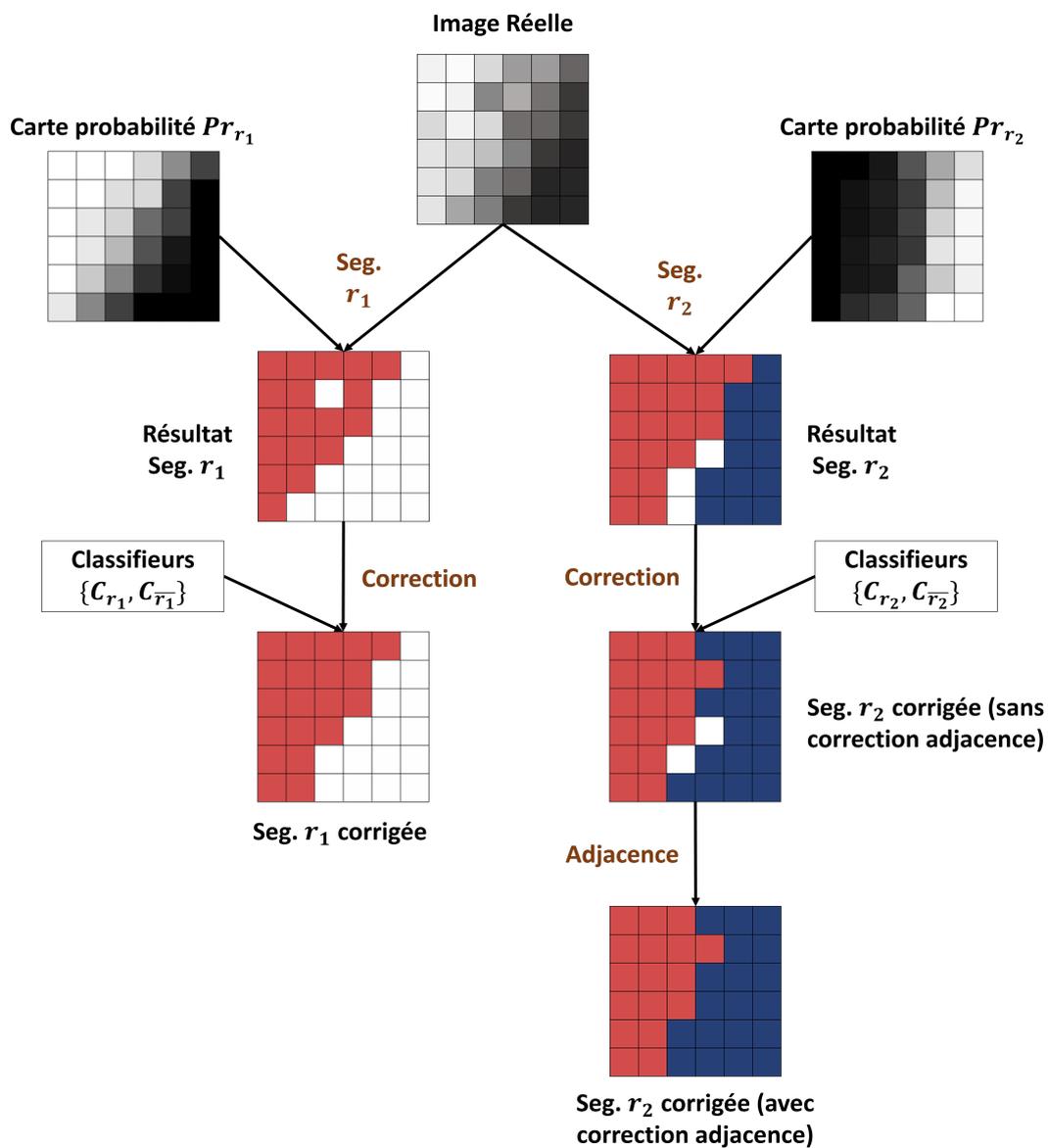


FIGURE 6.3 – Exemple de résultat de correction sur une partie d'image de 6x6 pixels. Deux régions R_1 et R_2 sont segmentées l'une après l'autre. Les processus $Seg.R_1$ et $Seg.R_2$ représentent les segmentations avec atlas locaux. Les processus $Correction$ représentent l'application des classifieurs Adaboost (avec et sans correction du problème d'adjacence).

6.5 Bilan

Nous proposons dans ce chapitre un processus original de segmentation interactive et incrémentale d'images 3D. Ce processus est guidé par des atlas locaux et des relations spatiales appris sur une base d'images d'apprentissage et structurés sous forme d'un graphe.

L'utilisation d'un champ de Markov décrit comment exploiter les informations provenant d'atlas probabilistes locaux. Ce champ de Markov est appliqué sur une sous-image correspondant au positionnement de l'atlas local. Dans le cas de l'exploitation de multi-atlas locaux, nous avons décrit simplement l'utilisation d'une méthode de vote afin d'obtenir une classification des voxels.

Nous avons aussi décrit un mode de positionnement automatique de nos atlas locaux à l'aide de relations spatiales préalablement apprises. Les distances relatives sont d'abord transformées en positionnement absolu dans l'image. Puis, l'ensemble des informations disponibles est fusionné pour fournir un positionnement. Chaque bord de la boîte englobante est positionné indépendamment des autres. Si les relations spatiales ne sont pas disponibles ou que le positionnement n'est pas correct, l'utilisateur a toujours la possibilité de positionner lui-même la boîte englobante de la région. Une méthode de calcul de l'ordonnement des segmentations des régions a aussi été présentée basée sur la "cohérence" entre les relations spatiales disponibles.

Enfin une étape de post-correction a été présentée dans laquelle des classifieurs Ada-boost utilisent comme caractéristique la probabilité d'appartenance à la région, mais aussi les informations provenant de l'image réelle 3D et la position par rapport au centroïde de la région. Chaque classifieur fournit une probabilité d'appartenance *a posteriori* qui est utilisée pour prendre une décision finale sur la segmentation. Nous proposons d'appliquer aussi une correction sur ces cartes de probabilité afin de limiter les problèmes d'adjacence pouvant se produire dans le cas de notre segmentation incrémentale.

Quatrième partie

Intérêts de l'approche proposée et applications

Chapitre 7

Évaluation des performances et validation

7.1 Données et outils utilisés pour les expériences

Différentes bases d'images ont été utilisées lors des expérimentations. Les modalités d'acquisition ainsi que les organes étudiés varient pour chaque base. Les structures anatomiques à rechercher peuvent aussi être différentes pour chaque base d'images car les vérités terrains disponibles associées ne sont pas toujours identiques. Le choix des bases d'images utilisées lors des expériences dépend des caractéristiques et paramètres que l'on désire étudier. Nous allons décrire dans cette section le type, le nombre d'images, les caractéristiques et les structures anatomiques associées à chaque base.

7.1.1 Bases d'images

7.1.1.1 Bases d'images de cerveaux humains T1 (MICCAI'12)

La base d'images MICCAI'12 est la base d'images ayant été exploitée lors du challenge MICCAI [57] effectué en 2012 sur la segmentation multi-atlas d'images IRM de cerveaux. Ces images IRM font partie des images présentes dans la base d'images ORASIS [62]. La base d'apprentissage est composée de 15 images IRM T1 du cerveau humain. La base de test est composée de 20 images IRM T1 du cerveau humain dont certaines provenant de personnes âgées ou atteintes du syndrome d'Alzheimer contrairement aux images contenues dans la base d'apprentissage. Les vérités terrains associées sont composées de 140 régions décrivant à la fois régions sous-corticales mais aussi les régions corticales. La figure 7.1 fournit un exemple de coupe coronale de cette base. Cependant lors de la plupart des expériences qui suivront, 13 structures anatomiques seront segmentées. La liste des 13 régions qui seront le plus souvent étudiées est notée dans le tableau 7.1 et est en partie illustrée sur la figure 7.2. Ces structures anatomiques sont des structures sous-corticales ou internes du cerveau qui sont plus stables que les structures corticales. De plus, elles sont en général plus étudiées que d'autres régions car elles peuvent apporter des informations sur

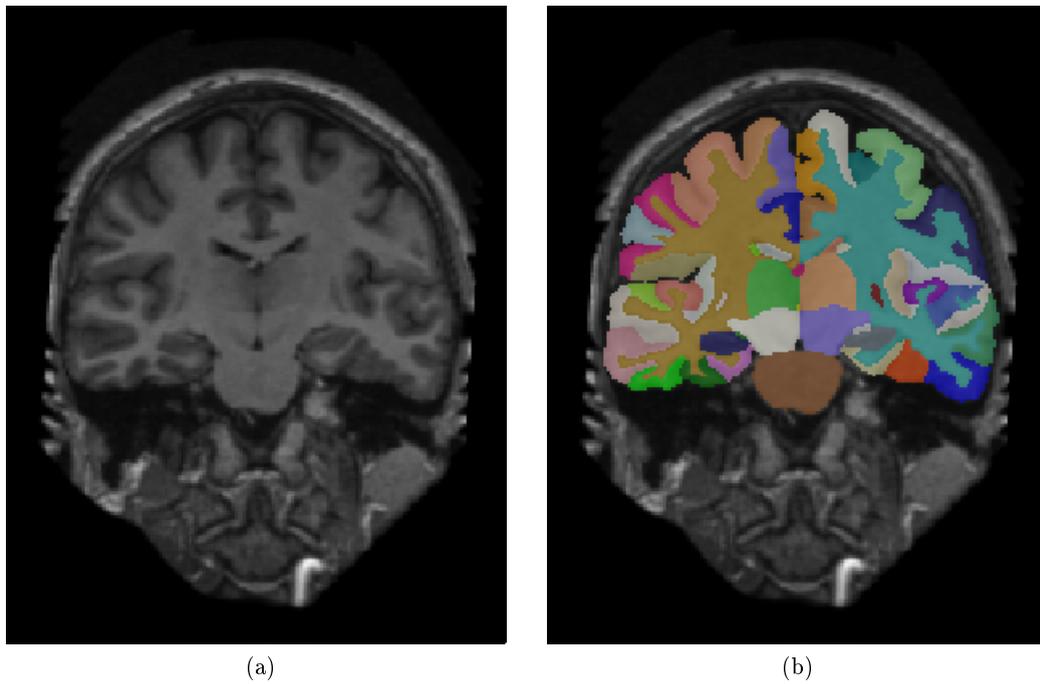


FIGURE 7.1 – Coupe coronale d’une image (a) de la base d’apprentissage et de l’image (b) labellisée correspondante

différentes pathologies. Les résultats provenant des régions bilatérales (gauche et droite) seront de temps en temps fusionnés.

TABLE 7.1 – Liste des 13 régions segmentées lors des expériences sur les images de cerveaux humains

Structures anatomiques
Noyau caudé gauche
Noyau caudé droit
Pallidum gauche
Pallidum droit
Thalamus gauche
Thalamus droit
Putamen gauche
Putamen droit
Ventricule gauche
Ventricule droit
Hippocampe gauche
Hippocampe droit
Tronc cérébral

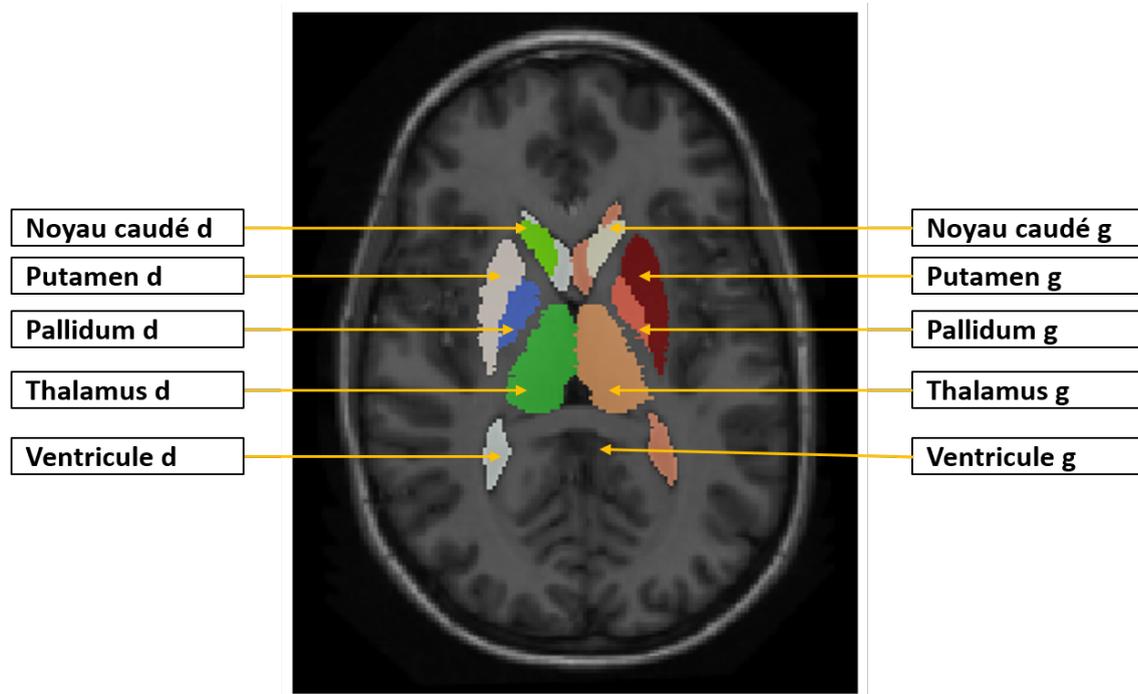


FIGURE 7.2 – Coupe transversale d'une image IRM et des labels de structures anatomiques du cerveau humain (10 des 13 structures sont représentées)

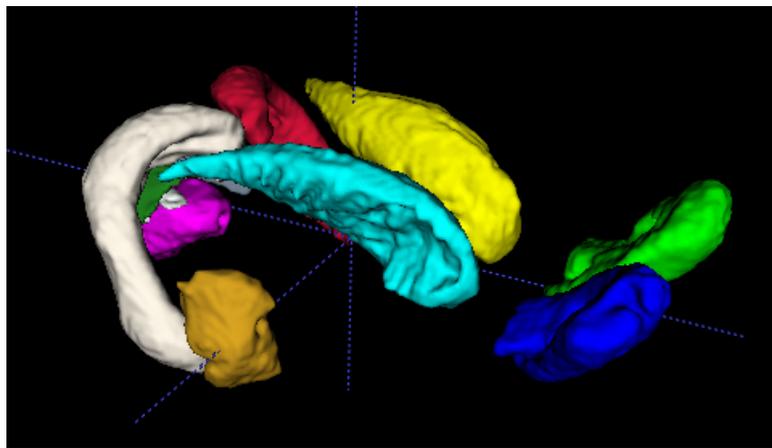
7.1.1.2 Base d'images de cerveaux de brebis ex vivo (NeuroGeoEx)

La base d'images NeuroGeoEx contient des images de cerveaux de brebis fournies par l'INRA et NeuroSpin dans le cadre du projet NeuroGeo. Ce sont des cerveaux ex vivo, le cerveau des brebis a donc été préalablement fixé puis prélevé afin d'être imagé. Six images de cerveaux de brebis ont été acquises dans un IRM 7T 50mT/m de l'institut NeuroSpin¹. La résolution spatiale des images est de $0,3 \times 0,3 \times 0,3$ mm. Ces images ont ensuite été labélisées par un chercheur en NeuroAnatomie avec 16 structures anatomiques chacune. Parmi l'ensemble des 16 structures anatomiques, 13 structures ont été particulièrement étudiées. En effet, lors d'une expérience utilisateurs que nous avons organisé pour évaluer notre chaîne de traitement et par soucis de temps, 13 structures seulement ont été segmentées. Les expériences qui suivront utilisant cette base d'images, se concentrent donc seulement sur ces 13 structures. La liste des structures anatomiques est présentée dans le tableau 7.2 et la figure 7.3 illustre ces différentes structures anatomiques. Les types de structures anatomiques recherchées ici sont assez variés. Certaines structures sous-corticales comme les Noyaux caudés et les hippocampes sont de taille importante et assez visible comme dans le cerveau humain. D'autres structures internes comme les colliculi sont de taille et de contraste beaucoup plus faible. Enfin, les bulbes olfactifs sont des structures externes du cerveau possédant des formes beaucoup plus variables à l'instar de la plupart des structures corticales.

1. http://joliot.cea.fr/drf/joliot/Pages/Entites_de_recherche/NeuroSpin.aspx

TABLE 7.2 – Liste des régions segmentées lors de l’atelier utilisateur du projet NeuroGeo sur les images de cerveau de brebis T2 ex vivo

Structures anatomiques
Bulbe olfactif gauche
Bulbe olfactif droit
Noyau caudé gauche
Noyau caudé droit
PAG
Amygdale gauche
Amygdale droit
Colliculi supérieur optique gauche
Colliculi supérieur optique droit
Colliculi supérieur moteur gauche
Colliculi supérieur moteur droit
Hippocampe gauche
Hippocampe gauche

FIGURE 7.3 – Structures anatomiques du cerveau de brebis qui sont étudiées et segmentées dans le cadre des expériences sur les images IRM de cerveau de brebis (Image *B1*)

7.1.1.3 Base d’images IRM et CT scan de cœurs (MM-WHS’17)

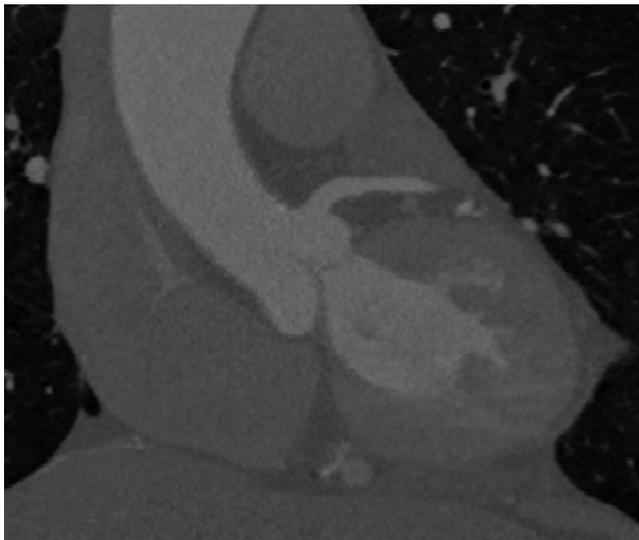
Des images 3D de cœurs ont été fournies pour le concours MM-WHS’17 (*Multi modality whole heart segmentation*) [7, 111, 110]. Cette base d’images est composée de 20 images de CT scans ainsi que de 20 images IRM de cœurs. Une vérité-terrain associée à chaque image est fournie et composée de 8 structures anatomiques (cf. tableau 7.3 et figure 7.5). Les structures du cœur sont de taille sensiblement plus importante que celles du cerveau qui ont été détectées dans les sections précédentes. Certaines de ces structures telles que les artères sont de forme plus variable que les structures anatomiques sous-corticales du cerveau.

7.1. DONNÉES ET OUTILS UTILISÉS POUR LES EXPÉRIENCES

Les images scanner (cf. figure 7.4a) possèdent une résolution élevée et les structures anatomiques y sont bien visibles. Dans le cas des images IRM (cf. figure 7.4b), la résolution de certaines images est beaucoup plus faibles ($1.3 \times 1.3 \times 1.3$ mm par voxel) et certains artefacts rendent difficile la détection de certaines structures particulièrement dans le cas des artères.

TABLE 7.3 – Liste des régions segmentées lors des expériences sur le cœur humain

Structures anatomiques
Ventricule gauche
Ventricule droit
Atrium gauche
Atrium droite
Myocardium du ventricule gauche
Aorte (ascendante)
Aorte pulmonaire



(a) Images CT scan



(b) Images MRI

FIGURE 7.4 – Image (a) scanner de cœur et image (b) IRM du cœur provenant de la base MM-WHS'17

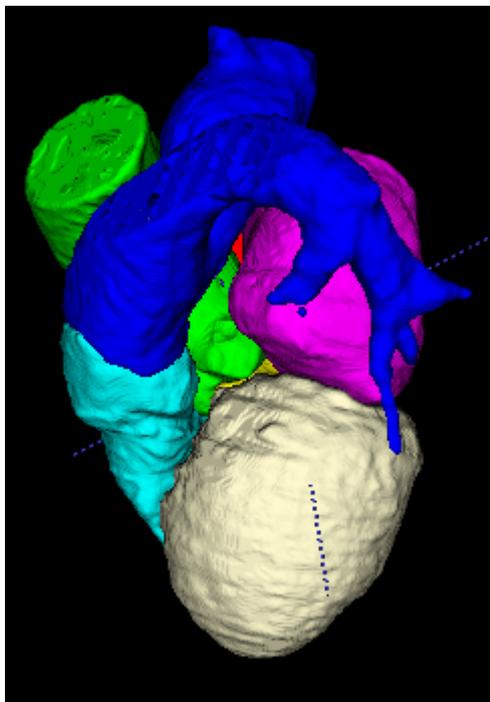


FIGURE 7.5 – Structures anatomiques du cœur liées aux images de la base CT scan du challenge MM-WHS'17

7.1.2 Validation des segmentations

Dans le but d'évaluer la qualité d'une segmentation mais aussi la qualité des algorithmes proposés, des indices de similarités doivent être utilisés. La plupart des expériences présentées dans ce document utilisent le ratio Dice [89] comme critère d'évaluation de la qualité des résultats. Si la segmentation effectuée est notée S et que la vérité associée est notée L alors le ratio Dice est défini tel que :

$$Dice = 2 \frac{card(S \cap L)}{card(L) + card(S)} \quad (7.1)$$

$card(S \cap L)$ représentant le nombre de voxel où la segmentation et la vérité-terrain sont en accord. Ce qui peut aussi être exprimé en terme de *faux positif* (VP), *faux négatif* (FP) et *vrai positif* (FN) :

$$Dice = 2 \frac{VP}{2VP + FP + FN} \quad (7.2)$$

Ce critère est l'un des critères les plus utilisés pour évaluer la qualité des segmentations en imagerie médicale.

7.2 Robustesse de la méthode vis à vis de ses paramètres

La méthode proposée dans ces travaux possède un certain nombre de paramètres. La figure 7.6 liste les paramètres dont l'impact sera étudié dans ce chapitre. Cette section a pour but de justifier les choix et les valeurs de paramètres qui ont pu être fait à l'aide de différentes expériences. Dans un premier temps, les paramètres liés directement à l'utilisation des atlas locaux vont être étudiés. Ces paramètres interviennent dans la classification des voxels, ainsi que dans le recalage. La deuxième partie a pour but de justifier la modélisation des relations spatiales et leur mode d'utilisation. Enfin une dernière section illustre l'intérêt de l'étape de correction *a posteriori* lors de la segmentation des images médicales.

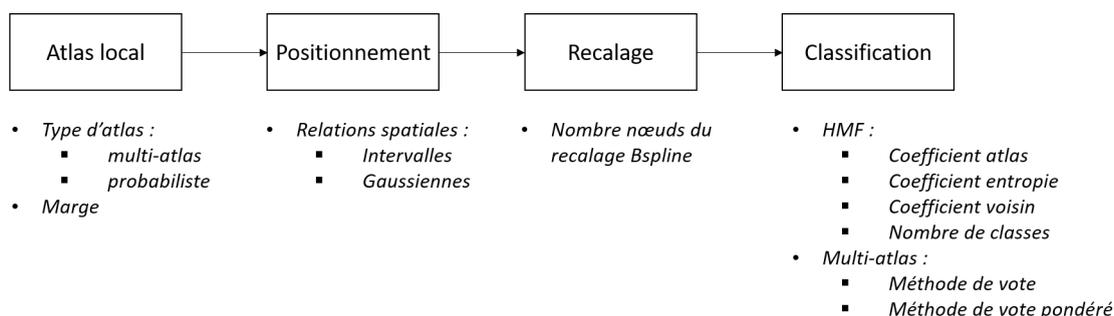


FIGURE 7.6 – Schéma et liste des différents paramètres liés à la méthode de segmentation proposée.

7.2.1 Robustesse de la méthode

Bases utilisées : MICCAI'12

Protocole : Un plan d'expérience ayant pour but de déterminer l'impact des différents paramètres liés à l'utilisation d'une approche locale (atlas locaux et recalage) a été défini. Six paramètres ont été sélectionnés car ils ont probablement une influence plus forte que les autres sur les résultats de segmentation. Le nombre d'itérations du K-means (permettant d'initialiser le HMF) ainsi que de l'algorithme EM du champ de Markov ne sont, par exemple, pas pris en compte dans cette étude et restent donc constamment fixés. De même le recalage peut être appliqué de manière pyramidale. C'est à dire qu'il peut être fait à différents niveau de sous-échantillonnage. Cette probabilité n'a pas été étudié car elle peut améliorer la qualité de la segmentation lorsqu'on l'applique jusqu'au niveau le plus fin mais en contre partie d'un temps de segmentation plus important. Cette expérience ainsi que les suivantes (sauf indication contraire) ont été faites avec un seul niveau et un sous-échantillonnage de 2. Ceci permettant d'avoir des temps de segmentations de l'ordre de la minute pour la plupart des régions. Les 6 paramètres finalement étudiées dans cette expérience sont :

— le coefficient d'importance de l'atlas dans le HMF

- le coefficient lié à l'entropie dans le HMF
- le coefficient lié à l'importance des voisins dans le HMF
- La marge (en pourcentage) prise autour des régions pour la construction et l'utilisation des atlas locaux
- le nombre de classes et de gaussiennes pour le K-means et le HMF
- le nombre de nœuds lors du recalage par B-spline

Pour étudier ces paramètres, un plan d'expérience de type Tagushi [79] a été effectué. Cinq valeurs différentes ont été choisies empiriquement pour chaque paramètre. Ce qui nécessite 25 expériences utilisant différentes combinaisons de paramètres pour obtenir les résultats. Outre ces paramètres, la segmentation a été appliquée sans utiliser les relations spatiales, les bords des boîtes englobantes étant positionnées à l'aide des vérités terrains. L'ordre de segmentation reste également fixé². Dans la suite de ces expériences et lorsque l'ordre de segmentation sera dit fixé, ce même ordre sera toujours réutilisé. La correction *a posteriori* n'a pas été utilisée. La base d'apprentissage a permis de créer le graphe G contenant les différents atlas topologiques nécessaires. Cinq graphes ont été construits, un pour chaque valeur de marge différente. Chaque expérience (combinaison de paramètres) a été évaluée par la moyenne du coefficient Dice (sur les 13 régions) de chacune des 20 images constituant la base de test.

Résultats quantitatifs : La figure 7.7 présente les résultats du plan d'expérience. Ce graphique (des effets principaux) montre les moyennes pour chacun des niveaux de paramètre, du Dice moyen obtenu sur les 20 images de test. Les résultats illustrent tout d'abord le fait que l'absence d'atlas pour guider la segmentation fait énormément diminuer la qualité des segmentations. Il faut donc mettre un coefficient non nul pour obtenir des segmentations correctes. Le coefficient d'importance des voisins ainsi que la marge sont les deux autres paramètres qui ont le plus d'impact. Le coefficient d'importance des voisins ne doit pas avoir une valeur trop élevée (>0.5). Dans le cas où l'on positionne de manière correcte les boîtes englobantes, une marge de 0 % semble aussi être le meilleur choix. Enfin les trois autres coefficients ont une influence plus légère, mais on peut noter que le nombre de classes 3 et 4 ainsi que pour le nombre de nœuds dans le recalage B-spline 5 et 6 semblent plus efficaces.

Bilan : Ces résultats nous ont permis de déterminer les valeurs des paramètres qui ont été utilisées régulièrement lors de ces travaux.

- Les 3 coefficients du champ de Markov utilisés sont de 0.75 pour le coefficient de l'atlas, de 2 pour le coefficient de l'entropie et de 0.05 pour l'importance des voisins. Ces valeurs correspondent aux valeurs fixées dans les travaux initiaux décrits en [84]. Elles ne représentent pas exactement les valeurs maximales obtenues par notre plan

2. Ordre : Noy g, Pal d, Put g, Tha d, Ven d, Noy d, Pal g, Put d, Ven g, Hip d, Hip g, Tro, Tha g

7.2. ROBUSTESSE DE LA MÉTHODE VIS À VIS DE SES PARAMÈTRES

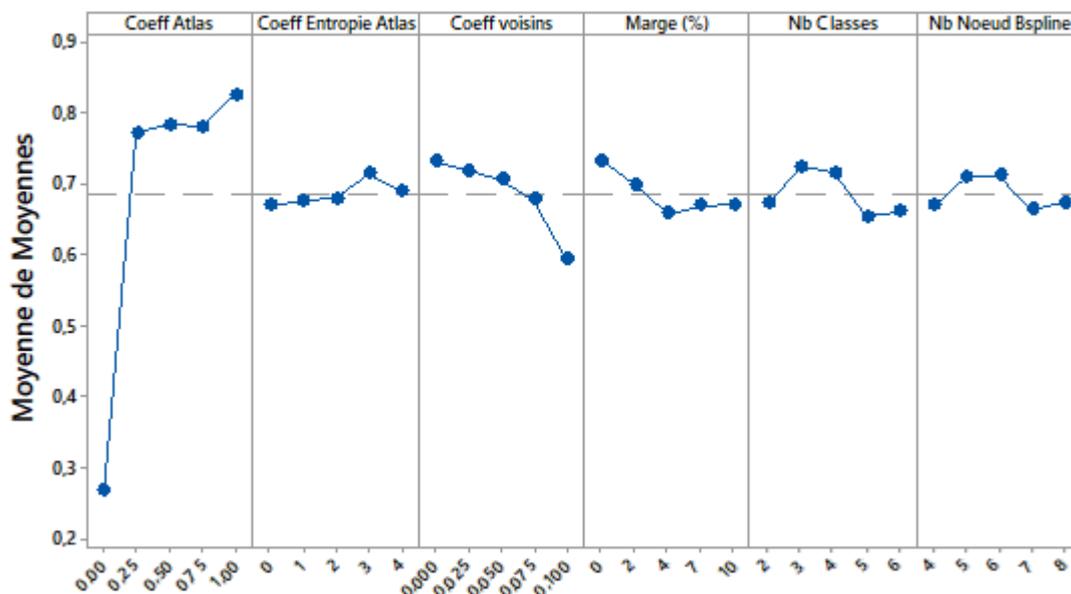


FIGURE 7.7 – Résultat du plan d’expérience de Tagushi pour observer l’impact des paramètres du champ de Markov et du recalage.

d’expérience mais en sont proches. Nous avons fait ce choix afin de pouvoir comparer plus facilement les résultats des modifications successives introduites dans la méthode proposée.

- Le nombre de classes dans le champ de Markov utilisées est de 3. Il semble fournir les meilleurs résultats et l’algorithme EM est plus rapide qu’avec 4 classes.
- Le nombre de nœuds utilisée lors du recalage B-spline qui a été conservé est de 6 car il fournit les meilleurs résultats. Cependant, il est très probable que la valeur optimum de ce paramètre soient dépendant de la taille des régions.
- La marge en pourcentage qui a été conservée n’est pas la marge de 0 %. Cette expérience montre que si l’utilisateur positionne de façon parfaite les bords de la boîte englobante (comme c’est le cas ici) alors il n’est pas nécessaire d’introduire de marge. Cependant, les relations spatiales ainsi que le positionnement de l’utilisateur est rarement parfait. Cette marge permet alors de compenser les erreurs de manière assez efficace. C’est pourquoi la marge finalement conservée est celle de 10 %.

Ces différents paramètres nécessaires pour la mise en place de notre approche locale seront conservés pour l’ensemble des expériences qui suivent. De plus, sauf indications contraires, nous préconisons l’utilisation d’atlas probabilistes locaux (et non pas les multi-atlas locaux) ainsi qu’un champ de Markov comme méthode de classification. Cette méthode sera notée par la suite *LIS* (local incremental segmentation).

7.2.2 Sélection du mode d'exploitation des relations spatiales

Le but de ces expérimentations est de définir le mode d'encodage et d'utilisation des relations spatiales, modélisés à l'aide des arcs du graphe G , le plus adapté à l'utilisation d'information topologique. Les différentes expérimentations décrivent plusieurs manières de définir les informations de distances relatives entre les bords des boîtes englobantes des régions à segmenter (modélisées par les nœuds de G). Plusieurs moyens de fusionner les informations fournis par différents arcs de G pour préparer une segmentation sont aussi évalués.

Bases utilisées : MICCAI'12

Protocole 1 : Pour cette expérience, les relations spatiales sont apprises sur la base d'apprentissage de différentes manières puis elles sont exploitées sur la base de test. Les structures anatomiques présentes dans la vérité-terrain sont utilisées comme positions cibles parfaites. Cette expérience a pour but de déterminer quelle est la définition des distances relatives d'une région à une autre la plus adaptée. Trois types de distance ont été testées.

- (I) la distance relativement à la taille de la région source (cf. eq. 5.11)
- (II) la distance relativement à la taille moyenne des régions sources dans la base d'apprentissage
- (III) la distance relativement à la taille moyenne du cerveau

Ces deux derniers cas ont pour but de vérifier si l'utilisation d'autres références peut rendre plus stable les distances. Deux types d'encodage ont été évalués.

- (1) Les informations sont stockées sous forme de distances relatives minimums et maximums comme définis à l'équation 5.13.
- (2) Les informations sont stockées sous formes de gaussiennes de distances relatives.

Les relations spatiales ont été apprises sur la base d'apprentissage. Pour chacune des images de la base test et chacune des régions r (parmi les 13 structures anatomiques), les 12 autres régions sont considérées comme segmentées (segmentation parfaite). Leur segmentation provient, en fait, directement de la vérité-terrain. Les relations spatiales provenant des 12 régions sont exploitées pour définir le positionnement de la région r . Cette position est comparée à la position parfaite provenant de la vérité-terrain.

Résultats quantitatifs : Le tableau 7.4 illustre les différences de positionnement entre la vérité-terrain et la position proposée à partir des relations spatiales apprises. La métrique utilisée est la distance moyenne en pixel par bord entre les deux positions des bords. Le cas (III) illustre le fait que les relations spatiales comparées à la taille du cerveau semblent

moins précis que les relations spatiales relatives à la taille de la régions sources. Ensuite, on peut noter que l'erreur la plus faible se produit lorsque l'on utilise les distances relatives par rapport à la taille moyenne des régions associées encodées sous forme de gaussiennes des distances.

Pendant la différence est très faible (< 0.02) entre le cas (I) et le cas (II). De plus dans le cas (1) c'est la distance (I) qui fournit les meilleurs résultats alors que c'est la distance (II) qui fournit les meilleurs résultats dans le cas (2). Et la différence entre les distances avec intervalles et avec gaussiennes reste assez faible elle aussi (< 0.09). Ce qui correspond à moins d'un voxel d'écart en moyenne sur toutes les régions et les images de la bases.

Distance conserva- tion	{Min,Max} des dis- tances relatives	{Moy,Var} des distances relatives
I	1.74	1.68
II	1.76	1.67
III	2.68	1.96

TABLE 7.4 – Écart moyen exprimé en pixels entre les bords de la vérité-terrain et le positionnement proposé pour différents types de distances.

Protocole 2 : La deuxième expérience a pour but de comparer les différentes méthodes de fusion d'informations spatiales provenant des différents arcs de G . En plus des méthodes de fusion présentées dans la section 6.3.2, une nouvelle pondération ou sélection des informations est effectuée sur les intervalles afin de vérifier si les distances provenant de structures proches de la structure que l'on recherche donnent des informations plus précises que les structures éloignées. Pour cela 4 méthodes de fusion ont été testées. Toutes les pondérations présentées sont normalisées.

- (I) L'ensemble des informations présentes dans G sont sommées indépendamment de leur distance à la région à segmenter (cas normal).
- (II) Les différentes informations sont sommées en utilisant seulement les 5 régions les plus proches.
- (III) Les différentes informations sont fusionnées en pondérant les informations avec un coefficient $1 - \frac{\text{classement}}{13}$ ($\frac{12}{13}$ pour la structure la plus proche et $\frac{1}{13}$ pour la plus éloignée).
- (IV) Les différentes informations sont sommées en pondérant les informations avec un coefficient égal à $1 - \frac{l_r}{\sum_r l_r}$ avec l_r la distance entre les deux bords des deux régions.

Résultats quantitatifs : Le tableau 7.5 décrit l'erreur moyenne en pixels par bord obtenue avec les différents types de fusion des données. La fusion des données sélectionnant seulement les 5 structures les plus proches fournit des informations moins précises que les autres. Les 3 autres méthodes fournissent des informations de qualité très similaires.

Pondérer les informations en fonction de la distance entre les régions ne semblent pas améliorer les résultats (sans toutefois les diminuer).

Fusion Conservation	(Min,Max) des distances relatives	(Moy,Var)des distances relatives
I	1.74	1.68
II	1.77	1.67
III	1.98	1.80
IV	1.77	1.68

TABLE 7.5 – Écart moyen en pixels entre les bords de la vérité-terrain et le positionnement proposé pour différents types de fusion de données.

Bilan : Ces différents résultats montrent que la qualité du positionnement varie assez peu selon le mode de fusion des informations spatiales stockées dans G . L’encodage des données à l’aide des gaussiennes n’est pas significativement plus performants que l’encodage à base d’intervalle (qui est plus simple à mettre en œuvre). C’est pourquoi, dans le but d’autoriser un apprentissage incrémental simple, l’encodage des informations de distances relatives sous forme d’intervalles a été choisi. En effet, la mise à jour des intervalles de distances relatives lorsque de nouvelles informations deviennent disponibles peut être faite directement. Les gaussiennes ne sont pas forcément simple à faire évoluer sans conserver l’ensemble des informations qui avaient permis de les créer. Il est à noter que si la base d’apprentissage est de taille très importante, les gaussiennes fourniront forcément des informations plus précises que de simples intervalles. Les intervalles de distance ne peuvent que s’agrandir lorsque l’on ajoute l’information provenant de nouvelles images, ce qui finira par réduire leur qualité. Cependant, dans les travaux présentés ici, les bases d’images ont des tailles assez restreintes (compte tenu de la difficulté de production des annotations manuelles).

7.2.3 Impact de l’étape de post-correction

Pour caractériser l’impact de l’étape de post-correction sur la qualité de la segmentation, plusieurs expériences ont été menées. Il s’agissait d’évaluer l’influence des différentes modifications qui ont été apportées à la méthode de correction originale (présentée à la section 6.4) afin de la rendre utilisable dans le cadre d’une approche locale et incrémentale.

Bases utilisées : MICCAI’12

La base d’images de cerveaux humains est utilisée dans le but de montrer l’intérêt de la correction *a posteriori* et d’une segmentation non définitive, les régions recherchées ne sont plus limitées aux 13 structures sous-corticales segmentées précédemment. Les régions supplémentaires sont choisies de manière à maximiser le nombre de régions adjacentes. Dans ce cas, le problème de segmentation définitive devient important. Ainsi, 23 régions sont segmentées (4 structures sous-corticales, la matière blanche et 19 structures corticales).

Protocole : L'expérience suivante permet d'évaluer l'intérêt de l'ajout d'une étape de correction a posteriori après la segmentation initiale par HMF. Différentes corrections sont évaluées ici :

- (*LISn*) une correction initiale proposée par [98] présentée dans la section 4.6

- (*LISp*) une correction utilisant la probabilité *a posteriori* comme caractéristique d'entrée au lieu des labels de classes.

- (*LISpb*) une correction utilisant un classifieur du fond pour chaque région plutôt qu'un unique classifieur de fond.

- (*LISpba*) une correction utilisant une correction de l'adjacence pour combler certains espaces vides entre les régions adjacentes.

Dans ces expériences les bords des boîtes englobantes sont positionnés à partir de la vérité-terrain. Les relations spatiales apprises ne sont donc pas utilisées. Les paramètres du recalage et du champ de Markov sont les mêmes que ceux présentés à la section 7.2.1.

Les 15 images d'apprentissage sont utilisées pour construire le graphe G (atlas locaux et relations spatiales). Cette même base d'apprentissage est segmentée à l'aide du graphe afin d'obtenir les informations nécessaires à l'apprentissage des classifieurs *Adaboost* impliqués dans la post-correction. La base entière est utilisée pour construire le graphe ainsi que pour apprendre les classifieurs. La méthode de correction initiale [98] a aussi été appliquée sur ces images afin de comparer les résultats obtenus. Contrairement à notre correction utilisant les probabilités provenant du champ de Markov, la correction initiale utilise les labels de région correspondant à la segmentation finale. Il est à noter que la segmentation finale est dépendante de l'ordre (segmentation définitive), ce qui entraîne que la correction initiale l'est aussi. C'est pourquoi l'ordre de segmentation des régions est fixé (arbitrairement) et est le même lors de la segmentation que lors de l'apprentissage.

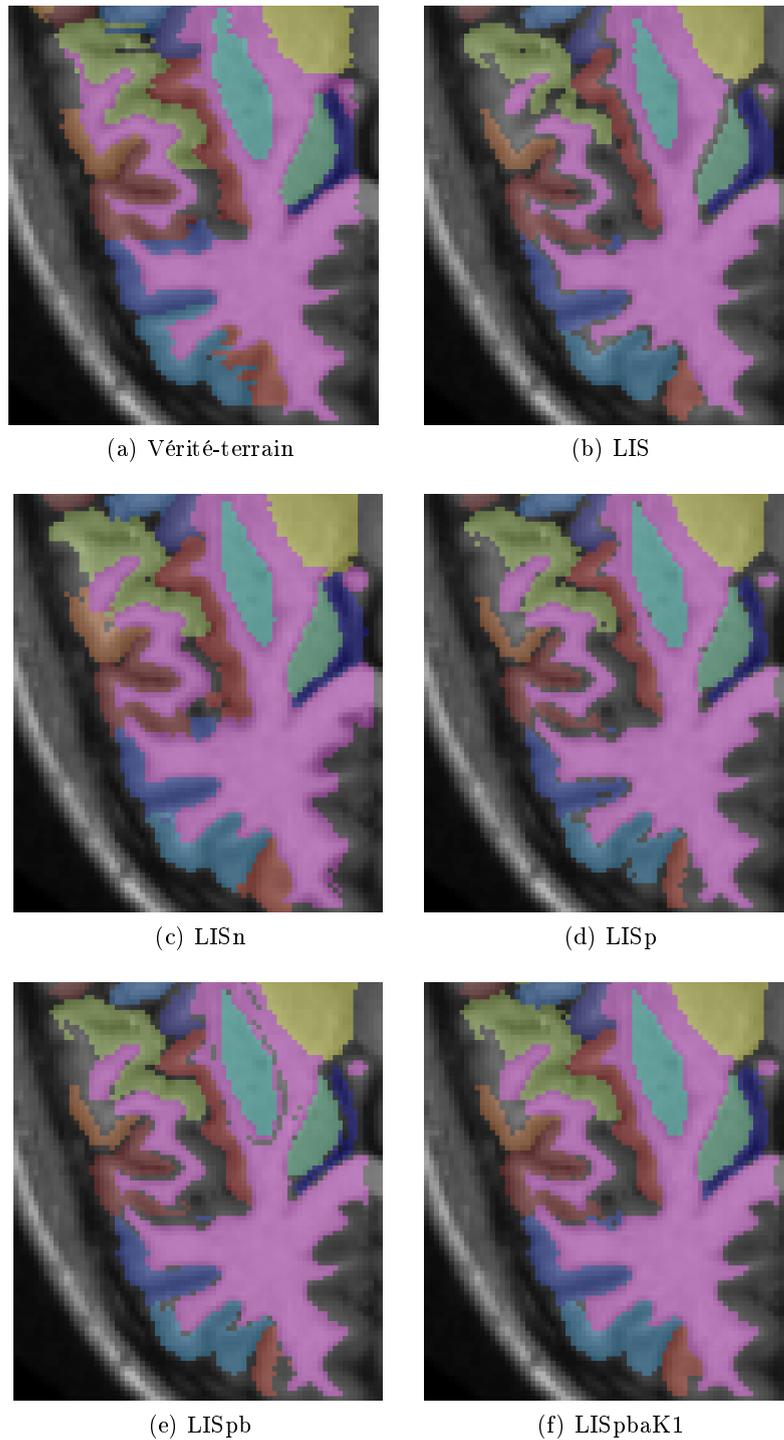


FIGURE 7.8 – Résultat de segmentation de l'image (1003 de la base MICCAI'12). (7.8a) vérité terrain, (7.8b) LIS sans correction, (7.8c) LIS avec correction classique, (7.8c) LIS avec correction utilisant les probabilités du HMF, (7.8d) LIS avec correction utilisant les probabilités du HMF et des classifieurs *non-région*, (7.8f) LIS avec correction utilisant les probabilités du HMF, des classifieurs *non-région* et la correction de l'adjacence.

Résultats qualitatifs : Les figures 7.8c, 7.8d et 7.8e illustrent les segmentations fournies par les différentes versions de la méthode pour une image test. On peut observer qu’il n’y a pas d’espaces vides entre les régions pour la méthode *LISn* et que les régions semblent de taille plus importante que pour la vérité-terrain. Pour les méthodes *LISp* et *LISpb*, les régions semblent plutôt de taille plus réduite que pour la vérité-terrain. Cette caractéristique peut aussi être observée sur la figure 7.9 montrant les résultats des différentes versions de la méthode pour les différents types de régions (sous-corticales, matière blanche et corticale). Pour le cas des régions corticales avec beaucoup d’adjacence entre les régions, la version *LISn* fournit de meilleurs résultats comparée aux deux autres versions. La matière blanche segmentée par la version *LISn* a une taille importante ce qui implique un ratio Dice assez faible. Ceci peut s’expliquer par le fait que lorsqu’un seul classifieur *non région* est appris pour représenter une large diversité de voxels, celui-ci a des difficultés pour être effectivement discriminant. On peut cependant remarquer des voxels non classifiés à l’intérieur de la matière blanche (fig. 7.8e), ce problème n’étant pas présent dans le cas de la correction classique *LISn*. Même si *LISpb* améliore les résultats de la segmentation classique *LIS*, il existe des espaces vides (non labellisés) entre les différentes régions corticales ainsi que l’intérieur de la matière blanche. Ces espaces ne sont pas présents dans la vérité-terrain et pour la méthode avec correction normale *LISn*. Ces espaces sont beaucoup moins présents avec la correction *LISpbaK1* sans pour autant avoir modifié la partie extérieure de la segmentation (extérieur du cortex). La correction de l’adjacence permet donc de corriger en partie les problèmes dus à l’incrémentalité de la méthode *LIS*.

Méthode	LIS	LISn	LISp	LISpb	LISpbaK2	LISpbaK1
Dice moyen	0.660	0.728	0.732	0.734	0.749	0.750

TABLE 7.6 – Indices Dice moyens de la segmentation avec les différentes méthodes de corrections appliquées.

Résultats quantitatifs : Les indices Dice moyens sur les 23 régions et les 20 images sont fournis dans le tableau 7.6. La qualité de la segmentation est fortement améliorée de 6.8% lorsque la correction *LISn* est appliquée à la méthode classique *LIS*. L’utilisation des informations probabilistes *LISp* en entrée des classifieurs Adaboost améliore encore le ratio Dice même lorsqu’un seul classifieur du fond est utilisé. L’ajout des classifieurs de fond pour chaque région améliore aussi les résultats mais la différence est moins significative (0.2%). On peut cependant noter que ce procédé permet de rendre la méthode entièrement incrémentale et fournit de meilleurs résultats que la méthode non incrémentale. Nous en concluons qu’il n’est plus nécessaire d’attendre que toutes les régions soient segmentées ; il est possible d’appliquer la correction après la segmentation de chaque région.

Le ratio Dice et la qualité générale de la segmentation sont améliorées avec la version *LISpb*. Cependant, les espaces vides entre les régions ne sont pas naturels (cf. figure 7.8e). Ceci explique la petite diminution du ratio Dice lié aux régions corticales sur la figure 7.9. L’utilisation de la correction de l’adjacence appliqué *LISpba* permet de corriger ce problème. Le ratio Dice des régions corticales est augmenté jusqu’à 0.739 (+1.7 % comparé

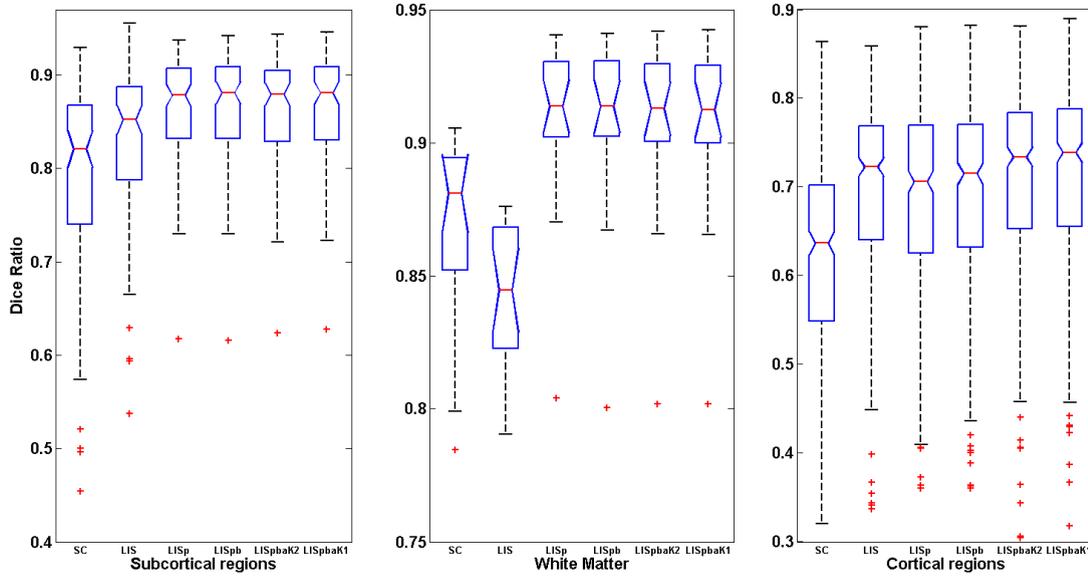


FIGURE 7.9 – Distribution du ratio Dice pour chaque méthode pour les régions sous-corticales, la matière blanche et les régions corticales

à la version normal $LISn$ et $+2.4\%$ comparé à la version avec probabilité et classifieur non-région $LISpb$). La qualité de la segmentation pour les régions sous-corticales et pour la matière blanche restent presque identiques. Ceci se traduit aussi sur le Dice moyen des régions sur le tableau 7.6. Le Dice moyen est amélioré avec les deux coefficients d'adjacence $K1$ et $K2$ (cf. section 6.4) de 0.734 à 0.750 et 0.749. La correction des problèmes d'adjacence des régions permet donc de fournir des segmentations de meilleure qualité pour les régions complexes où la surface en contact avec les autres régions est importante. Elle limite les problèmes liés au fait que la méthode de segmentation LIS soit incrémentale et que les segmentations locales soient indépendantes.

Bilan : La méthode de post-correction incrémentale proposée dans ces travaux permet d'améliorer la qualité de la segmentation produite par la méthode LIS seule. L'utilisation des probabilités ainsi que des classifieurs non-région indépendants permet de rendre la correction incrémentale, tout en conservant une correction de qualité similaire voir légèrement supérieure à la correction initiale [98]. L'ajout d'une prise en compte de l'adjacence entre régions lors de la fusion des décisions locales prises par les classifieurs Adaboost permet de minimiser l'impact du caractère incrémental de notre méthode, en particulier dans les endroits complexes d'une image où beaucoup de régions s'entremêlent.

7.3 Sensibilité et généralité de la méthode proposée

Dans cette section, nous mettons en évidence l'impact des propriétés que nous considérons comme fondamentales de notre méthode. Il s'agit de démontrer que cette méthode

n'est pas trop dépendante ou sensible vis à vis des disparités qu'il peut y avoir entre les images à segmenter et les images utilisées pour l'apprentissage du graphe structurel. L'intérêt d'une approche local plutôt que global sera également mis en valeur. Dans un premier temps, l'impact de la base d'apprentissage sur les résultats de segmentation sera mis en évidence. Ensuite l'impact de l'ordre de segmentation des régions sera étudié. Enfin, nous décrirons l'influence que peut avoir le positionnement manuel sur les résultats de segmentation.

7.3.1 Influence de la base d'apprentissage

7.3.1.1 Sensibilité vis à vis de la base d'apprentissage utilisée

Base utilisée : MICCAI'12

Protocole : L'objectif de cette expérience est d'évaluer l'impact du nombre d'images utilisées pour construire le graphe sur les résultats de segmentation. La base d'apprentissage est composée de 15 images au total. Quinze graphes différents sont construits à partir de ces images. Le premier graphe est construit à partir d'une seule image, le deuxième graphe est construit à partir des deux premières images, ainsi de suite jusqu'au quinzième graphe qui est appris à partir de l'ensemble des images de la base d'apprentissage. Chaque graphe est utilisé pour segmenter les images de la base de test. Deux types de segmentation sont produites avec chaque graphe. La première (E1) est une segmentation où les relations spatiales ne sont pas utilisées. Le positionnement des boîtes englobantes est déterminé à l'aide de la vérité-terrain. La deuxième expérience (E2) utilise les relations spatiales. Seule la première région est positionnée à l'aide de la vérité-terrain. Le positionnement des régions suivantes est effectué à partir des relations spatiales provenant des graphes appris.

Résultats quantitatifs Les figures 7.10 et 7.11 décrivent la qualité de segmentation pour les différents graphes et pour les deux expériences. La qualité de la segmentation est ici aussi décrite par le ratio Dice. Les deux différentes courbes illustrent l'évolution de la qualité en fonction du nombre d'images disponibles pour l'apprentissage. Chaque *diagramme en boîte* représente la statistique sur les 20 images de test disponibles.

Dans le cas de l'expérience (E1), la qualité de la segmentation augmente avec le nombre d'image dans la base d'apprentissage jusqu'à la 4^{ème} image puis se stabilise. La moyenne reste stable tandis que la médiane décroît légèrement lorsque le nombre d'images d'apprentissage augmente et tend vers 15. Dans le cas de l'expérience (E2) les résultats sont plus instables. On peut noter que la qualité de la segmentation est supérieur lorsque le nombre d'images disponibles est supérieur à 10. Mais les résultats sont assez fluctuant et peuvent être de moins bonne qualité avec l'ajout d'une nouvelle image.

Bilan : La première expérience illustre le fait que les atlas probabilistes locaux ne nécessitent que peu d'images pour être efficace. Dans la situation de la base d'image MICCAI'12,

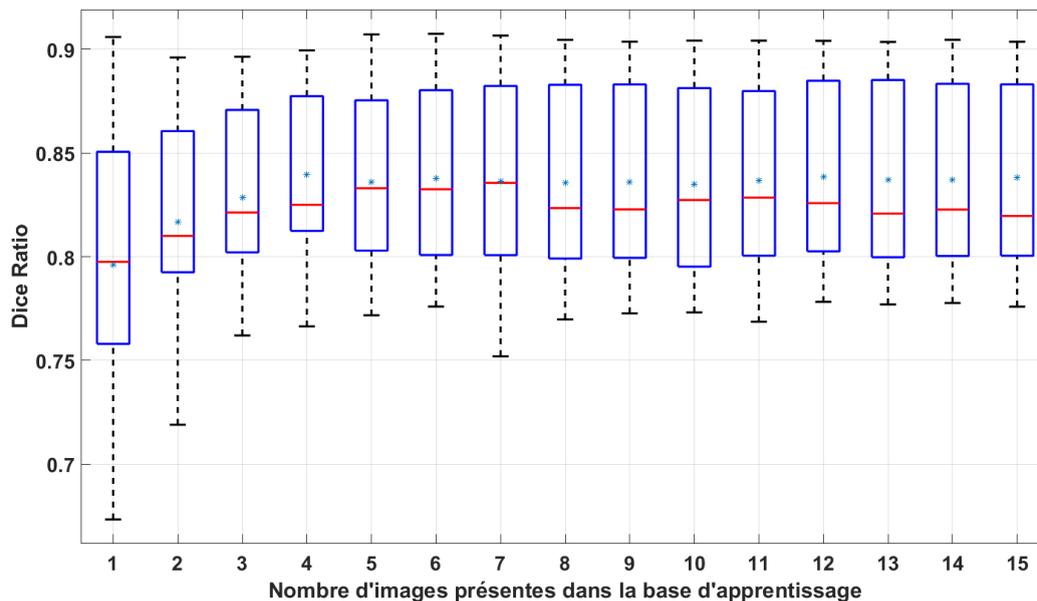


FIGURE 7.10 – Évolution de la qualité de la segmentation en fonction du nombre d'images présentes dans la base d'apprentissage. La moyenne est représentée par l'étoile bleue.

4 images suffisent à atteindre un plancher en terme de qualité de segmentation. Lors de l'utilisation des relations spatiales, les résultats moins nets. Le caractère plus variable du succès ou non des relations spatiales rend moins visible l'impact du nombre d'images dans la base d'apprentissage. Un nombre plus important de graphes différents pourrait être construit afin d'augmenter la statistique.

7.3.1.2 Apprentissage inter modalités

Le but de cet expérimentation est de montrer qu'il est possible de segmenter des images de modalité différentes de celles utilisées pour construire le graphe des connaissances *a priori*. Il s'agit ainsi d'une première démonstration de la généralité de l'approche proposée.

Bases utilisées : NeuroGeoEx

Dans le cadre du projet NeuroGeo et du développement d'outils de segmentation pour les images IRM de brebis, les biologistes utilisent plusieurs types d'images. En plus de la base d'images haute résolution *NeuroGeoEx*, des images *in vivo* T1 ont été acquises. Cette base d'images provient de la plateforme CIRE de l'INRA de Nouzilly. Contrairement à la base NeuroGeoEx, la tête de la brebis est visible. L'intensité des voxels n'est pas nulle à l'extérieur du cerveau. La résolution de l'image est de $0.5 \times 0.5 \times 0.5$ mm par voxel. La figure 7.12 illustre un exemple de cette base d'images. L'ensemble des régions anatomiques

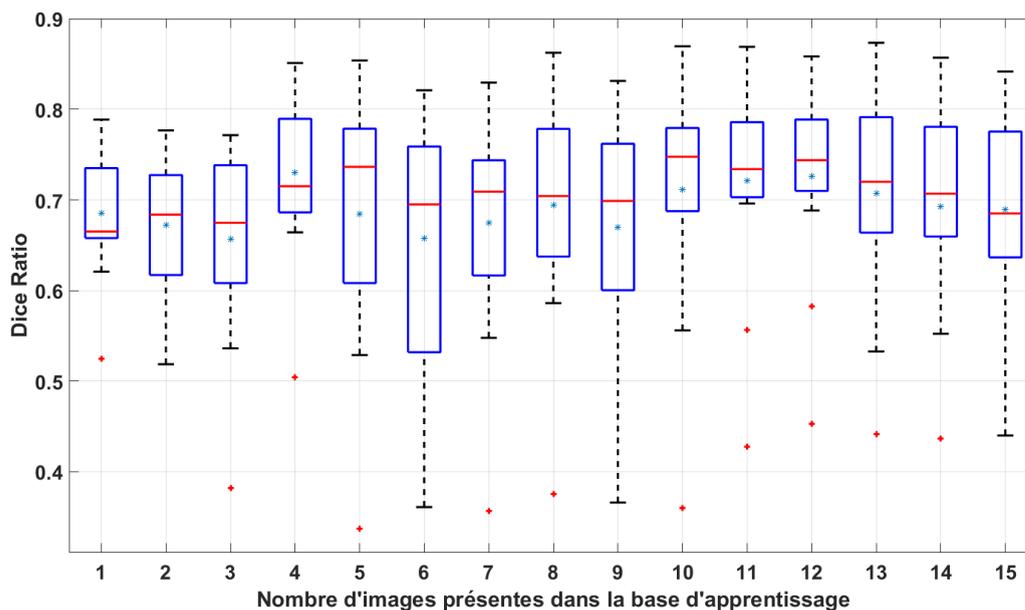


FIGURE 7.11 – Évolution de la qualité de la segmentation en fonction du nombre d'images de la base d'apprentissage. La moyenne est représentée par l'étoile bleue.

décrit dans la section 7.1.1.2 ne sont pas forcément visibles sur ce type d'images (à cause de la résolution et de la modalité).

Protocoles : Dans cette expérience les images *in vivo* sont segmentées en utilisant un graphe de connaissances *a priori* provenant de la base NeuroGeoEx. En plus de ne pas être acquises avec la même modalité, ces images ne proviennent pas du même contexte, dans le sens où les bases possèdent des conditions d'acquisitions très différentes (*ex vivo* et *in vivo*).

Six images d'apprentissage (de type T2 *ex vivo*) sont exploitées pour créer le graphe des connaissances *a priori*. Les atlas locaux et les relations spatiales sont appris durant cette phase. Quatre régions ont été utilisées durant cet apprentissage. Les images à segmenter sont constituées de 4 images de cerveau (de type T1) *in vivo*. Durant une première expérience, les images sont segmentées avec notre méthode guidée par un expert. Le positionnement des boîtes englobantes de chaque région recherchée a donc pu être corrigé si nécessaire. Dans la deuxième expérience seule une région a été bien positionnée (Noyau caudé) puis les autres ont été positionnées automatiquement à l'aide des relations spatiales préalablement apprises.

Résultats qualitatifs : Les données segmentées (vérité-terrain) étant difficile à obtenir car demandant un temps considérable aux biologistes, les résultats présentés ici sont seulement des résultats qualitatifs sur les segmentations produites.

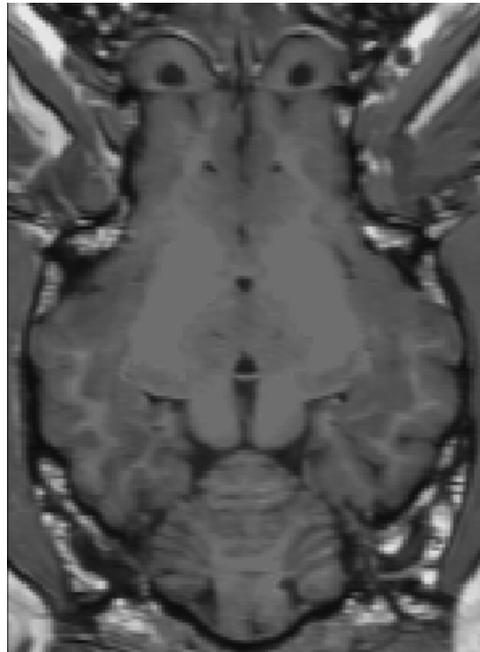


FIGURE 7.12 – Coupe transversal d'une image IRM T1 d'un cerveau de brebis in vivo (acquisition sur la plateforme CIRE).

La figure 7.13 présente la segmentation d'une image produite par notre outil de segmentation. Les Noyaux caudé ainsi que les hippocampes ont pu être segmentés. On peut remarquer que la segmentation de ces régions semble correcte. La figure 7.13 montre la même segmentation lorsque les relations spatiales sont utilisées. Seul le Noyaux caudé gauche (bleu) a été positionné manuellement. On peut remarquer que la segmentation du Noyau caudé gauche et de l'hippocampe droite restent de bonne qualité. Par contre, le deuxième hippocampe est cependant est de taille trop importante.

Bilan : Cette expérience démontre que la segmentation entre différentes modalités est possible. En effet, l'apprentissage a été fait sur des cerveaux extraits alors que la segmentation a été appliquée sur des cerveaux où la boîte crânienne est toujours présente. Enfin on peut remarquer que les relations spatiales peuvent être aussi efficaces lorsque les images sont différentes. Les relations spatiales relatives permettent de gérer les différences de résolution entre les images.

7.3.2 Sensibilité vis à vis du séquençage et des positionnements

Le positionnement des boîtes englobantes joue un rôle important sur les résultats de segmentation. Dans cette section, nous étudions les différentes caractéristiques qui influent sur le résultat du positionnement des atlas locaux et leur influence sur la segmentation finalement obtenue. Nous décrirons, dans un premier temps, l'impact que l'ordre de segmentation des régions peut avoir. Puis, nous analyserons le lien existant entre le mode de

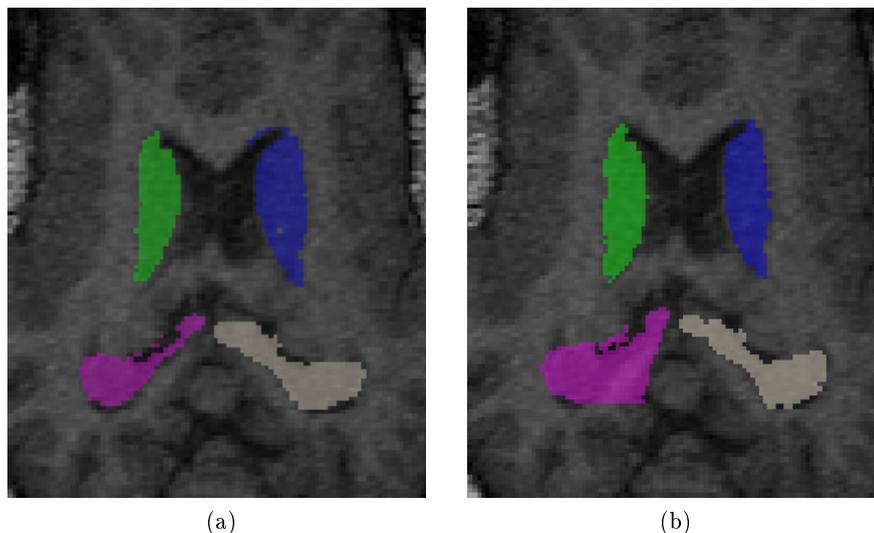


FIGURE 7.13 – Segmentation guidée par un utilisateur (7.13a) et guidée par les relations spatiales (7.13b) d’une image T1 de brebis in vivo

positionnement et la qualité de la segmentation.

7.3.2.1 Influence de l’ordre de segmentation

L’ordre de segmentation joue un rôle qui peut devenir important lorsque les relations spatiales sont utilisées pour le positionnement des atlas locaux ; et que l’utilisateur ne valide pas ou ne corrige pas les positionnements. En effet, positionnements dépendent de la qualité des segmentations précédentes. Elles nécessitent d’avoir l’information de taille et de position des bords ces régions. Si les premières régions sont mal segmentées, les relations spatiales provenant de ces régions peuvent être de mauvaise qualité et faire diverger la suite de la segmentation.

Bases utilisées : MICCAI’12

Protocole : La base d’apprentissage a permis de créer un graphe qui va être utilisé pour l’ensemble des segmentations. Durant cette expérience, l’étape de correction *a posteriori* n’est pas activée. Une seule région (la première) est positionnée à l’aide de la vérité-terrain. Ensuite, les relations spatiales sont utilisées pour positionner les 12 régions restantes. Le nombre d’ordres potentiels étant très important, 13 ordres aléatoires ont été choisis, en plus de l’ordre automatique défini tel que décrit dans la section 6.3.3. Treize segmentations différentes ont donc été appliquées. Dans chacune de ces segmentations, une région différente a été positionnée correctement en premier. Ensuite, l’ordre a été déterminé automatiquement selon le critère décrit précédemment (cf section 6.3.3).

En plus des ordres aléatoires et des ordres déterminés automatiquement par le système, des ordonnancements manuels définis lors d'un atelier utilisateurs ont aussi été étudiés. Lors de cette expérience les utilisateurs devaient segmenter un cerveau humain provenant de la base MICCAI'12. Le positionnement des boîtes englobantes étaient effectué à l'aide de la vérité-terrain et seul l'ordre de segmentation était choisi par les utilisateurs. Les ordres choisis par les différents utilisateurs ont ensuite pu être appliqués sur l'ensemble des images tests, les boîtes englobantes étant positionnées automatiquement à partir des connaissances stockées dans le graphe.

Résultats quantitatifs : Le tableau 7.7 décrit les différents ordres calculés automatiquement obtenus pour les 13 différentes possibilités de première région. Le nombre indiqué correspond au rang moyen dans la séquence parmi les 13 régions (moyenne sur les 20 images de la base de test). Ce tableau comprend des *patterns* d'ordres intéressants. La position de la région et son caractère central dans le cerveau, ainsi que son hémisphère sont les deux principaux critères expliquant les ordres obtenus. En effet, lorsque la première région segmentée est une structure appartenant à un hémisphère alors les prochaines régions choisies par l'ordre automatique ont tendance à être situées dans le même hémisphère. Par exemple, le thalamus, le noyau caudé, le putamen et le pallidum (qui sont les Noyaux gris centraux) se trouvent dans le même hémisphère que la première région et ont généralement des valeurs inférieures à 5. De même, on peut remarquer que les Ventricules, les hippocampes ainsi que le tronc cérébral ont des valeurs élevées et sont segmentés en dernier. Donc, de manière générale, l'ordre de segmentation se déroule de la manière suivante. La première région est segmentée, puis l'un des Noyaux gris centraux du même hémisphère est segmenté. Ensuite, l'ensemble des Noyaux gris centraux de cet hémisphère est segmenté. Puis, l'ensemble des Noyaux gris centraux de l'autre hémisphère est segmenté. Enfin les régions externes et plus variables sont extraites. En général, les hippocampes sont les deux dernières régions à être segmentées. Ces différentes observations expriment le fait que les distances relatives à la taille de la région source semblent être plus stables entre les régions d'un même hémisphère ainsi qu'entre les structures sous-corticales centrales.

La figure 7.14 montre les résultats de segmentations pour différents ordres. Pour chacune de ces expériences, seule la première région a été positionnée de manière correcte à l'aide des vérités-terrain. Ensuite, les relations spatiales ont été utilisées pour positionner les différentes régions. Seul l'ordre de segmentation est donc modifié. Une première série d'expériences a été effectuée avec un ordre aléatoire. Une deuxième série avec l'ordre automatique présenté dans la section 6.3.3 (13 différentes avec une région différente comme 1^{ère} région). Enfin, les ordres choisis par des utilisateurs lors d'un exercice qui consistait à segmenter des images de cerveaux humains ont aussi été testés. La figure 7.14 illustre pour chacun de ces types d'ordre, la qualité de la meilleure segmentation, du cas médian et de la plus mauvaise segmentation. Les diagramme en boîte décrivent la statistique du ratio Dice obtenue sur les 20 images de la base de test.

Les résultats provenant des ordres *utilisateurs* et aléatoires sont assez similaires. L'écart entre le meilleur ordre et le moins bon est assez important. La qualité des segmentations provenant de l'ordre utilisateur peut rester assez faible en raison du choix des premières régions à segmenter. Plusieurs utilisateurs ont ainsi choisi de commencer la segmentation du

7.3. SENSIBILITÉ ET GÉNÉRICITÉ DE LA MÉTHODE PROPOSÉE

TABLE 7.7 – Rang moyen des régions dans l’ordre de segmentation. Les colonnes représentent quelle région a été positionnée et segmentée en premier. Les lignes représentent à quelle position cette région a été segmentée en moyenne sur les 20 images de test.

region	Noy ^g	Pal ^d	Put ^g	Tha ^d	Ven ^d	Noy ^d	Pal ^g	Put ^d	Ven ^g	Hip ^d	Hip ^g	Tro	Tha ^g	total
Noy ^g	1	8	5.3	7.8	9.3	8	4.6	8	7	9	8.5	9	7.5	7.1
Pal ^d	5.1	1	4.5	3.6	3.3	2.3	5.0	2	5.7	2.1	5.1	4.3	3.6	3.6
Put ^g	2.1	7	1	5.3	7.7	7	2	7	2.55	7.7	5.6	6.0	5.1	5.1
Tha ^d	5.6	3.9	5.2	1	4.8	3.9	5.1	3.9	5.6	4.5	3.5	3.3	2	4.0
Ven ^d	10.4	10.6	10.6	10.5	1	10.5	10.6	10.5	10.5	11.7	11.8	11	10.8	10.0
Noy ^d	7.3	3.3	6.8	5.9	4.3	1	7.2	3.3	7.6	5.5	7.3	7.1	6.3	5.6
Pal ^g	2.9	6	2	4.3	6.7	6	1	6	2.9	6.7	4.7	4.9	4.1	4.5
Put ^d	7.3	2	6.9	6.3	5.1	2.9	7.2	1	8.1	3.2	7.5	6.9	6.5	5.4
Ven ^g	9.1	9.1	9.1	9.1	8	9.1	9.1	9	1	10.3	10.5	10	9.1	8.6
Hip ^d	12.0	12.0	12.0	12.1	11.7	12.0	12.0	11.9	11.9	1	11.7	12	12	11.1
Hip ^g	12.9	12.9	12.9	13.0	12.8	12.8	12.9	13.0	12.9	13	1	13	13	12.0
Tro	10.7	10.6	10.7	10.5	11.6	10.7	10.5	10.7	10.8	11	11.3	1	10.2	10.0
Tha ^g	4.8	4.9	4.4	2	5.1	4.9	4.2	4.9	4.6	5.4	2.7	2.6	1	3.9

cerveau à partir des Ventricules ou du tronc cérébral. Ces régions ont des tailles variables ou sont plus éloignées des autres régions. Les informations spatiales qui proviennent de ces régions peuvent alors être imprécises.

Cependant on peut remarquer que les résultats produits par l’ordre automatique semblent de meilleures qualités. L’écart entre la qualité de la segmentation dans le meilleur cas et le moins bon est plus faible qu’avec les autres ordres de segmentation. L’ordre le moins bon fournit tout de même des segmentations qui paraissent correctes et proches de la qualité des meilleurs cas utilisateurs et aléatoires.

Bilan : Ces deux expériences ont permis de mettre en évidence l’influence et les propriétés de l’ordre automatique d’extraction des régions. En choisissant, de manière automatique, la prochaine région à segmenter en fonction de la cohérence des relations spatiales, on augmente les chances d’obtenir des segmentations correctes qui engendreront à leurs tours des positionnements corrects. Certains *patterns* de segmentation semblent aussi plus efficaces que d’autres. Commencer par des régions simples avec des formes et des tailles stables fournies de meilleures informations pour les régions suivantes. De plus ils semblent aussi préférable que les régions soient internes et proches des régions qui vont suivre. Ces conclusions ne concernent néanmoins que le cas où les seules relations spatiales sont utilisées. Lors de l’utilisation normale du logiciel construit, l’utilisateur pourra corriger le positionnement. L’ordre pourra alors devenir moins décisif. De plus, on peut noter que plusieurs ordres différents peuvent fournir les mêmes types de résultats.

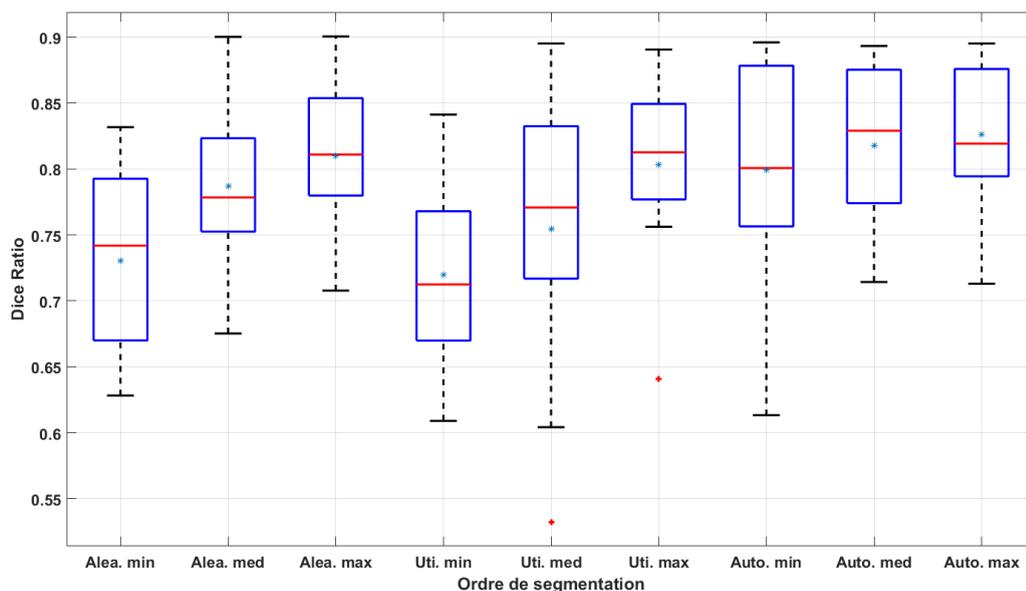


FIGURE 7.14 – Qualité des segmentations en fonction de différents types d'ordre. Les relations spatiales sont utilisées et seule la première région est bien positionnée. Pour chaque type d'ordre, la plus mauvaise segmentation, le cas médian et la meilleure sont exposés.

7.3.2.2 Étude du nombre de régions nécessaires au positionnement automatique

La correction interactive du positionnement des bords des boîtes englobantes par l'utilisateur assure d'obtenir des segmentations et des informations fiables qui serviront lors des segmentations suivantes. Plus le nombre de régions bien segmentées est important, plus les positionnements suivants seront potentiellement précis.

Cependant, le nombre de structures nécessaires pour que les informations spatiales apprises puissent guider correctement les segmentations suivantes doit rester faible afin que l'utilisateur n'ait pas besoin de corriger un nombre trop important de positionnements automatiques. Cette expérimentation vise donc à étudier la relation pouvant exister entre le nombre de structures bien positionnées et la qualité de la segmentation globale obtenue.

Bases utilisées : MICCAI'12

Protocole : Un graphe est appris sur les 15 images de la base d'apprentissage. Les atlas probabilistes locaux sont utilisés lors de la segmentation. La correction *a posteriori* n'est pas activée. L'ordre de segmentation des régions est fixe. Les 13 régions habituelles sont segmentées ici. Le nombre n de régions bien positionnées varie entre 1 et 13. Les régions bien positionnées sont placées à l'aide de la vérité-terrain. Les autres régions sont positionnées à l'aide des relations spatiales apprises. Si n est égale à 1, alors seule la première région sera positionnée avec la vérité-terrain et toutes les suivantes seront positionnées à l'aide des

relations spatiales apprises. Si n est égale à 13 alors toutes les régions sont bien positionnées et les relations spatiales ne sont pas utilisées. Pour chaque valeur de n , les 20 images de la base de test sont segmentées.

Résultats Quantitatifs : La figure 7.15 présente l'évolution du ratio Dice moyen de la segmentation en fonction du nombre de régions positionnées à l'aide de la vérité-terrain. La qualité de la segmentation atteint très rapidement la valeur maximale obtenue lorsque toutes les régions sont placées correctement à l'aide de la vérité-terrain. À partir de deux régions bien positionnées le ratio Dice moyen est seulement de 2.7 % en dessous du maximum. Celui-ci augmente ensuite lentement jusqu'à la 7^{ème} régions.

La figure 7.16 illustre la qualité de la segmentation pour chaque région. L'ordre des régions sur la figure correspond à l'ordre de segmentation effectuée. La courbe *bleu clair* correspond à la segmentation pour laquelle une seule région est placée avec la vérité-terrain. La courbe *vers foncé* correspond à la segmentation où toutes les régions sont placées avec la vérité-terrain. On peut voir que les courbes convergent rapidement vers la courbe où toutes les régions sont bien placées. À partir de la deuxième région les courbes deviennent similaires. Seuls le premier thalamus ainsi que le deuxième noyau caudé ont un écart non négligeable entre la segmentation avec le positionnement automatique et le positionnement avec la vérité-terrain. La convergence des courbes indique que les relations spatiales permettent de placer les boites englobantes avec une qualité suffisante.

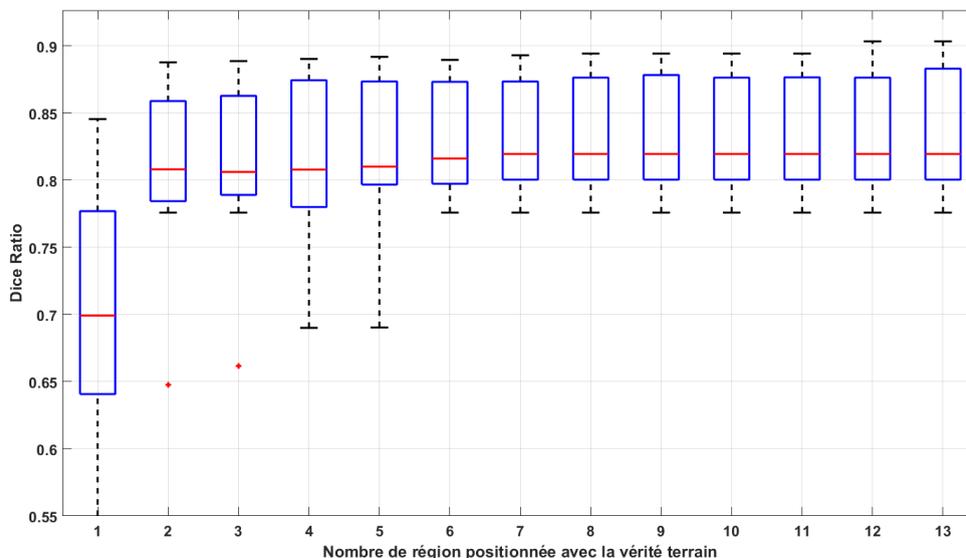


FIGURE 7.15 – Qualité de la segmentation en fonction du nombre de région positionnée correctement (vérité-terrain), les autres régions étant placées automatiquement (relations spatiales).

Bilan Malgré le fait que cette propriété soit probablement dépendante de l'ordre de la segmentation ainsi que du type des images à segmenter. Cette expérience met en évidence

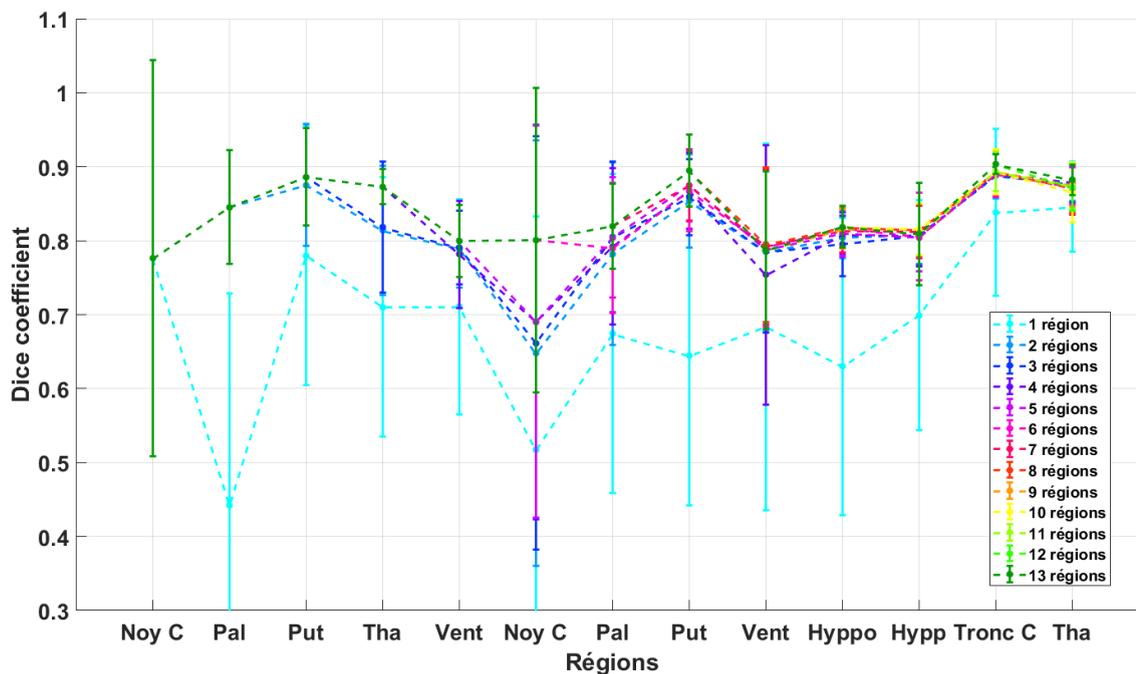


FIGURE 7.16 – Qualité de la segmentation par région en fonction du nombre de régions correctement positionnées (vérité-terrain), les autres régions sont placées automatiquement (relations spatiales). Les régions en abscisse sont ordonnées selon l'ordre de segmentation appliqué.

qu'un nombre de régions bien placées peu important (2 ou 3) peut suffire à bien segmenter les régions sous-corticales du cerveau lorsque l'ordre de segmentation est bien choisi.

7.3.2.3 Comparaison entre positionnement manuel et automatique

Un atelier utilisateur (présenté section 8.2) a permis d'obtenir des données d'interaction concernant les modifications de positionnements de boîtes englobantes faites sur les positionnements automatiques par un panel d'utilisateurs. Cette étude vise à analyser ces données pour comparer le positionnement automatique au positionnement manuel.

Bases utilisées : MICCAI'12 / NeuroGeoEx

Protocoles : Lorsque l'outil de segmentation est utilisé de manière libre, les utilisateurs peuvent corriger des positionnements automatiques proposés. Dans le cas de la base MICCAI'12, les informations *a priori* ont été construites sur la base d'apprentissage et appliquées sur certaines images de la base de test. Dans le cas de la base d'images de cerveaux de brebis, les informations ont été apprises sur 5 images afin de segmenter la 6^{ème} (*leave-one-out*). La correction *a posteriori* n'est pas activée.

Résultats Quantitatifs : Les figures 7.17 et 7.18 mettent en évidence l'erreur de positionnement manuel comparée à l'erreur de positionnement automatique. L'erreur indiquée est celle mesurée après le repositionnement de la boîte englobante par l'utilisateur. Il s'agit de l'erreur relative, c'est à dire l'erreur en pixels comparée à la taille de la région à segmenter.

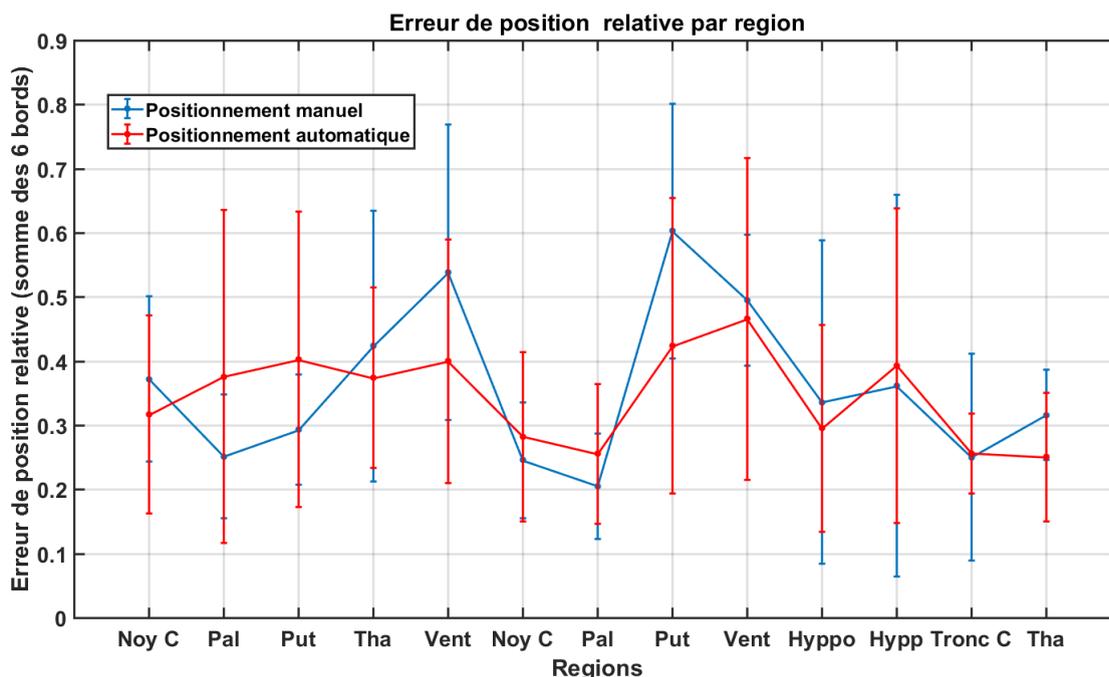


FIGURE 7.17 – Erreur de positionnement en fonction des régions du cerveau humain, positionnement manuel (rouge) et automatique (bleu).

Les résultats sont fortement dépendants de la région étudiée. Certaines structures sont mieux positionnées par les utilisateurs tandis que d'autres sont mieux positionnées automatiquement. Dans le cas du cerveau humain, les Noyaux caudés et les Ventricules sont mieux positionnés par les utilisateurs que par le système automatique. Ces structures sont en fait bien visible avec un contraste assez élevée. Même les utilisateurs qui ne sont pas experts du cerveau peuvent détecter facilement les limites de ces régions. À l'inverse, l'hippocampe est une structure connue du cerveau mais avec une forme assez complexe. Les bords de cette structure ne sont pas forcément facile à positionner. Dans ce cas, le positionnement manuel est en fait de moins bonne qualité que le positionnement automatique proposé par la plupart des utilisateurs (même experts).

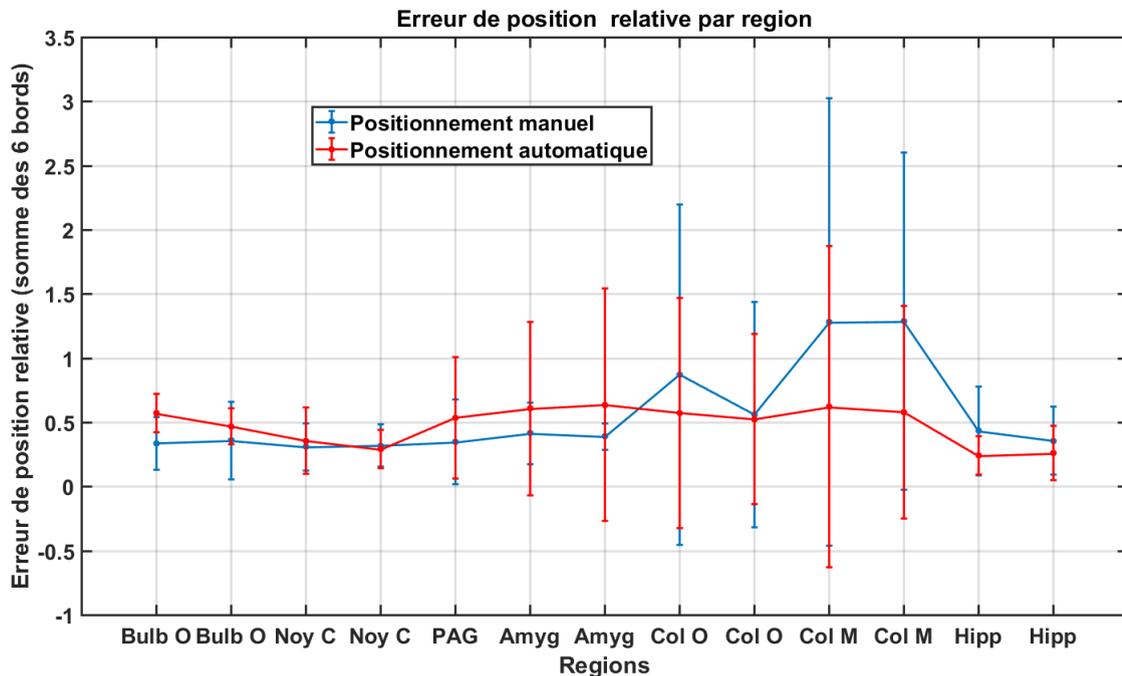


FIGURE 7.18 – Erreur de positionnement en fonction des régions du cerveau de brebis, positionnement manuel (rouge) et automatique (bleu).

Les mêmes conclusions sont illustrées sur la figure 7.18 pour les images de cerveaux de brebis. Les bulbes olfactifs, les noyaux caudés et la PAG sont, par exemple, des structures assez discernables. Le positionnement manuel est alors meilleur que le positionnement automatique. Les colliculi sont par contre des structures de petite taille avec un contraste très faible. Le positionnement de l'utilisateur est très souvent de moins bonne qualité que le positionnement automatique proposé.

Bilan : Ces différents résultats mettent en évidence la qualité du positionnement automatique. Celui-ci est de qualité similaire au positionnement manuel fourni par l'utilisateur. Ces relations spatiales sont même de meilleures qualités dans le cas des régions plus difficilement discernables. Bien évidemment, ces résultats sont dépendants de la pertinence (ressemblance) des données apprises et stockées dans le graphe par rapport aux images à segmenter. Si la variabilité des images à segmenter est trop grande (présence de pathogène, utilisation d'un graphe ex vivo pour segmenter des images in vivo, etc.) alors le positionnement peut significativement être amélioré grâce aux interactions d'un véritable expert.

7.3.2.4 Impact du positionnement sur les résultats de segmentation locale

Pendant l'atelier utilisateurs (cf. section 8.2), un grand nombre de segmentations de régions ont pu être effectuées. À chaque fois, l'utilisateur a validé un positionnement qui

pouvait avoir été effectué manuellement ou automatiquement. Cependant ce positionnement n'est pas forcément un placement parfait. Dans cette section, nous étudions l'impact que peut avoir un positionnement imparfait (validé par un utilisateur) sur les résultats de segmentation.

Bases utilisées : NeuroGeoEx

Protocole : Les données utilisées proviennent de la segmentation des images de brebis *ex vivo*. La correction *a posteriori* n'est pas activée. La segmentation des images s'est déroulée de manière libre, l'utilisateur choisissant l'ordre de segmentation ainsi que le positionnement des régions. Lors de cette expérience, le positionnement des régions a été sauvegardé. Chaque positionnement a été visualisé par l'utilisateur et accepté comme étant correct. La qualité de la segmentation de chaque région en fonction de l'erreur de positionnement peut donc être évaluée.

Résultats quantitatifs : La figure 7.19 décrit le résultat de segmentation des différentes structures anatomiques en fonction de la qualité du positionnement. La qualité du positionnement est évaluée par l'erreur en voxels vis à vis de chaque bord. Chaque couleur représente un type de structure différent. Les résultats des régions présentes dans les deux hémisphères (bilatérales) sont fusionnés.

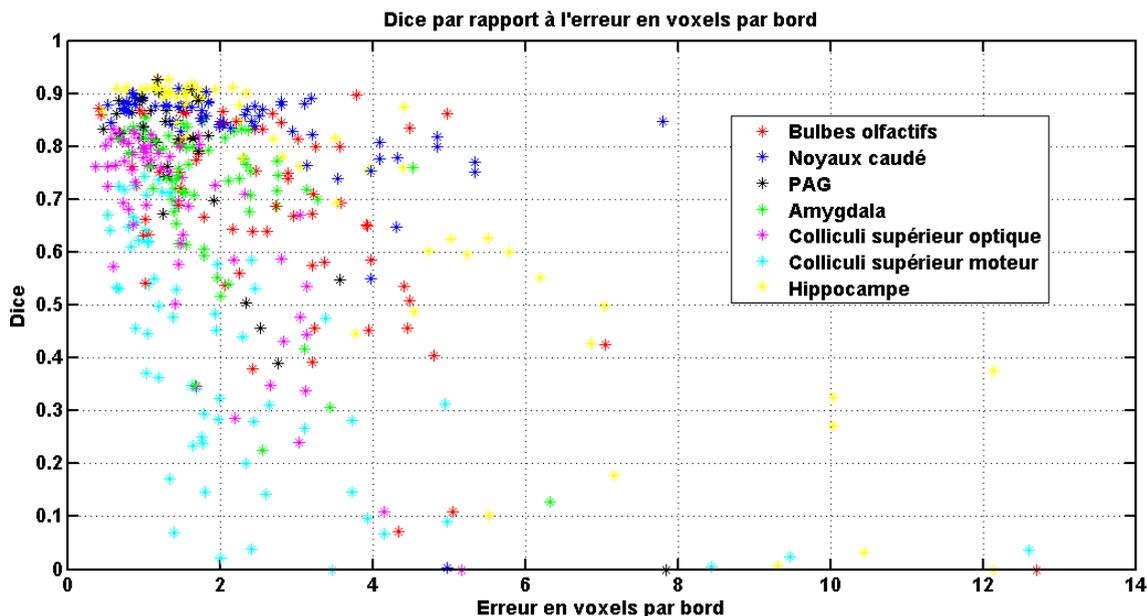


FIGURE 7.19 – Qualité de la segmentation des régions lors d'une expérience utilisateurs. L'indice Dice est représenté en fonction de l'erreur en voxels par bord lors du positionnement.

L'effet du positionnement a un impact différent selon le type de région. Les régions comme les hippocampes et surtout les Noyaux caudés sont très robustes au positionnement de la boîte englobante. Le ratio Dice ne semble pas impacté lorsque l'erreur en voxels par bord augmente. Dans le cas des bulbes olfactifs, des colliculi supérieurs optiques ou des amygdales, la dépendance est plus forte. Lorsque l'erreur en voxels est proche de 0, la qualité de la segmentation est bonne mais celle-ci a de fortes chances de décroître assez vite lorsque l'erreur de positionnement augmente. Enfin dans le cas des colliculi supérieurs moteurs, le bon positionnement ne garantit pas forcément une segmentation de bonne qualité.

Bilan : Deux facteurs jouent probablement un rôle sur le lien entre le positionnement et la qualité de la segmentation d'une région. Le contraste des régions à l'intérieur de l'image ainsi que la taille de celles-ci. Les régions avec un contraste élevé et très visible sont beaucoup plus robustes au mauvais positionnement. Les régions de faible taille sont très sensibles à un positionnement qui serait décalé par rapport à la véritable segmentation. De plus, le ratio Dice est une mesure de qualité qui est sensible à la taille de la région, une erreur de quelques voxels ayant plus d'impact lorsque la région est de taille réduite. Enfin, le recalage la classification peut être plus difficile lorsque la sous-image de travail est trop petite. Ce problème reste cependant relié à la résolution de l'image et n'est pas directement lié intrinsèquement à la structure anatomique.

En pratique, les régions de petite taille et qui semblent peu visibles doivent donc être positionnées avec beaucoup d'attention. Ce qui n'est pas forcément le cas des structures simples et visibles du cerveau. Dans le cas idéal, les choix de l'utilisateur pour guider une segmentation doivent prendre en compte cette caractéristique ainsi que les informations provenant de la section 7.3.2.1 relative à l'ordre de segmentation.

7.3.3 Intérêt de l'interactivité

7.3.3.1 Amélioration de la segmentation avec le positionnement

Cette expérimentation témoigne que le positionnement "parfait" du point de vue de la vérité-terrain n'est pas toujours celui qui fournit la meilleure segmentation. En effet, dans certains cas le positionnement provenant de la vérité-terrain engendre de moins bonnes segmentations qu'un positionnement manuel (plus subjectif).

Bases utilisées : NeuroGeoEx

Protocole : Le bulbe olfactif d'une image de brebis ex vivo a été segmenté en positionnant la boîte englobante de la région à l'aide de la vérité-terrain. Cette même région a été segmentée à l'aide d'un positionnement manuel (par un expert) prenant en compte le type d'erreur que le positionnement correct fournissait. Le même processus a été appliqué sur l'amygdale d'une autre image.

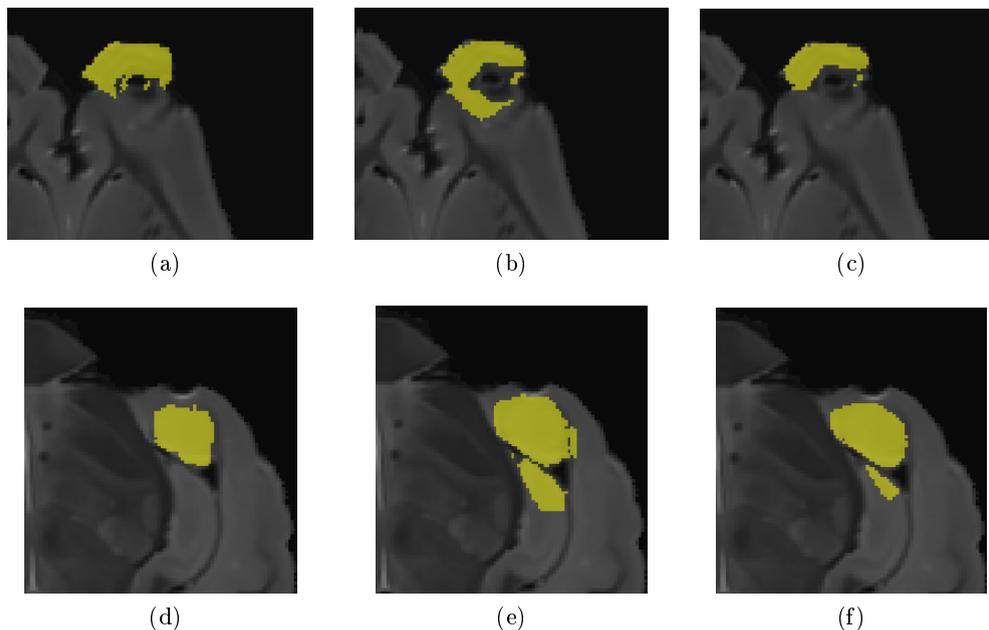


FIGURE 7.20 – (7.20a) Vérité-terrain du bulbe olfactif, (7.20b) segmentation avec la boîte englobante positionnée avec la vérité-terrain, (7.20c) segmentation avec la boîte englobante positionnée manuellement. (7.20d) Vérité-terrain de l’amygdale, (7.20e) segmentation avec la boîte englobante positionnée avec la vérité-terrain, (7.20f) segmentation avec la boîte englobante positionnée manuellement.

Résultat qualitatif : Les résultats de ces segmentations et les vérités-terrain correspondantes sont illustrés sur la figure 7.20. Dans les deux situations, la segmentation provenant du positionnement correct des régions fournit une segmentation de dimension trop importante. L’amygdale (cf. figure 7.20e) déborde sur l’hippocampe tandis que le bulbe olfactif (cf. figure 7.20b) comprend une partie du cortex. Lorsque l’utilisateur positionne lui-même les bords, les erreurs sont amoindries. La segmentation du bulbe olfactif n’est plus de taille trop importante et l’amygdale ne déborde pas de manière aussi importante sur l’hippocampe.

Résultat quantitatif : D’un point de quantitatif, le ratio Dice du bulbe olfactif est 73.7 % lorsque la région est placée automatiquement et de 83.0 % lorsque l’utilisateur positionne lui-même la région. Pour l’amygdale, le résultat Dice augmente de 71,2 % à 76.7 %. Ces deux exemples montrent que dans certaines situations, l’amélioration de la qualité de segmentation pouvant être apportée de manière interactive est loin d’être négligeable.

Bilan : L’interactivité de la méthode provenant du placement de la boîte englobante de la région permet d’influer sur le résultat finalement obtenue. Il peut donc être utile de

visualiser le résultat que fourni un positionnement "normal" (proche de la vérité-terrain) puis de corriger le résultat en proposant un autre positionnement de la région. En pratique, l'utilisateur pourra positionner le bord plus près du centre de la région lorsque la région était trop volumineuse, ou plus loin du centre lorsque la région était de taille trop faible afin de compenser les erreurs du système.

7.3.3.2 Positionnements manuels contre positionnements "parfaits"

Dans le but d'évaluer, l'intérêt de proposer un système interactif et incrémental, nous avons comparé dans cette section les résultats d'une segmentation interactive et d'une segmentation où les régions sont positionnées avec la vérité-terrain.

L'impact de l'interactivité sur les résultats de segmentation est (probablement) plus important lorsque la base d'apprentissage est de taille faible et que l'image à segmenter n'est pas ressemblante à celles présentes dans la base d'apprentissage. C'est pourquoi l'expérience suivante est appliquée sur les images de cerveaux de brebis.

Bases utilisées : NeuroGeoEx

Protocoles : Lors de l'atelier utilisateur que nous avons organisé dans le cadre du projet NeuroGeo, plusieurs images de cerveaux de brebis ont été segmentées par différents types d'utilisateurs. Une méthode de type leave-one-out a été effectuée pour la construction des différents graphes de connaissances *a priori* nécessaires. Ce qui se traduit par l'apprentissage de six différents graphes construits à partir de 5 images. Dans la suite de cette section, nous montrons les résultats de segmentations pour l'image de cerveau de brebis qui a été la plus segmentée (en terme de nombre d'utilisateur). L'image *B2* a été segmentée par 10 utilisateurs différents.

Résultats Quantitatifs : Le tableau 7.21 compare la qualité de la segmentation de l'image de la brebis *B2* produit par des utilisateurs avec la segmentation obtenue lorsque les boîtes englobantes sont positionnées avec la vérité-terrain. On peut remarquer une large disparité entre les segmentations provenant des utilisateurs. Le plus mauvais résultat de segmentation utilisateur est par exemple assez faible. Ceci s'explique par la présence d'utilisateurs non-experts en cerveau de brebis lors de l'expérience. Cependant le cas médian est assez comparable au cas où les bords sont bien positionnés. Seules les colliculi possèdent des segmentations de qualité sensiblement inférieure. Ces régions étant très peu visibles, certains utilisateurs ont eu des difficultés à évaluer la localisation qu'ils avaient effectué. Enfin, on peut voir que la segmentation du meilleur utilisateur est de qualité similaire voir meilleure au cas où les régions sont positionnées à partir de la vérité-terrain, le Dice moyen du meilleur utilisateur étant de 83.8 % et celui des régions bien placées de 83.6 %.

7.4. PERFORMANCES DE LA MÉTHODE PROPOSÉE PAR RAPPORT À L'ÉTAT DE L'ART

Region	Meilleur seg.	Median seg.	Mauvaise eg.	Seg. régions pos. avec la vérité-terrain
Bulb. o g	0.866	0.851	0.425	0.832
Bulb. o d	0.897	0.834	0.509	0.887
Noy. c g	0.882	0.875	0.754	0.898
Noy. c d	0.878	0.870	0.777	0.887
PAG.	0.837	0.816	0.698	0.848
Amy. g	0.855	0.814	0.517	0.847
Amy. d	0.776	0.680	0.348	0.823
Colli. op. g	0.799	0.779	0.681	0.818
Colli. op. d	0.825	0.802	0.686	0.830
Colli. mot. g	0.737	0.598	0.089	0.688
Colli. mot. d	0.724	0.521	0.068	0.713
Hipp. g	0.912	0.908	0.850	0.901
Hipp. d	0.906	0.903	0.446	0.900

FIGURE 7.21 – Qualité de la segmentation (ratio Dice) d'une image de brebis ($B2$) segmentée par différents utilisateurs comparée à un positionnement selon la vérité-terrain.

Bilan : Ces résultats mettent en évidence l'impact positif que peut avoir l'interactivité sur la qualité finale de la segmentation. Il est bien sûr plus difficile pour les utilisateurs non-experts d'obtenir des segmentations cohérentes car il faut pouvoir repérer et corriger d'éventuelles erreurs au cours de la segmentation. Cependant, les utilisateurs experts et connaissant les images à segmenter (anatomie de l'organe), obtiennent des résultats comparables voir meilleurs que lorsque les régions sont positionnées avec la vérité-terrain. Ceci s'explique grâce au itération (essai/erreur) possibles qui permettent de corriger certains défauts.

7.4 Performances de la méthode proposée par rapport à l'état de l'art

Les sections précédentes ont permis de mettre en évidence les paramètres optimaux à utiliser lors d'une segmentation. L'influence de la base d'apprentissage et des interactions utilisateurs ont été analysés. Dans cette section, après avoir démontré l'intérêt d'une approche locale par rapport à une approche globale en terme de performance, la méthode proposée est comparée à différentes méthodes de l'état de l'art sur les différentes bases d'images disponibles.

7.4.1 Étude de l'influence du mode d'apprentissage

Le mode de création et de modélisation des informations *a priori* sous forme d'un graphe permettant un apprentissage local mérite également d'être étudiés. Dans un premier temps,

7.4. PERFORMANCES DE LA MÉTHODE PROPOSÉE PAR RAPPORT À L'ÉTAT DE L'ART

nous justifierons l'utilisation d'informations locales plutôt qu'une approche globale. Ensuite nous comparerons l'utilisation des atlas locaux probabiliste et multi-atlas locaux.

7.4.1.1 Approches locales contre approches globales

Nous étudions ici l'intérêt de l'aspect local de l'approche que nous proposons en comparant les segmentations obtenues avec des multi-atlas locaux à celles obtenues par une approche globale (multi-atlas globaux).

Base MICCAI'12

Protocole : Pour être le plus possible indépendant du mode d'utilisation des atlas, et ne mettre en évidence uniquement l'impact de la localité, une segmentation avec multi-atlas et méthode de vote classique sont utilisées pour les deux approches (locale et globale). En effet, la méthode de classification de voxels par champ de Markov associée aux atlas probabilistes locaux ne fournirait pas le même type de résultats si elle était appliquée sur toute l'image. L'algorithme EM étant dépendant de l'espace sur lequel il est appliqué.

Pour les multi-atlas globaux, le recalage a été effectué à l'aide de la librairie ANTs. Un recalage rigide, puis affine, puis *symétrique diffeomorphe* [11] ainsi qu'une métrique de type SSD sont utilisés. Ensuite la segmentation a proprement dite a été effectuée à l'aide d'une méthode de vote classique sans pondération. Les voxels sont labellisés vers la région où le plus grand nombre d'atlas indique cette région (via la propagation de label).

Dans le cas des multi-atlas locaux, le recalage effectué est de type B-spline. Ce recalage est le même que pour les atlas probabilistes utilisés dans la section 7.2.1. Une méthode de vote classique est aussi utilisée mais de manière incrémental. À chaque nouvelle segmentation de région, si plus de la moitié des atlas indique que le voxel appartient à la région, alors le voxel est labellisé *région*, sinon il n'est pas labellisé. Les relations spatiales ne sont pas utilisées, les régions étant positionnées avec la vérité-terrain.

Les 15 images de la base d'apprentissage ont servi de multi-atlas globaux ainsi que pour construire les multi-atlas locaux.

Résultats Quantitatifs : La figure 7.22 décrit les résultats de segmentation de la méthode multi-atlas avec atlas locaux et avec atlas globaux. La segmentation utilisant les atlas définis localement fournit des résultats de segmentation de meilleures qualités pour presque toutes les régions. Seuls les Ventricules sont mieux segmentés avec les multi atlas globaux. De manière général le ratio Dice moyen obtenu avec les atlas locaux est 79,2 % alors qu'il n'est que de 76.6 % pour les multi-atlas globaux.

Bilan : Cette expérience démontre l'avantage d'une approche locale. Dans le cas où la méthode de classification est la même, les atlas locaux fournissent des segmentations de

7.4. PERFORMANCES DE LA MÉTHODE PROPOSÉE PAR RAPPORT À L'ÉTAT DE L'ART

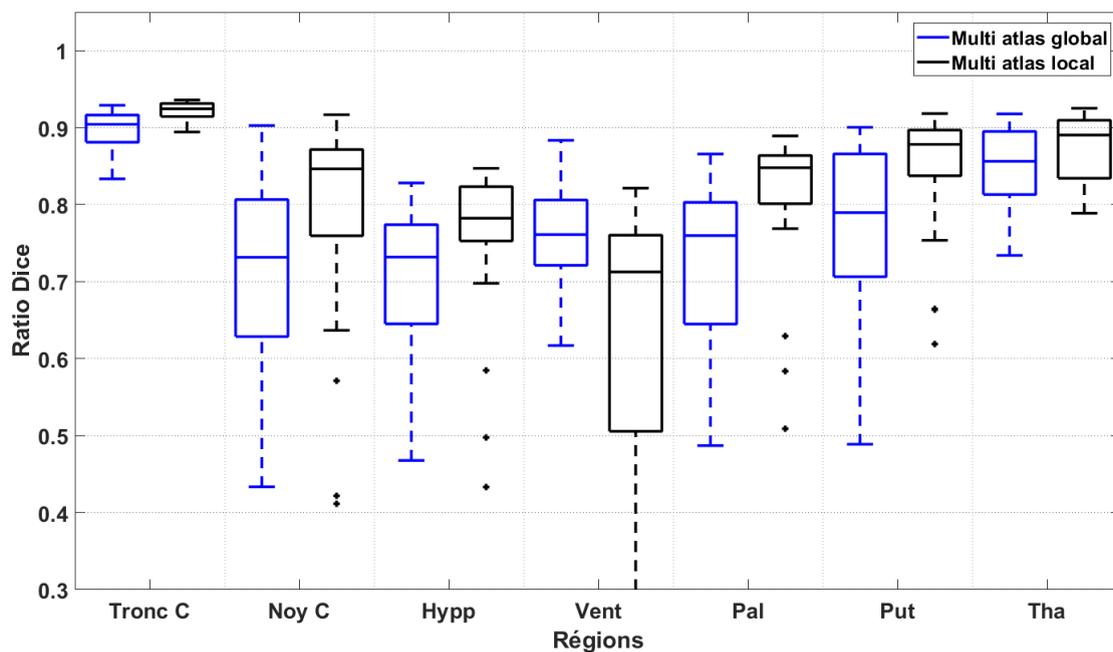


FIGURE 7.22 – Qualité des segmentations obtenues à l’aide des multi-atlas locaux et de multi-atlas globaux. Les diagrammes de boîtes représentent les résultats provenant des 20 images de test disponibles pour chaque type de région.

qualité supérieure aux segmentations fournies par les atlas définis de façon globale. La modélisation des régions sous forme d’atlas locaux a donc des avantages certains au moins lorsque le positionnement de ces derniers est correct.

7.4.1.2 Comparaison multi-atlas locaux et atlas locaux probabilistes

Plusieurs manières de modéliser et d’exploiter les connaissances *a priori* ont été décrites dans la section 5.1. Il est possible d’utiliser des atlas locaux de type probabiliste ou de type multi-atlas. Cependant les segmentations produites par ces deux méthodes ne sont pas identiques. Cette section a pour objectif de comparer la qualité des segmentations produit par ces deux méthodes.

Base d’images : MICCAI’12

Protocole : Trois méthodes ont été appliquées sur la même base d’images : la méthode par atlas probabilistes locaux, la méthode par multi-atlas locaux et méthode de vote simple, et enfin la méthode par multi-atlas locaux avec vote pondéré. Les paramètres de recalage sont ceux utilisés jusqu’à maintenant et sont les mêmes pour chacune des 3 méthodes. Chaque méthode exploite la base d’images d’apprentissage et est appliquée sur les 20

7.4. PERFORMANCES DE LA MÉTHODE PROPOSÉE PAR RAPPORT À L'ÉTAT DE L'ART

images de la base de test. Enfin, deux résultats sont produits correspondant, d'une part, à un positionnement des boîtes englobantes avec la vérité-terrain (E1) et d'autre part, à l'utilisation des relations spatiales apprises (seule la première région étant bien positionnée E2). L'ordre de segmentation et les paramètres de recalage des différentes méthodes sont fixés.

Résultats quantitatifs : La figure 7.23 montre les différents ratios Dice pour la segmentation des images IRM de cerveaux humains dans le cas où les boîtes englobantes sont positionnées à partir de la vérité-terrain. La segmentation avec méthode de vote simple semble donner des résultats inférieurs aux autres méthodes pour l'ensemble des régions et aussi avec une variabilité plus importante. Les résultats des ratios Dice des atlas probabilistes locaux et de la méthode de vote pondéré sont assez similaires. Certaines structures anatomiques telles que les noyaux caudés, les putamens ou les ventricules semblent un peu mieux segmentées avec les atlas probabilistes locaux. Les thalamus et le tronc cérébral semblent quant à eux un peu mieux segmentés avec les multi-atlas locaux. La figure 7.24 illustre les différents ratios Dice lorsque les relations spatiales sont utilisées. Les résultats sont sensiblement similaires à l'expérience précédente. Cependant, la qualité générale de la segmentation de l'atlas probabiliste est légèrement supérieur avec atlas probabiliste dans le cas de la première expérience (+0.8%). Par contre, dans le cas de la deuxième expérience, ce sont les multi-atlas locaux qui obtiennent la meilleure segmentation (+0.3%). Il est possible que la méthode de vote pondéré supporte mieux les mauvais positionnements, en ne donnant qu'un poids faible aux atlas locaux qui ne seraient pas bien recalés à cause d'un mauvais positionnement.

La qualité de la segmentation semble donc proche entre ces deux dernières méthodes, néanmoins le temps de calcul n'est pas le même. La méthode par multi-atlas locaux nécessitent le recalage des 15 atlas locaux de la base d'apprentissage ce qui la rend significativement plus lente que la méthode avec atlas probabilistes locaux qui ne nécessite qu'un seul recalage. Le temps de calcul moyen pour la segmentation d'une région est de 45 secondes dans le cas des atlas probabilistes locaux alors qu'il est de 4 minutes et 40 secondes dans le cas des multi atlas locaux.

7.4. PERFORMANCES DE LA MÉTHODE PROPOSÉE PAR RAPPORT À L'ÉTAT DE L'ART

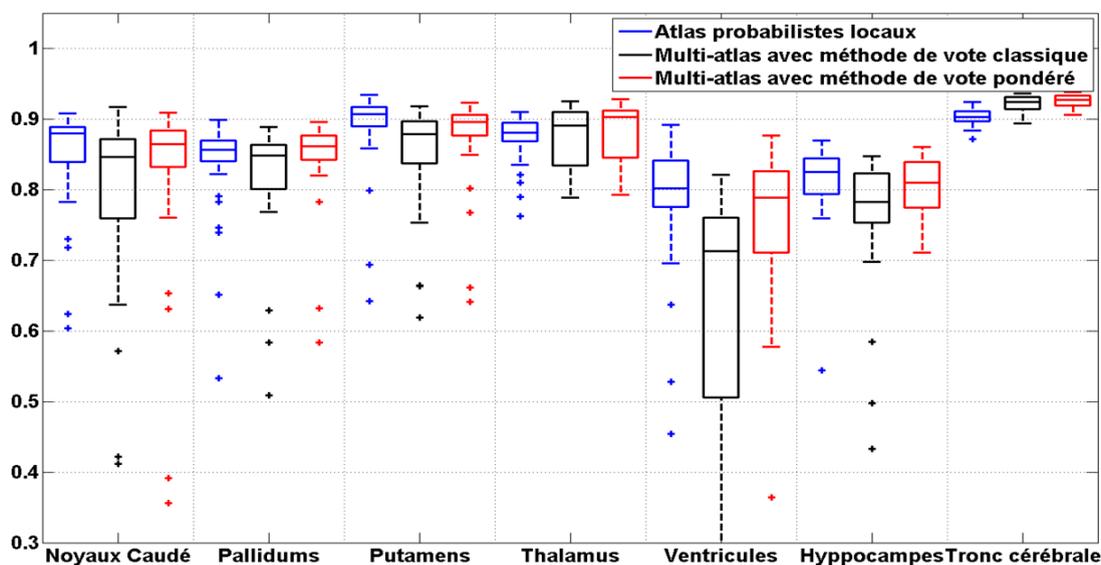


FIGURE 7.23 – Qualité des segmentations obtenues à l'aide des atlas probabilistes locaux et des multi-atlas locaux lorsque les boîtes englobantes sont positionnées avec la vérité-terrain. Les diagrammes de boîtes représentent les résultats provenant des 20 images de test disponibles pour chaque type de région.

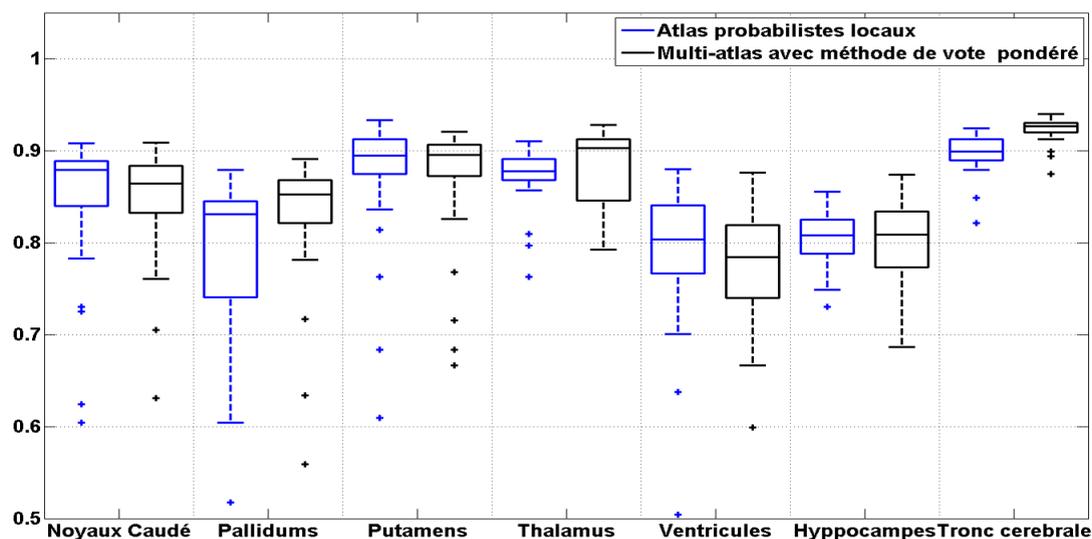


FIGURE 7.24 – Qualité des segmentations obtenues à l'aide des atlas probabilistes locaux et des multi-atlas locaux lorsque les boîtes englobantes sont positionnées avec les relations spatiales (sauf première région). Les diagrammes de boîtes représentent les résultats provenant des 20 images de test disponibles pour chaque type de région.

Bilan : Les résultats avec la méthode multi-atlas n'étant pas significativement meilleurs, il est préférable d'utiliser la méthode la plus rapide. Ceci afin de limiter le temps d'attente de l'utilisateur entre chacune des segmentations locales. C'est pourquoi, les atlas probabilistes locaux ont été privilégiés dans nos travaux et sélectionnés pour produire le prototype du logiciel livré aux participants du projet NeuroGeo.

7.4.2 Comparaison en terme de qualité des résultats

La méthode décrite dans ces travaux étant générale et pas spécifique à certaines structures anatomiques du corps humain (ou animal), nous avons choisi de comparer les résultats de segmentation avec des méthodes de segmentation capables de segmenter des images entières (comme les méthodes multi-atlas). Ces méthodes sont entièrement automatiques. Deux principales méthodes de l'état de l'art ont été sélectionnées et vont être étudiées ici. Dans le cas de la segmentation des images de cerveau humain, FreeSurfer [37] sera utilisé. c'est un outil classique de segmentation d'images cérébrales, à la fois pour les régions sous-corticales que pour les régions corticales. FreeSurfer est dédié uniquement aux cerveaux humains. Les algorithmes et les connaissances *a priori* exploités par cette méthode ayant été conçus spécifiquement pour ce type d'images. La méthode de *Joint Label Fusion* (JLF) décrit dans [98] sera quant à elle appliquée sur l'ensemble des images étudiées (cf. 4.4.2). Aucun apprentissage n'est nécessaire pour FreeSurfer car les informations *a priori* utilisées sont déjà encodées dans le logiciel. Pour la méthode multi-atlas JLF, les mêmes images d'apprentissage que celles permettant d'apprendre le graphe nécessaire à notre méthode seront exploitées.

7.4.2.1 Résultats sur les images de cerveaux humains

Bases utilisées : MICCAI'12

Protocoles : La base d'images de cerveaux humains a été segmentée avec trois méthodes différentes. *FreeSurfer* a été appliqué afin d'obtenir une segmentation sous-corticale des images de cerveaux. La méthode de segmentation multi-atlas *Joint Label Fusion* (JLF) a aussi été appliquée. Cette méthode nécessite le recalage des images de la base d'apprentissage vers la base de test. Le recalage effectué est le même que celui appliqué lors de l'utilisation des atlas globaux (cf. section 7.4.1.1). Enfin, notre méthode a été appliquée de différentes manières :

- une segmentation avec les boîtes englobantes des régions positionnées à l'aide de la vérité-terrain et sans inclusion de la phase de correction *a posteriori* (LISVT)
- une segmentation avec seulement la 1^{ère} région positionnée et sans inclusion de la phase de correction *a posteriori* (LIS₁)
- une segmentation avec les boîtes englobantes des régions positionnées à l'aide de la vérité-terrain et avec inclusion de la phase de correction *a posteriori* (LISVT+)

7.4. PERFORMANCES DE LA MÉTHODE PROPOSÉE PAR RAPPORT À L'ÉTAT DE L'ART

- une segmentation avec seulement la 1^{ère} région bien positionnée et avec inclusion de la phase de correction *a posteriori* (LISVT₁+)

Pour l'ensemble de ces situations, les 15 images de la base d'apprentissage ont été utilisées pour la construction des graphes de connaissances *a priori*. Les 20 images de test ont été segmentées à partir de ces graphes informations *a priori*.

Résultat quantitatifs : La figure 7.25 décrit les ratios Dice pour les différentes régions pour les 3 méthodes étudiées. On peut remarquer que la segmentation par atlas locaux (LISVT, LIS₂, LISTVT+) fournit des résultats intermédiaires. Les résultats sont supérieurs à ceux obtenus par FreeSurfer mais restent en deçà des résultats obtenus par JLF.

Le tableau 7.26 décrit les résultats de Dice moyen de chaque méthode. Les résultats de Freesurfer sont inférieurs à ceux des deux autres méthodes. La méthode JLF fourni des résultats supérieurs de 5 % et 4 % comparés à notre méthode avec ou sans post-correction.

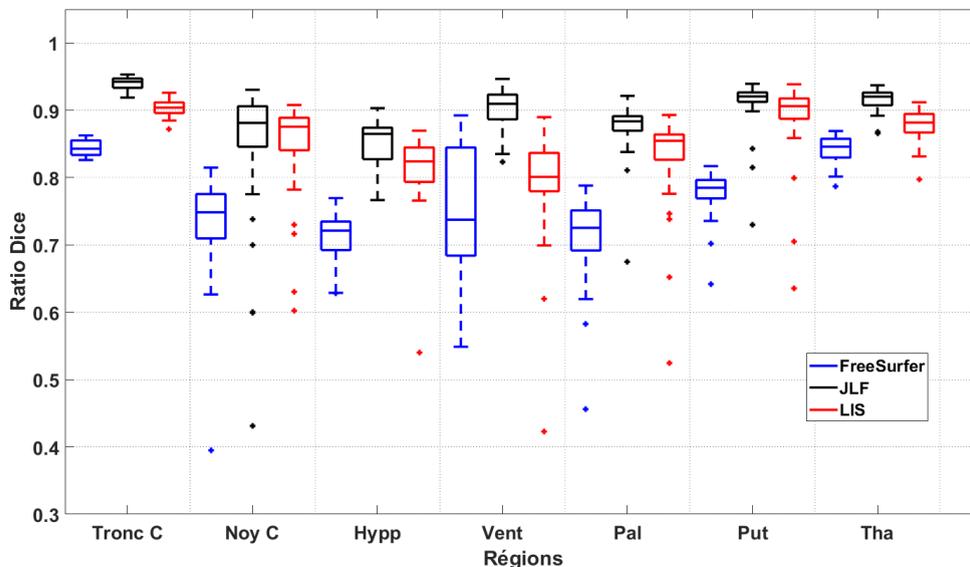


FIGURE 7.25 – Qualité de la segmentation des images de cerveaux humains de notre méthode LIS (régions bien placées), de la segmentation multi-atlas JLF et de Freesurfer. Les diagrammes de boîtes représentent les résultats provenant des 20 images de test disponibles pour chaque type de région.

7.4.2.2 Résultats sur les images de cerveau de brebis

Bases utilisées : NeuroGeoEx

Protocoles Les images de brebis ex vivo ont été segmentées à l'aide de notre méthode ainsi que la méthode JLF. FreeSurfer n'ai pas à même de traiter ce type d'images. La correction *a posteriori* n'est pas ici utilisée. Les boîtes englobantes des régions sont positionnées à l'aide de la vérité-terrain. Pour les deux segmentations, une méthode de type

7.4. PERFORMANCES DE LA MÉTHODE PROPOSÉE PAR RAPPORT À L'ÉTAT DE L'ART

Méthode	Dice moyen
Atlas locaux automatique LIS ₁ (1 région placée)	0.689
Atlas locaux automatique et correction <i>a posteriori</i> LIS ₁ +	0.726
Atlas locaux automatique LIS ₂ (2 régions placées)	0.811
Atlas locaux automatique et correction <i>a posteriori</i> LIS ₂ +	0.824
Atlas locaux bien placé LISVT	0.838
Atlas locaux bien placé et correction <i>a posteriori</i> LISVT+	0.847
FreeSurfer	0.761
JLF	0.888

FIGURE 7.26 – Dice moyen de chaque méthode sur la base MICCAI'12

leave-one-out est utilisée. Six images étant présentes dans la bases d'images, cinq images sont utilisées pour construire les informations *a priori* qui permettront de segmenter la sixième.

Résultats quantitatifs La figure 7.27 montre les résultats de segmentation des différentes régions avec les deux méthodes. La méthode JLF est ici aussi plus efficace. Le Dice moyen de la méthode JLF est de 83 % au lieu de 81.3 % pour la méthode LIS. Notre méthode semble néanmoins plus efficace pour les bulbes olfactifs qui sont des régions de forme et de position très variables, particulièrement sur les images ex-vivo.

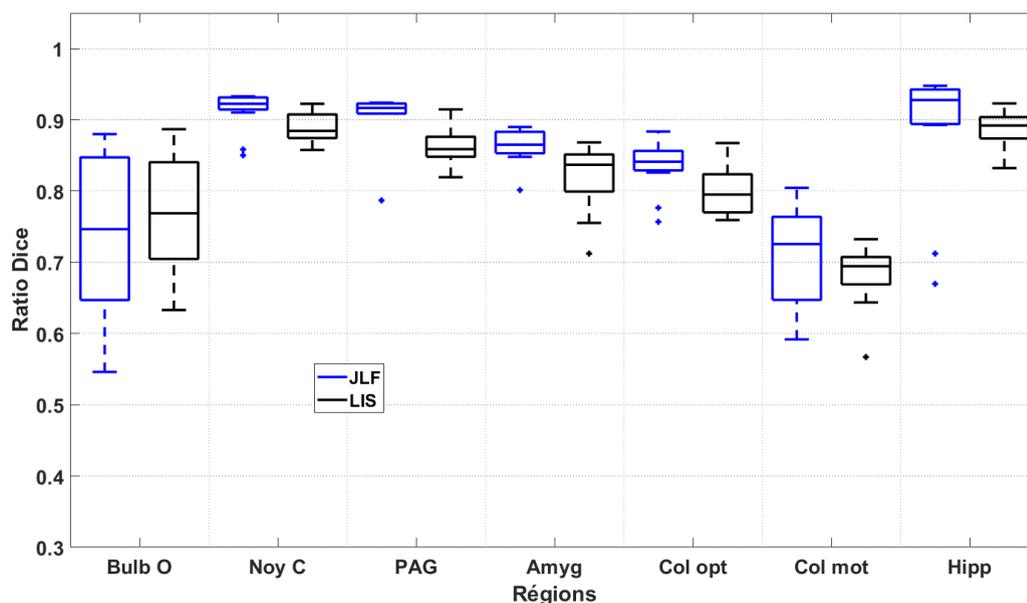


FIGURE 7.27 – Qualité de la segmentation des images brebis ex vivo de notre méthode et de la segmentation multi-atlas JLF. Les diagramme de boîtes représentent les résultats provenant des 6 images disponibles.

Bilan : Ces deux expériences démontrent les qualités des différentes méthodes évaluées sur deux types d'images. La segmentation sous-corticales de FreeSurfer semble moins efficace que celles des deux autres (LIS et JLF). Lorsque la base d'apprentissage est de taille assez importante (pour le cerveau humain), la méthode JLF fournit des résultats significativement meilleurs comparé à la méthode LIS (5 %), lorsque la base d'images d'apprentissage est de taille moins importante (pour le cerveau de brebis), l'écart entre les deux méthodes est beaucoup moins significatif (1.7 %).

7.4.2.3 Images de cœurs humains

Notre outil de segmentation a d'abord été développé pour segmenter des images IRM de cerveaux, et plus particulièrement du cerveau de brebis. Cependant, l'outil n'est pas spécifique à ce type d'images. Il peut ainsi être exploité très facilement sur d'autres problématiques similaires. Dans cette partie, nous décrirons les résultats obtenus par notre approche dans le cadre de la segmentation du cœur à la fois sur des images IRM et des images CT scan. Ces résultats correspondent à notre participation au concours durant le workshop STACOM 2017.

Bases utilisées : MM-WHS'17

Protocole : Pour cette évaluation, deux types de graphes ont été appris : Un graphe pour les images IRM et un graphe pour les images de CT scans. Les deux bases d'images ont été découpées en 4 parties avec 5 images afin de faire une segmentation de type leave-5-out. Ainsi 15 images ont été utilisées pour apprendre l'ensemble des informations a priori (graphe et classifieurs pour la correction *a posteriori*) dans le but de segmenter les 5 images restantes. Ce processus a été reproduit 4 fois pour chaque base d'images. Dans le cas des images CT scan, une transformation des intensités a été effectuée pour l'apprentissage ainsi que la segmentation. La plage de valeurs des intensités de certaines images étant très importante, nous avons choisi de fixer les limites inférieure et supérieure à -800 et 800. Les valeurs d'intensité inférieurs à -800 et supérieurs à 800 sont donc respectivement fixées à -800 et 800. Enfin la métrique liée au recalage utilisée est de type information mutuelle car celle-ci fournissait des recalages de meilleures qualités.

— Apprentissage des atlas locaux :

Pour la création des atlas locaux. La taille des régions étant plus importante et les formes plus variables que pour les images de cerveaux, il a été plus efficace de recalibrer les vérités terrains entre elles puis d'appliquer la même transformation aux images réelles (IRM ou CT scan), plutôt que d'utiliser les images IRM pour faire le recalage et déterminer la transformation nécessaire.

— Apprentissage de classifieurs de correction *a posteriori* :

7.4. PERFORMANCES DE LA MÉTHODE PROPOSÉE PAR RAPPORT À L'ÉTAT DE L'ART

Chaque graphe a permis de segmenter les images qui ont été utilisées lors de leur construction afin d'obtenir les informations nécessaires pour l'apprentissage de l'étape de correction a posteriori. Les cartes de probabilité par région provenant du champ de Markov sont utilisées comme entrées pour l'apprentissage des classifieurs *Adaboost*. Les images utilisées pour construire le graphe sont les mêmes qui sont segmentées pour apprendre les classifieurs.

Deux expérimentations sont réalisées :

- Une expérience (E1) où les boîtes englobantes des régions sont positionnées en utilisant la vérité-terrain.
- Une expérience (E2) où seule la première structure est positionnée en utilisant la vérité-terrain. Les autres régions étant positionnées automatiquement en utilisant les relations spatiales. La première région segmentée est le Ventricule gauche, car c'est l'une des structures les plus visibles et stables à segmenter dans les images IRM et scanner de cœur, qui peut ainsi fournir des informations cohérentes pour de la suite de la segmentation.

Résultats qualitatifs : De manière générale, les segmentations produites par notre méthode sont de bonne qualité pour la segmentation des images de CT scan. La forme des régions est très souvent bien respectée comme le montre la figure 7.28a. On peut cependant noter la présence de frontières entre les régions qui ne sont pas toujours nettes. La frontière entre le Ventricule droit (bleu) et l'artère pulmonaire (blanc), est par exemple, assez floue (cf. figure 7.28a). Les images IRM sont en revanche plus difficiles à segmenter. Les régions sont globalement bien représentées mais des débordements apparaissent plus régulièrement comme on peut le voir sur la figure 7.28b.

Résultats quantitatifs : Le tableau 7.8 décrit les ratios Dice de chaque région obtenue lors des différentes expériences. On remarque que la segmentation des images de CT scan est de bonne qualité dans le cas (E1). Tous les résultats Dice sont supérieurs à 88 %. Lorsque les relations spatiales sont utilisées, la qualité des segmentations reste supérieure à 81 % pour 6 régions. On peut remarquer que la qualité de segmentation des aortes ainsi que de l'atrium droit diminue. Les aortes ont des formes et surtout des positions et tailles variables. Cette variabilité peut être difficile à déterminer avec les relations spatiales. Dans le cas des images IRM, les ratios Dice obtenus pour chaque région sont un peu plus faibles, en particulier pour les aortes. Ici encore, leur variabilité ainsi que les artefacts IRM rendent difficiles la détection de ces structures.

Bilan : Ces expériences mettent en évidence la généralité de la méthode que nous proposons qui peut être appliquée à différents types d'images, aussi bien pour ce qui concerne le

7.4. PERFORMANCES DE LA MÉTHODE PROPOSÉE PAR RAPPORT À L'ÉTAT DE L'ART

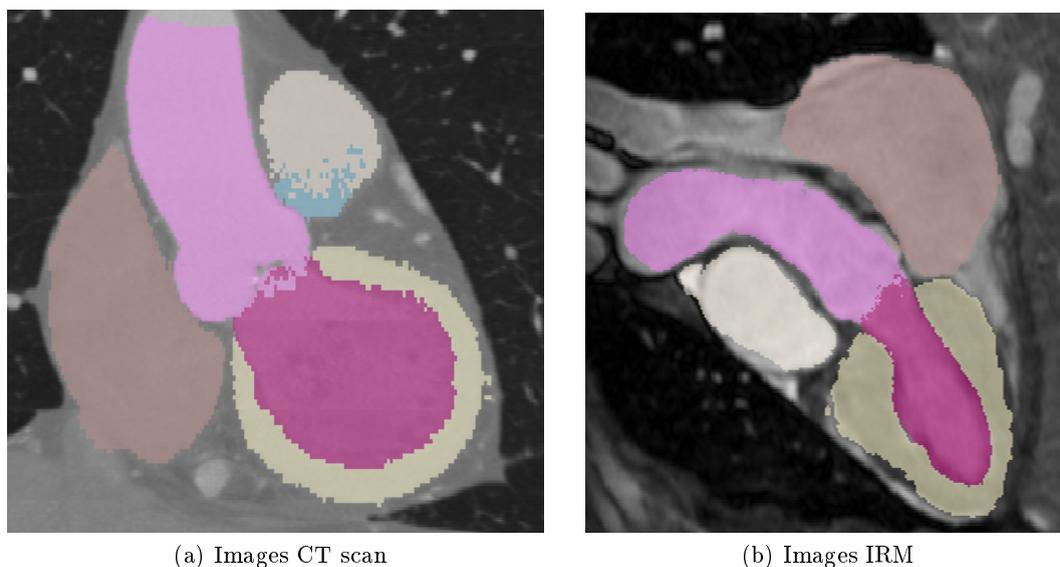


FIGURE 7.28 – Résultat de segmentation du cas médian de l'expérience (E1) sur les images CT scan et sur les images IRM.

TABLE 7.8 – Dice Ratio pour les différentes expériences et structures du cœur

Régions	E1 CT	E2 CT	E1 MR
Ventricule gauche	88.2 ± 21.1	88.1 ± 21.6	88.4 ± 6.5
Ventricule droit	88.8 ± 3.8	86.3 ± 7.4	82.4 ± 12.4
Myocardium du ventricule gauche	88.2 ± 3.5	87.4 ± 7.9	75.5 ± 9.7
Atrium droite	89.0 ± 4.4	81.3 ± 20.2	85.3 ± 5.6
Aorte (ascendante)	88.1 ± 0.21	81.3 ± 20.2	77.1 ± 10.9
Aorte pulmonaire	82.0 ± 12.1	72.3 ± 24.8	67.7 ± 15.5
Atrium gauche	92.3 ± 3.5	86.0 ± 19.4	78.4 ± 5.4

type d'organe étudié que la modalité utilisée. Cette généralité nous a permis de participer à un challenge du workshop STACOM associé à la conférence MICCAI.

7.4.3 Comparaison en terme de temps de calcul

L'approche que nous proposons étant interactive, le temps nécessaire à la segmentation est un critère très important. Les temps de traitements doivent être suffisamment courts pour que l'utilisateur ne patiente pas trop avant de pouvoir poursuivre sa segmentation. Dans cette section nous analysons le temps de calcul nécessaire à une segmentation ainsi que le temps nécessaire aux interactions lorsqu'un utilisateur intervient dans le processus.

7.4. PERFORMANCES DE LA MÉTHODE PROPOSÉE PAR RAPPORT À L'ÉTAT DE L'ART

Méthode	Temps
JLF	1 h 10 + 5 h 45 min
Freesurfer	11 h
Multi atlas global	1 h 10
LIS sans utilisateur	11 min
LIS sans utilisateur avec correction	17 min
Multi-atlas locaux	61 minutes

FIGURE 7.29 – Temps de calcul mesurés sur des machines différentes nécessaire pour obtenir la segmentation de 13 régions anatomiques. Le temps de la méthode JLF comprend le temps de recalage des images ainsi que le temps de calcul de segmentation à proprement dit. Le temps de calcul de la notre méthode correspond au temps nécessaire à l'enchaînement des 13 segmentations sans tenir compte des temps d'interaction.

7.4.3.1 Analyse des temps de calculs

La base d'images de cerveaux humains MICCAI'12 a été segmentée dans les sections précédentes avec notre méthode ainsi qu'avec FreeSurfer et JLF. Durant ces expériences, nous avons mesuré les temps de traitement utilisées pour chaque méthode. Le temps mesuré et indiqué pour notre méthode représente le temps nécessaire à la segmentation des 13 régions anatomiques recherchées. Les segmentations de FreeSurfer et JLF ont été effectuées sur une machine virtuelle linux (4 cœur, 16Go ram) sur laquelle l'accessibilité à l'ensemble des ressources ne peut être garantie à 100 % à chaque instant. Les temps de segmentation sont donc donné à titre indicatif mais ils sont tout de même très représentatifs de l'ordre de grandeur de la durée nécessaire à la segmentation d'une image. Pour notre méthode, les temps de calcul ont été mesurés sur un ordinateur portable HP sous Windows 10 (2-4 cœur, 16Go ram). Le temps de calcul mesuré ne prend pas en compte le temps nécessaire à l'utilisateur pour guider la segmentation.

Le tableau 7.29 présente les différents temps de calcul nécessaire aux différentes méthodes. On peut remarquer que Freesurfer et JLF nécessitent plusieurs heures de calcul pour produire le résultat de segmentation des 13 régions. La segmentation multi-atlas classique qui ne nécessite que le recalage des images est un peu plus rapide (1 h 10). Notre méthode fournit la segmentation désirée en un temps beaucoup plus court (11 min). Ces méthodes peuvent effectivement fournir une segmentation complète du cerveau comprenant une multitude de régions ce qui n'est pas le cas de notre méthode LIS. Cependant, si l'utilisateur souhaite ne segmenter qu'une partie des régions présentes dans une image alors le temps de segmentation est sensiblement moins long. Dans le cas des atlas probabilistes et sans correction *a posteriori*, le temps moyen de segmentation d'une région sous-corticales du cerveau humain est inférieur à une minute. L'ajout de la correction *a posteriori* augmente légèrement le temps de segmentation (1 min 20).

Lorsque le temps d'interaction n'est pas pris en compte ou lorsque la seule première région est positionné manuellement, on peut voir que l'utilisation d'atlas locaux fournit des segmentation plus rapide que les méthodes de la littérature présentée dans ce chapitre. La

7.4. PERFORMANCES DE LA MÉTHODE PROPOSÉE PAR RAPPORT À L'ÉTAT DE L'ART

différence de temps entre une méthode utilisant des atlas classiques et une notre méthode locale dépend du nombre de régions que l'on souhaite segmenter et de leur taille. Bien évidemment, moins le nombre de région à segmenter est important, plus la segmentation avec LIS sera rapide. La taille des régions joue aussi un rôle sur ce temps de segmentation. Plus les régions à rechercher sont petites (ou en tout cas leur boîte englobante) par rapport à la taille de l'image entière, plus la segmentation avec LIS sera avantageuse d'un point de vue des temps de calcul. A contrario, les régions de taille très importantes conduisent à l'utilisation d'atlas locaux devenant similaires aux atlas classiques.

7.4.3.2 Temps d'interaction

Lorsque l'outil de segmentation est manipulé par des utilisateurs, le temps de segmentation ne se réduit pas aux simples temps de calcul. En effet, l'utilisateur doit analyser l'image pour choisir quelle région il croit bon de segmenter à l'étape suivante. Il doit observer le positionnement automatique et le corriger si nécessaire. Enfin, il doit visualiser et analyser le résultat de la segmentation pour décider si le résultat obtenu lui convient. Toutes ces opérations peuvent être coûteuses en temps. Ce temps d'interaction peut aussi être une indication pour juger de la facilité de prise en main de l'outil fourni aux utilisateurs.

Comme nous l'avons déjà mentionné, dans le cadre du projet NeuroGeo, un atelier utilisateurs a été organisé afin d'évaluer notre méthode. Lors de cet atelier, plusieurs images ont été segmentées par les mêmes utilisateurs. Lors de ces segmentations, les utilisateurs étaient libres dans leurs manipulations afin d'obtenir la segmentation des 13 régions du cerveau de brebis de la base NeuroGeoEx. Nous nous intéressons à la segmentation des images de cerveau de brebis, car le nombre de personnes et le nombre d'images de brebis segmentées sont plus élevés que pour les images de cerveau humain. Le temps de segmentation de chaque image et de chaque utilisateur a été enregistré. Le temps mesuré est le temps entre le choix de la première région à segmenter et la fin de la segmentation de la 13^{ème} région.

Au fil des segmentations successives, les utilisateurs ont pu se familiariser avec le logiciel. Ce qui se traduit par une diminution du temps nécessaire à la segmentation au cours des segmentations. La figure 7.30 décrit le temps moyen nécessaire à la segmentation de l'image en fonction du nombre d'images que la personne a déjà segmentées. L'image 1 est la première image de brebis que l'utilisateur segmente avec notre outil de segmentation. Lors de la segmentation de la 3^{ème} image, le temps de segmentation moyen de l'ensemble des régions présentes dans l'image de brebis est de 54 min.

7.4. PERFORMANCES DE LA MÉTHODE PROPOSÉE PAR RAPPORT À L'ÉTAT DE L'ART

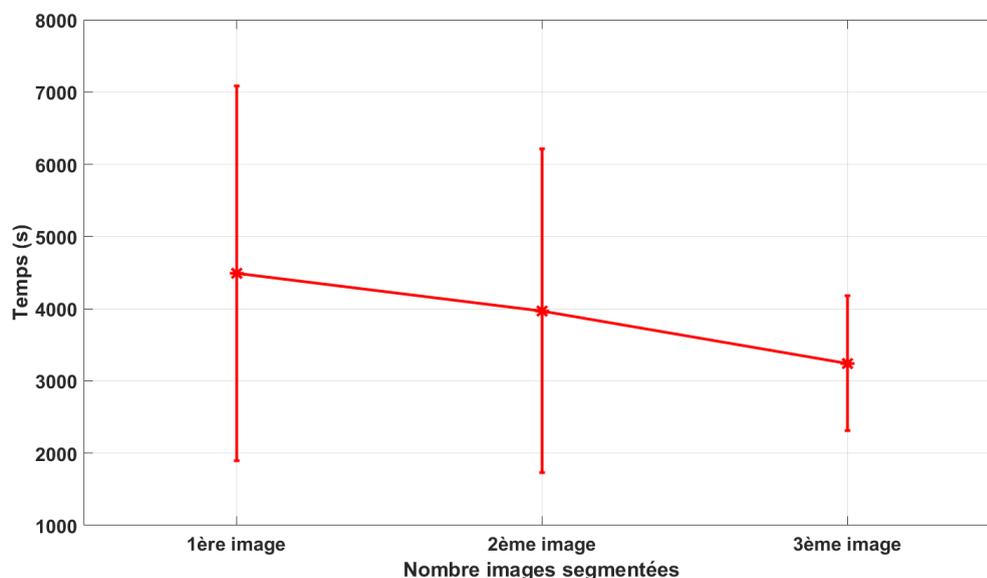


FIGURE 7.30 – Évolution du temps moyen nécessaire à la segmentation des 13 régions des images de cerveaux de brebis ex-vivo en fonction du nombre d'image déjà segmenté par l'utilisateur.

Cette expérience met en évidence le fait que la prise en main de l'outil de segmentation est assez rapide. Les utilisateurs s'habituent rapidement à utiliser le logiciel. Le temps nécessaire aux différentes actions étant réduit après la segmentation de seulement quelques images. On peut imaginer que ce temps de segmentation peut encore diminuer car le logiciel actuel est loin d'être un produit fini. Il s'agit actuellement d'un prototype de laboratoire dont l'ergonomie n'a pas été travaillé. À titre indicatif, un utilisateur expert du logiciel peut obtenir une segmentation des 13 régions d'une image brebis (ou humain) en moins de 20 minutes. Le temps total est de l'ordre de une heure pour des utilisateurs prenant en main l'outil. Ce temps de segmentation reste donc toujours plus rapide que les méthodes de type FreeSurfer et JLF. Ils sont du même ordre de grandeurs que la segmentation multi-atlas classique. Cependant, comme montré dans la section 7.4.1.1, notre méthode est potentiellement plus efficace.

Chapitre 8

Le prototype 3DBrainSeg et son utilisation

L'ensemble des travaux décrit dans les chapitres précédents a débouché sur la création d'un prototype logiciel permettant de segmenter des images médicales 3D. Ce logiciel est nommé 3DBrainSeg. Dans ce chapitre, nous présenterons les caractéristiques de ce prototype qui a été exploité durant un atelier utilisateurs effectué dans le cadre du projet NeuroGeo visant à segmenter des cerveaux de brebis.

8.1 3DBrainSeg

8.1.1 Implémentation

L'implémentation de 3DBrainSeg a été effectuée sous Windows avec le langage C++ et des bibliothèques du framework .NET. Les graphes contenant les connaissances *a priori* apprises sont gérés à l'aide d'une librairie de manipulation de graphes développée au laboratoire. Les recalages d'images non-linéaires utilisés lors de la construction des atlas locaux probabilistes ou lors de l'utilisation des atlas globaux, sont effectués via la librairie ITK. La visualisation des résultats passe par une scène 3D gérée par la librairie Ogre.

La création des graphes de connaissances *a priori* nécessite un ensemble d'images au format *nifti* composé d'images réelles et d'images labels correspondantes. Le stockage des connaissances *a priori* se fait sous la forme d'un graphe au format *.gxl*, dont les nœuds pointent vers des atlas locaux (composés d'images template et de cartes de probabilité), et d'une liste de fichiers textes contenant les informations de chaque classifieur Adaboost.

L'état courant de la segmentation est représenté sous la forme d'un graphe au format *.gxl* et d'une image contenant pour chaque voxel les labels de leur région d'appartenance. Ce graphe représente la segmentation actuelle. Les nœuds sont les régions déjà segmentées et les arcs représentent l'adjacence entre ces régions.

8.1.2 Fonctionnalités

Lors du déroulement d'une segmentation interactive, plusieurs fonctionnalités sont proposées afin de permettre une segmentation incrémentale et interactive.

- Création d'un graphe de connaissances *a priori* :
Les images présentes dans la base d'apprentissage sont utilisées pour construire les atlas locaux, les relations spatiales et les classifieurs Adaboost. Une fonction *sauvegarder graphe* permet ensuite de choisir la localisation de stockage des connaissances *a priori* apprises. Si un graphe est déjà chargé, un apprentissage incrémental est appliqué et les informations de la base d'apprentissage sont ajoutées aux informations déjà présentes dans le graphe.
- Chargement d'un graphe de connaissances *a priori* :
Les informations *a priori* disponibles sont chargées et permettent d'avoir à disposition les atlas locaux, les relations spatiales ainsi que les classifieurs Adaboost nécessaires pour la correction *a posteriori*. Le chargement d'un graphe est nécessaire sinon la segmentation d'une image ne peut avoir lieu.
- Chargement d'une modalité image :
L'image réelle 3D à segmenter est chargée (au format *nifti*). Plusieurs modalités peuvent être chargées si elles sont positionnées dans le même espace (même taille).
- Sélection de la région à segmenter :
L'utilisateur choisi, parmi la liste des régions présentes dans le graphe chargé, la région qu'il souhaite segmenter. Si des régions ont déjà été segmentées, l'outil fournit une position probable de la région grâce aux relations spatiales disponibles dans le graphe.
- Positionnement/Correction la boite englobante :
L'utilisateur positionne les bords de la boite englobante de la région qu'il veut segmenter. Six bords au total doivent être positionnés. L'utilisateur agit directement sur les coupes 2D. Un exemple de visualisation de boite englobante est fourni sur la figure 8.3.
- Validation du positionnement :
L'utilisateur accepte le positionnement de la boite englobante qui lui est présenté. Celui-ci peut avoir été défini aussi bien manuellement que automatiquement (relations spatiales). Lorsque l'utilisateur accepte le positionnement, la segmentation par HMF est lancée. L'utilisateur n'intervient plus jusqu'à la fin de la segmentation de la région.
- Rejet du positionnement :
Le positionnement proposé, qui peut être manuel ou automatique, est réinitialisé afin de laisser l'utilisateur repositionner la région. Les bords (rouge) de la boite englobante sont ainsi effacés sur les coupes 2D.
- Retour à la segmentation précédente :
La dernière segmentation effectuée est annulée. Avant chaque nouvelle segmentation, l'état courant de la segmentation est sauvegardée dans un fichier temporaire. Lorsque

l'utilisateur annule la segmentation d'une région, la précédente segmentation est rechargée.

- Sauvegarder du projet :

L'utilisateur sélectionne une localisation pour sauvegarder la segmentation actuelle. Elle se compose d'une image labels au format *nifti* mais aussi d'un graphe représentant la structure de la segmentation actuelle. Ce graphe attribué permet de stocker des annotations ou des informations de couleur pour chaque nœud (région) et arc (relation d'adjacence) du graphe pour améliorer la visualisation.

- Chargement d'un projet :

Une segmentation précédemment sauvegardée est rechargée. L'utilisateur peut continuer la segmentation après avoir rechargé le graphe de connaissances *a priori* correspondant et l'image réelle 3D.

- Visualisation :

À chaque étape, l'utilisateur a la possibilité de visualiser les résultats obtenus (cf. IHM 8.2). Trois coupes 2D ainsi qu'une scène 3D sont disponibles afin de pouvoir visualiser l'image réelle ou sa segmentation. La scène 3D permet de visualiser simultanément les différentes régions déjà segmentées et les coupes de l'image réelle.

La figure 8.1 décrit le schéma classique d'utilisation du logiciel lorsque celui-ci est utilisé de manière libre par un utilisateur. Celui-ci peut intervenir à toutes les étapes présentées excepté durant l'étape de post-correction. La sauvegarde et le chargement d'une segmentation ne sont pas représentés ici. En plus des fonctionnalités permettant de segmenter des images, le logiciel permet également la création (apprentissage) de graphes de connaissances *a priori*.

8.1.3 Paramétrage des méthodes

Lors de ces travaux un grand nombre de paramètres et de modélisations ont été présentés et comparés. Certains paramètres sont fixés et utilisés par défaut par le logiciel 3DBrainSeg.

La liste des paramètres par défaut utilisés par 3DBrainSeg est :

- Utilisation d'atlas locaux probabiliste
- Marge de 10 % autour de la boîte englobante des régions pour la construction et l'utilisation des atlas locaux
- Normalisation des intensités avec une transformation linéaire par morceaux et l'utilisation de 3 quartiles
- Le transformation non-linéaire du recalage est de type B-spline avec 6 nœuds par dimension
- La métrique pour le recalage est la somme des distances au carré
- Le recalage est appliqué à un niveau de sous échantillonnage de 2.

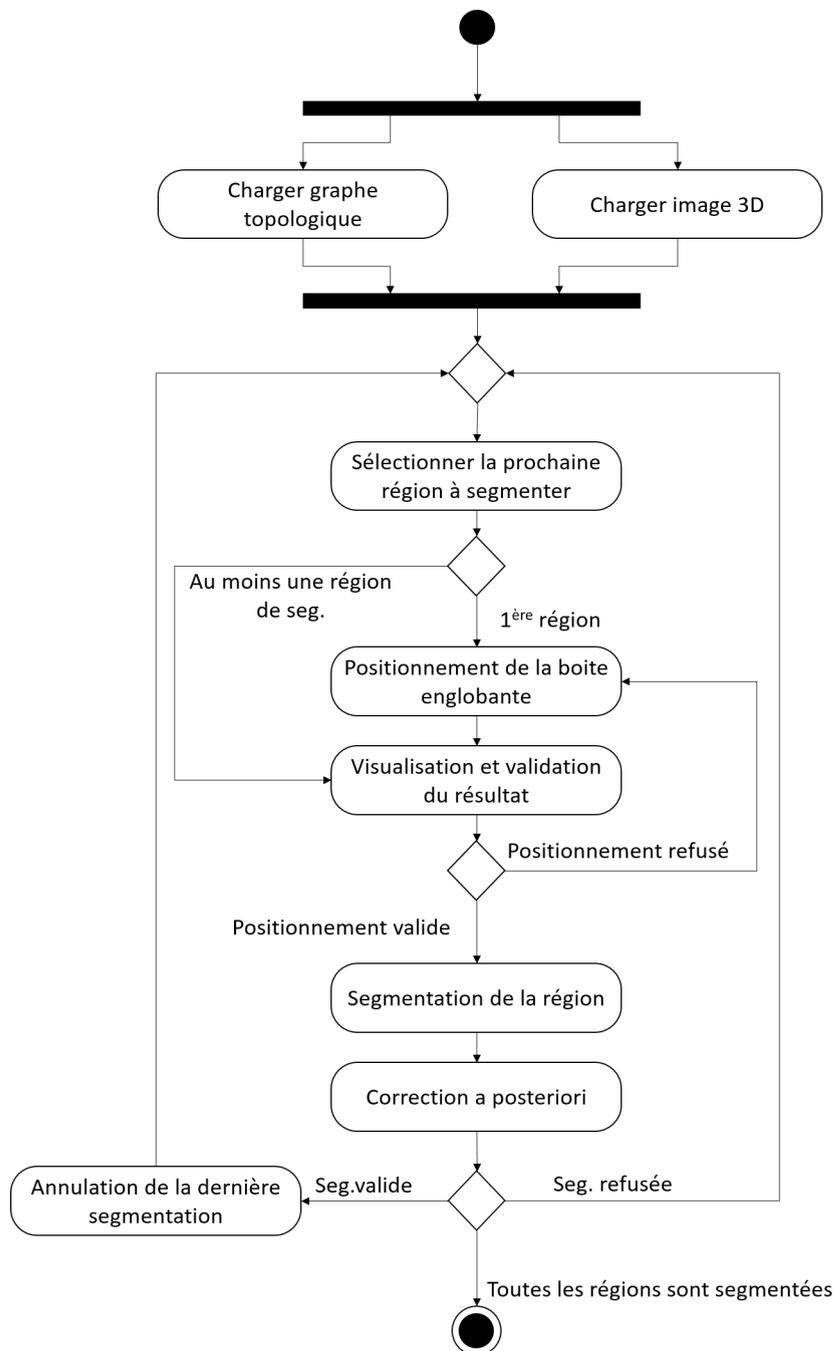


FIGURE 8.1 – Diagramme d’activité du logiciel 3DBrainSeg. En plus ces interactions, l’utilisateur peut enregistrer ou charger une segmentation.

- Le nombre de gaussiennes pour le K-means et le champs de Markov est de 3.
- Le coefficient de d’importance de l’atlas est de 0.75.
- Le coefficient de d’importance de voisin est de 0.05.

- Le coefficient d'importance de l'atlas via l'entropie des intensités est de 2.
- Les relations spatiales sont des intervalles de distance (minimum et maximum relatives)
- Correction *a posteriori* avec une taille de patch de 5x5x5 et une dilatation de x voxels.

Ces paramètres se sont révélés les plus efficaces pour les situations qui nous ont été soumises durant ces travaux. Cependant, il est tout a fait possible de modifier l'un de ces paramètres simplement au travers d'une interface de configuration / d'un fichier de configuration.

8.1.4 Interface graphique

L'interface homme machine a été développé à partir de *Windows Form*. Elle permet à l'utilisateur de visualiser facilement l'état d'une segmentation à chaque étape. La figure 8.2 présente les interactions que l'utilisateur peut effectuer via l'IHM :

- (1) Visualisation des coupes 2D de l'image courante.
- (2) Scène 3D. Les 3 coupes 2D sont visibles. Le déplacement des coupes 2D [1] permet de déplacer aussi les coupes sur le modèle 3D. Les régions déjà segmentées à un instant donné sont aussi visibles à l'intérieur de la scène 3D.
- (3) L'onglet *Affichage* permet de changer l'image courante (visible sur les coupes 2D et la scène 3D). Il est possible de visualiser les différentes modalités si elles sont disponibles ainsi que l'image des labels de région.
- (4) Liste des régions déjà segmentées, la région 1 étant le fond/extérieur. Cocher et décocher les régions permet de visualiser ou cacher les régions dans la scène 3D.
- (5) Permet d'écrire des annotations ou commentaire sur une régions qui seront sauvées dans le graphe lors de la sauvegarde du projet.
- (6) Modifie la transparence des régions dans la scène 3D.

8.2 Atelier « Utilisateurs » NeuroGeo et retours d'expériences

L'outil de segmentation présenté dans ces travaux étant interactif, un atelier « utilisateurs » a été organisé afin d'évaluer les fonctionnalités proposées (qualité des dispositifs d'interaction) ainsi que les résultats ou performances pouvant être obtenus par différents types d'utilisateurs.

Plusieurs résultats déjà présentés dans les chapitres précédents proviennent de cette expérimentation utilisateurs. Dans ce chapitre, nous présentons de manière plus détaillée l'organisation de cet atelier, les différentes expériences effectuées ainsi que les retours et commentaires que les différents participants nous ont communiqués.

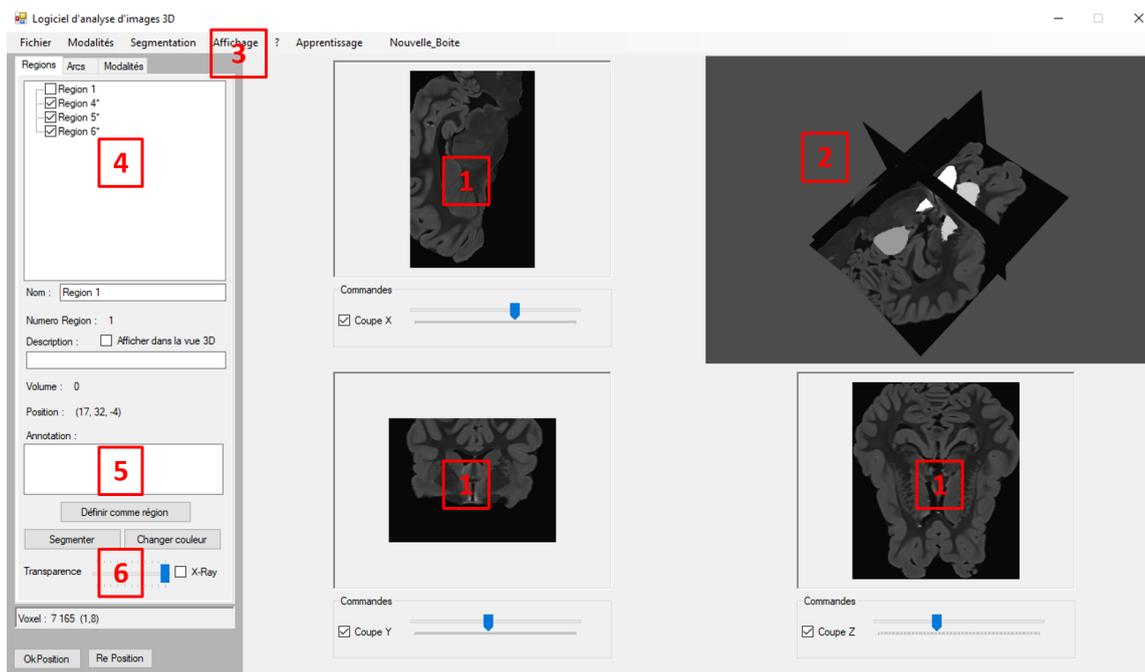


FIGURE 8.2 – Interface graphique du logiciel 3DBrainSeg. Exemple avec une image de cerveau de brebis de la base NeuroGeoEx et 3 régions segmentées.

8.2.1 Organisation de l'atelier

8.2.1.1 Organisation et participants

L'atelier a été organisé sur 2 journées. Une journée a été consacrée à la segmentation d'images de cerveaux humains provenant de la base MICCAI'12 tandis que la 2^{ème} journée s'est focalisée sur la segmentation d'images de cerveaux de brebis (correspondant plus au projet NeuroGeo). Lors de cette 2^{ème} journée, les participants ont pu segmenter des images de la base NeuroGeoEx (cf. 7.1.1.2) mais aussi des images in vivo de cerveaux de brebis.

19 personnes ont participé à cet atelier :

- 15 personnes pour les images de cerveaux humains
- 13 personnes pour les images de cerveaux de brebis

Nous avons essayé de sélectionner les participants parmi un panel de profils diversifiés :

- 11 participants étaient des experts en anatomie cérébrale de l'INRA ou de l'INSERM ; les personnes de l'INRA étant spécialisées dans le cerveau de brebis et celles de l'INSERM dans les cerveaux humains
- 4 personnes avaient des compétences en imagerie médicale et étaient familières à l'utilisation d'outils de visualisation ou segmentation d'images (avec des connaissances des concepts de base de l'anatomie du cerveau)

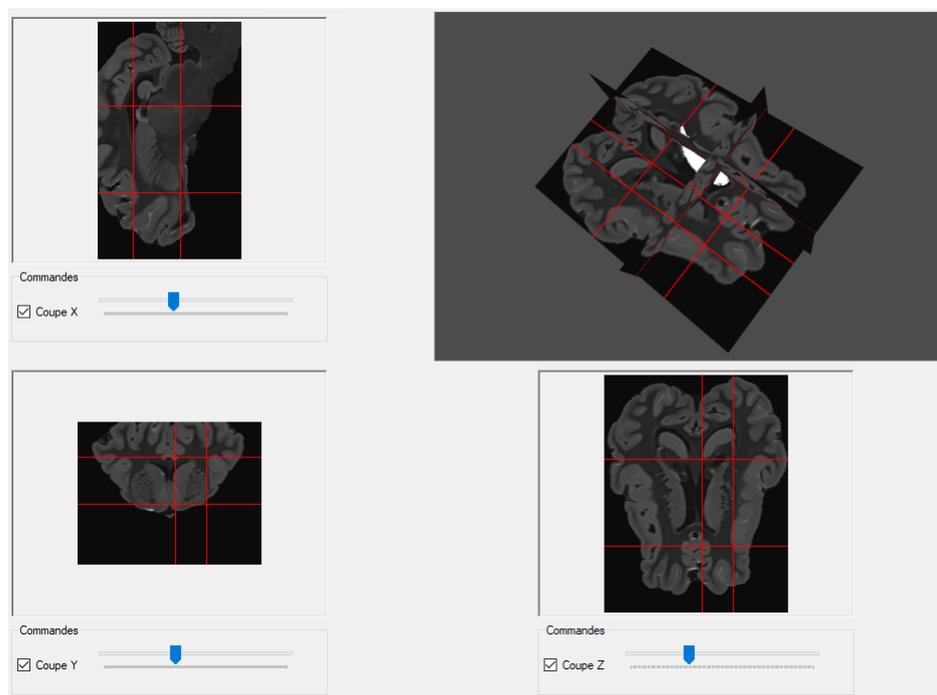


FIGURE 8.3 – Visualisation du positionnement d'une boîte englobante (Noyau caudé d'une image de cerveau de brebis de la base NeuroGeoEx).

- 4 personnes étaient des informaticiens n'ayant que peu de connaissances anatomiques du cerveau ou de l'imagerie médicale. Elles avaient par contre des compétences de base en analyse d'images (pixels, segmentation, régions, contours, ?).

Une formation rapide à l'usage du logiciel (1 heure) a été réalisée en début d'atelier. Durant cette présentation, les principales fonctionnalités du logiciel ont été rappelées (cf section précédente) et quelques recommandations mentionnées. Tous les participants avaient, au préalable, eu la possibilité de consulter un site web¹ proposant une vidéo de démonstration du logiciel, une documentation textuelle et une machine virtuelle contenant 3DBrainSeg avec des données permettant de s'entraîner. Il semble cependant que peu de personnes aient trouvé le temps de s'entraîner avant l'atelier.

Durant l'atelier, les participants avaient tous à disposition une documentation papier contenant des images de coupes 2D (humain et brebis) permettant de localiser les différentes structures anatomiques qu'ils devaient segmenter durant les expériences.

8.2.1.2 Expériences réalisées

La version du logiciel utilisée exploitait les atlas probabilistes locaux selon le jeu de paramètres décrit précédemment dans la section 7.1.3. La correction a posteriori n'était pas activée dans cette version.

1. http://www.rfai.li.univ-tours.fr/PublicData/3D_Brain_Seg/home.html

Trois types de situations ont été proposés :

- Segmentation de l'image sans avoir à positionner les boîtes englobantes. Seul l'ordre de la segmentation était modifiable par l'utilisateur
- Segmentation de l'image avec positionnement manuel des boîtes englobantes mais sans avoir le choix de l'ordre d'extraction des régions
- Segmentation de l'image de manière libre. L'utilisateur pouvait alors choisir l'ordre de la segmentation, utiliser le positionnement automatique ou non, corriger ou non les positionnements proposés

8.2.2 Retours d'expériences

Lors de ces expériences plusieurs propriétés et caractéristiques de notre méthode ont pu être étudiées. Plusieurs de ces résultats ont déjà été présentés dans les chapitres précédents. Il s'agit ici plutôt de faire une synthèse générale sur les modes d'utilisation du logiciel, les difficultés rencontrées et les points positifs identifiés.

8.2.2.1 Notre analyse globale

Les positionnements manuels réalisés par les utilisateurs ont été de plutôt bonne qualité. En général, les bords des régions avec des frontières assez nettes sont très bien positionnés par l'utilisateur. Par contre, le positionnement automatique semble plus adapté lorsque les régions sont petites et/ou possèdent des frontières plutôt floues.

La qualité générale de la segmentation dépend beaucoup de l'utilisateur. Les utilisateurs experts en imagerie ont très souvent obtenu des résultats de bonne qualité parfois même meilleurs que ceux obtenus lorsque les régions sont idéalement positionnées. Par contre, lorsque l'utilisateur n'est pas expert, il est possible d'obtenir de très mauvaises segmentations. Certains utilisateurs, non experts en anatomie, ont par exemple cherché à corriger les positionnements automatiques fournis par le logiciel. Cependant leur niveau d'expertise ne permettait pas d'améliorer ce positionnement.

Le choix de l'ordre de segmentation a semblé plus difficile à appréhender par les utilisateurs. Par habitude, les participants ont eu tendance à commencer par des régions visibles mais pas forcément stables (comme les ventricules). De plus, les utilisateurs ont souvent segmenté les régions deux par deux (les régions bilatérales) au lieu d'enchaîner des régions proches d'un même hémisphère. Dans ces deux situations, les relations spatiales qui en découlent ne sont pas toujours les meilleurs possibles.

Le temps de segmentation moyen d'une image pour les bases de cerveaux utilisées est d'environ 1 à 2h alors qu'il est d'environ 11 minutes (sur des PC actuels) si toutes les régions sont segmentées automatiquement sans interaction.

8.2.2.2 Retours utilisateurs

À la suite de l'expérimentation, un questionnaire a été proposé aux participants afin de leur permettre de donner leurs avis et suggestions à propos de l'outil de segmentation.

Douze réponses ont été obtenues dont 4 personnes ayant participé à toutes les expérimentations organisées durant les 2 journées. Les questions posées aux utilisateurs portaient à la fois sur la qualité des segmentations obtenues, le temps de segmentation, la qualité de la visualisation et l'ergonomie de l'outil de segmentation 3DBrainSeg.

Qualité des segmentations obtenues

L'ensemble des utilisateurs a jugé la qualité des segmentations bonne ou très bonne lors de l'utilisation libre du logiciel. 83 % des utilisateurs ont jugé que le positionnement automatique était de bonne ou très bonne qualité. Un tiers des utilisateurs ont jugé "mauvaises" les segmentations lors de l'utilisation du placement automatique des régions, les autres ont jugé la segmentation au moins "bonne".

Le positionnement des régions semble fonctionner correctement dans la plupart des cas. Cependant, d'après certains utilisateurs, les bulbes olfactifs ont posé un problème pour les relations spatiales apprises.

Les utilisateurs ont fait remarquer qu'il n'était pas facile d'évaluer la qualité d'une segmentation avec le système de visualisation actuel.

Certains utilisateurs ont proposé de mettre en place un outil de type pinceau afin de corriger manuellement les segmentations obtenues.

Temps et résultats

L'ensemble des utilisateurs ont estimé que la segmentation d'une région était un processus rapide. Le temps d'exécution a semblé "court" ou "très court" pour 66.6 % des utilisateurs et "long" ou "très long" pour 33.3 %.

Le temps de prise en main du logiciel a été jugé positivement. Certains biologistes ont par exemple pris un temps important pour segmenter la première image mais ont ensuite pu manipuler beaucoup plus rapidement le logiciel lors des segmentations suivantes.

Qualité de la visualisation

La qualité des visualisations a été la lacune la plus souvent mentionnée pour notre logiciel : 58.3 % ont trouvé la qualité de la visualisation bonne et 41.7 % mauvaise ou très mauvaise.

Aucune remarque négative n'a été faite sur la visualisation globale de l'image à travers les coupes 2D ou la scène 3D. Par contre, l'absence d'*overlay* pour visualiser à la fois l'image réelle et la segmentation sur les coupes 2D a été un vrai problème pour plusieurs utilisateurs. La plupart des suggestions provenant des utilisateurs sont liées à ce problème (possibilité de visualiser simultanément la segmentation et l'image réelle sur une coupe 2D par l'intermédiaire de contours ou d'*overlay*).

Les autres remarques sont aussi liées aux coupes 2D. En effet, la modification du contraste et du zoom permettrait de mettre en évidence plus facilement où positionner un bord de région. Enfin, les biologistes aimeraient connaître l'information du numéro des coupes.

Qualité des interactions

La qualité des interactions doit aussi être améliorée d'après les utilisateurs, principalement le positionnement de la boîte englobante des régions : 58.3 % pense que le mode de positionnement actuel devrait être amélioré. 42.7 % ont cependant trouvé le positionnement facile à réaliser.

Nous étions conscients que le mode de positionnement de la boîte englobante, nécessitant de placer les bords dans un ordre précis, serait gênant surtout lors de la prise en main du logiciel. De plus, actuellement, si l'utilisateur n'est pas satisfait du positionnement d'un bord, il n'a pas d'autres solutions que de repositionner tous les bords de la boîte englobante. Il s'agit ici d'un dispositif d'interaction complexe que nous n'avons pas pris le temps de mettre en place.

Les utilisateurs proposent d'améliorer le mode de positionnement de la boîte englobante en permettant de déplacer un bord déjà placé indépendamment des autres mais aussi en déplaçant la boîte englobante tout entière. Il semble assez fréquent qu'une boîte englobante soit de taille correcte mais mal positionnée.

8.2.3 Bilan

Cette expérience utilisateurs a d'abord joué le rôle de preuve de concept et ainsi mettre en évidence que l'outil pouvait être utilisé par des utilisateurs non-experts en informatique ou en traitement d'images (biologistes).

Cet atelier a donc eu des retours très positifs en terme de compréhension et d'approbation de la méthode proposée (approche locale et interactive) qui laissait dubitatif de nombreux collègues avant que cet atelier ne soit organisé. De nombreuses personnes ont été convaincues par l'approche proposée pendant et grâce à cet atelier. Il est important de noter qu'il devient rare de voir passer aussi rapidement un prototype de laboratoire en un outil utilisable par un panel varié d'utilisateurs.

L'expertise liée au type d'image à segmenter se révèle être beaucoup plus importante : les spécialistes en anatomie du cerveau ont ainsi obtenu de meilleurs résultats que les spécialistes en informatique (en particulier pour le cerveau de brebis).

Ces expériences et ces retours nous ont permis de connaître les points faibles du logiciel et de l'IHM en particulier. Même si nous avons conscience d'une partie de ces problèmes, ces suggestions nous ont permis de formaliser/prioriser les différentes améliorations à prévoir en priorité. Ainsi, le logiciel est actuellement en reconstruction (par un prestataire) afin d'harmoniser les bibliothèques et technologies utilisées, d'améliorer la modélisation et de rendre le processus général de segmentation plus efficace. Une interface web plus efficace et corrigeant les lacunes identifiées devrait aussi être mise en place et s'interfacer avec notre outil de segmentation afin qu'il puisse être utilisé en ligne.

Cinquième partie

Conclusion

Chapitre 9

Conclusion

9.1 Bilan

Au travers de ces travaux, nous proposons une nouvelle méthode de segmentation interactive et incrémentale d'images 3D médicales. Pour cela, nous proposons une nouvelle manière, plus locale, de modéliser les connaissances *a priori* décrivant les caractéristiques anatomiques de chacune des structures des organes à segmenter. Cette modélisation s'inspire des méthodes se basant sur des atlas pour guider la segmentation mais propose de l'instancier de manière plus locale en permettant la création ou l'apprentissage incrémental d'atlas locaux décrivant chacun uniquement une des structures d'intérêt dans un organe donné. Les atlas sont définis localement sur une zone autour de chaque région. Ces atlas peuvent aussi bien être probabilistes et composés d'un template et d'une carte de probabilité, que de type multi-atlas et composés d'un certain nombre de couples d'images réelles et d'images labellisées. Contrairement au cas classique pour lequel chaque région est définie à l'aide d'un seul template (espace de référence) représentant l'organe entier, nos atlas locaux possèdent tous leur propre espace de référence. Les informations provenant de chaque région sont donc séparées (indépendantes), il est aussi possible d'imaginer utiliser des modalités différentes pour apprendre chaque région.

Nous proposons d'exploiter un graphe attribué pour lier les structures anatomiques (représentées par des atlas locaux) les unes avec les autres. Les nœuds du graphe correspondent alors aux structures (régions) à segmenter et les arcs représentent les relations spatiales existantes entre ces régions. Les relations spatiales ont pour objectif de positionner les atlas locaux au sein de l'image entière, les atlas locaux n'étant définis que sur une partie de l'image. Les relations spatiales apprises et stockées par l'intermédiaire des arcs du graphe sont des informations de distances entre les bords de la boîte englobante associée à chaque région. Ces distances sont stockées sous forme d'intervalles de distance relative. Elles peuvent être apprises entre chacune des régions du graphe si la base d'apprentissage le permet. Nous appelons cette représentation "graphe des connaissances *a priori*". Il peut être appris à partir de très peu d'images et représente la scène à analyser. Il renferme les connaissances *a priori* associées à chaque région (ou sous-ensemble de régions) de manière indépendante afin de pouvoir être utilisées facilement et individuellement. Cette modélisation est générique et peut être appliquée à différents organes ou type d'images médicales

3D. Lors de la segmentation, ce graphe des connaissances *a priori* est utilisé pour obtenir une labellisation de l'image. Ainsi, la segmentation est réalisée suivant un processus incrémental et il devient possible de réaliser des segmentations partielles, rapides en faisant varier ou en choisissant l'ordre de segmentation des différentes régions. Outre le choix de l'ordre d'extraction des régions, l'utilisateur peut intervenir avant et après chaque positionnement, segmentation locale afin de corriger, de manière interactive, le positionnement et la qualité de la segmentation obtenues automatiquement. Pour obtenir les segmentations locales, nous avons proposé une classification des voxels par champ de Markov dans le cas des atlas locaux probabilistes et une méthode de vote classique dans le cas des multi-atlas locaux.

Enfin, notre méthode locale englobe un post-traitement de correction des erreurs systématiques qu'une méthode de segmentation peut produire et d'optimisation de l'adjacence entre régions. Cette post-correction se base sur les travaux décrits dans [98] mais nous avons proposé d'utiliser des informations probabilistes en entrée des classifieurs plutôt que la segmentation en labels. Nous avons aussi proposé l'utilisation de classifieurs de fond pour chaque région. Nous profitons aussi de ce post-traitement pour corriger certains problèmes d'adjacence entre les régions liés à notre méthode incrémentale.

De nombreuses expérimentations ont permis de mettre en avant les avantages et les inconvénients de notre méthode mais aussi de sélectionner les paramètres les plus adéquats.

L'utilisation d'atlas locaux probabilistes et d'un champ de Markov semble avoir plus d'avantage que l'utilisation des multi-atlas locaux couplée à une méthode de vote. La qualité de la segmentation obtenue est similaire mais cependant les multi-atlas locaux s'avèrent être plus coûteux en temps. La méthode étant interactive, il est important de limiter les temps de calcul afin de préserver l'utilisabilité de la méthode (interactivité).

D'autres expérimentations ont aussi pu démontrer qu'un choix de l'ordre de segmentation basé sur la cohérence des relations spatiales apprises semble le plus pertinent. La variabilité des résultats diminue avec cet ordonnancement par rapport à un ordre aléatoire ou aux ordres choisis par les utilisateurs. L'ordre de segmentation a un impact plus important lorsque l'utilisateur n'intervient pas ou peu dans la segmentation. Si celui-ci agit de manière importante par l'intermédiaire de repositionnement manuel des boîtes englobantes alors l'influence de l'ordre de segmentation devient plus faible.

Le nombre de structures anatomiques à positionner manuellement pour initialiser le processus de segmentation incrémental afin d'obtenir une segmentation complète correcte dépend beaucoup du type d'images. Nous avons néanmoins montré que ce nombre reste tout à fait raisonnable lorsqu'il s'agit de segmenter les régions sous-corticales du cerveau humain. Seules deux ou trois structures positionnées convenablement suffisent pour déclencher ensuite le positionnement automatique pour les régions suivantes.

Enfin la segmentation des régions corticales du cerveau a mis en évidence un contexte d'application pour lequel la post-correction a un impact important sur les résultats obtenus. Lorsque les régions à segmenter s'entremêlent, les problèmes liés à l'adjacence entre les régions deviennent importants. En effet, notre approche (locale sans post-correction) ne tient pas compte et ne modifie pas les segmentations précédentes dans la séquence en cours. Ainsi, il arrive fréquemment que des voxels non labellisés se retrouvent présents à la frontière entre deux régions alors qu'il semble évident que ces voxels appartiennent à l'une

CONCLUSION

des deux régions adjacentes. La post-correction proposée corrige en partie ces problèmes et par conséquent améliore la qualité de la segmentation.

Les atlas locaux se sont montrés plus efficaces que les atlas globaux dans le cas où la méthode de segmentation est la même et que les atlas sont bien positionnés. Le caractère incrémental de la segmentation permet d'obtenir des segmentations partielles rapides des images. Cette propriété permet à l'approche proposée de fournir des segmentations plus rapidement que les méthodes de l'état de l'art. Nous avons montré que le temps de segmentation d'une région d'un cerveau était inférieur à une minute alors que les méthodes de segmentation complète du cerveau nécessitent plusieurs heures de calcul pour fournir la labellisation des voxels.

L'intérêt de l'aspect interactif de la méthode a aussi pu être mis en avant. Nous avons montré, d'une part, que lorsque l'utilisateur est un expert et qu'il connaît bien les images manipulées, un repositionnement manuel permet d'obtenir des résultats de qualité similaire à un positionnement "parfait" (provenant de la vérité-terrain). D'autres part, nous avons aussi mis en évidence le fait que l'utilisateur peut, par l'intermédiaire de repositionnements successifs, corriger certaines erreurs. Si un positionnement et le recalage qui en découle convergent vers une solution biaisée, l'utilisateur a la possibilité d'annuler cette segmentation et de repositionner l'atlas de manière à essayer de compenser l'erreur précédente. L'atelier utilisateur a cependant mis en avant que le positionnement manuel n'était pas toujours de très bonne qualité lorsque des régions peu visibles sont recherchées et que l'utilisateur n'est pas un véritable expert. Dans ces situations, il est courant que le positionnement automatique donne des résultats de meilleure qualité que le positionnement manuel.

Les expérimentations ont aussi démontré que la quantité de données d'apprentissage nécessaire pour obtenir une segmentation de bonne qualité n'est pas élevée. Dans le cas des images de cerveaux humains, nous avons montré que seul trois ou quatre images suffisent pour obtenir une segmentation de bonne qualité. De même, nous avons pu voir que, dans le cas des images de brebis ex vivo pour lequel le nombre d'images d'apprentissage disponibles est assez faible, les résultats de segmentation obtenus avec notre approche sont très proches de ceux obtenus par la méthode JLF.

La qualité de la segmentation produite par notre approche est supérieure à celle fournie par FreeSurfer [36] mais reste inférieure à celle fournie par la méthode JLF [99]. La différence sur le cerveau humain est d'environ 4% mais diminue à 1.3% pour le cerveau de la brebis. La segmentation d'une partie des régions avec notre approche reste cependant plus rapide et notre méthode de modélisation locale des connaissances a priori n'est pas spécifique à un problème particulier. Les expérimentations ont montré que cette approche locale pouvait être adaptée très facilement et rapidement à un large type de situations : Notre approche reste opérationnelle et efficace aussi bien dans le cas de segmentation de régions sous-corticales que de régions corticales de cerveaux humains et de cerveaux de brebis. Nous avons aussi montré qu'il était possible de segmenter des images provenant de modalités différentes avec les images de cerveaux de brebis in-vivo. Enfin, des images de cœurs acquises selon différentes modalités d'imagerie ont aussi pu être traitées à l'aide de notre outil (participation au concours MICCAI'2017 [57]). Cette capacité de généralisation est liée à l'utilisation des atlas locaux mais aussi au mode d'apprentissage et d'exploitation des

relations spatiales. L'utilisation de distance relative permet de rendre ces informations indépendantes du type d'images. La segmentation des images in vivo basse résolution à partir des informations apprises sur les images ex vivo haute résolution en est un exemple.

Ces travaux ont débouché sur un prototype opérationnel qui a pu être testé en « grandeur nature », lors d'un atelier organisé dans le cadre du projet NeuroGeo. Cet atelier est une preuve de concept quant à l'utilisabilité du logiciel dans un contexte de segmentation d'images médicales 3D. Les utilisateurs experts ont réussi à obtenir des segmentations correctes à la fois d'images de cerveaux humains mais aussi de brebis. Même si les temps d'interaction nécessaires restent assez élevés comparé au temps de calcul, ceux-ci diminuent assez rapidement au fil des expériences. Cet atelier nous a aussi permis d'obtenir des suggestions en ce qui concerne les différentes voies d'amélioration possible des IHM. La mise en place de ces suggestions devrait pouvoir rendre complètement opérationnel l'outil de segmentation pour les biologistes et ainsi leur faire gagner un temps considérable lors de leur travaux d'analyse d'images médicales 3D.

9.2 Perspectives

La première piste d'amélioration de ces travaux qui peut être envisagée est la modification de la méthode de classification de voxels utilisée localement. Le champ de Markov (avec les atlas probabilistes) et la méthode de vote (avec les multi-atlas) utilisés sont peut-être des éléments limitant la qualité de segmentation. De meilleures méthodes de fusion de données existent aujourd'hui pour la segmentation multi-atlas qui pourraient être implémentées dans le cas des multi-atlas locaux. Le temps d'exécution sera plus important mais la qualité de la segmentation devrait être augmentée. Le deep learning montre son efficacité sur un large type d'images médicales avec, par exemple, l'utilisation de réseaux de type U-Net, on peut aussi imaginer l'utiliser dans le cadre de notre classification locale. Des réseaux pourraient être appris et appliqués localement. Il serait alors intéressant d'observer si l'efficacité de réseaux locaux peut être meilleure que l'efficacité d'un réseau global.

Le recalage des atlas locaux inclus dans notre méthode est indépendant de la structure anatomique à segmenter. Le type de recalage ainsi que les paramètres restent toujours les mêmes. Pourtant certaines structures sont très différentes, en particulier en terme de taille mais aussi d'intensité et de contraste. Il pourra être intéressant d'utiliser un recalage adapté à chaque région. Mettre en place une adaptation des paramètres en fonction de la taille de la région ne paraît pas difficile : dans le cas du recalage B-spline, le caractère pyramidal du recalage ainsi que le nombre de nœuds par dimension pourrait facilement être ajusté en fonction de la taille de la région. Il pourrait aussi être intéressant d'apprendre sur la base d'apprentissage quel recalage est le plus adapté à chaque région. La transformation ou la métrique optimale à utiliser ne sont peut-être pas les mêmes pour chaque région.

Nous aurions également aimé pouvoir conduire certaines expérimentations supplémentaires, en particulier sur les questions d'apprentissage incrémental ; Il serait intéressant de mettre plus en évidence la capacité de notre approche à pouvoir construire le graphe de connaissances *a priori* de manière incrémentale ainsi que de pouvoir apprendre en continu à partir d'images segmentées par la méthode elle-même.

Dans un futur proche, l'outil de segmentation 3DBrainSeg va être utilisé par des bio-

logistes sur des images de cerveaux de brebis. Quelques segmentations manuelles sont déjà disponibles et ont permis de construire les connaissances a priori liées aux régions corticales recherchées dans le cerveau des brebis. De plus, de nouvelles images ont été acquises sur des brebis d'âge différents. 3DBrainSeg sera exploité pour évaluer l'évolution anatomique des cerveaux. Les connaissances a priori apprises sur le cerveau de la brebis adulte seront utilisées pour analyser les cerveaux de brebis "jeunes". Le caractère local et interactif de la méthode devrait permettre d'obtenir des résultats satisfaisants.

Enfin, les recommandations liées à l'amélioration de l'IHM de 3DBrainSeg devront être prises en compte, en particulier, les remarques liées à la visualisation des résultats ou des positionnements des bords des régions. Les coupes 2D devront permettre de visualiser les résultats de segmentation en *overlay*. En ce qui concerne les interactions, le positionnement de la boîte englobante pourra être rendu plus ergonomique. Les bords de la boîte doivent pouvoir être déplacés à l'aide d'interactions de type *drag and drop*. Enfin la possibilité d'une correction manuelle par l'utilisateur (de type pinceau ou baguette magique) au sein du logiciel pourrait être véritablement intéressante si l'on souhaite que l'outil soit efficacement utilisé par les biologistes. Cet ajout permettra à l'utilisateur d'obtenir, dans tous les cas, la segmentation qu'il désire et sera réellement complémentaire de notre méthode de segmentation. Une prestation externe est actuellement en cours pour réaliser une première partie de ces évolutions souhaitées.

Pour conclure, notons que la région Centre-Val-de-Loire, ainsi que les partenaires impliqués dans le projet NeuroGeo, ont reconnu l'intérêt des résultats obtenus durant ce travail de thèse. Un nouveau financement de 3 ans a été obtenu (projet régional Neuro2Co) visant, entre autres, à poursuivre le développement et l'exploitation de notre méthode et de l'outil 3DBrainSeg toujours avec l'objectif de pouvoir segmenter efficacement des cerveaux provenant de diverses espèces (partenariat avec le Zoo-parc de Beauval) et de manière interactive et intuitive (partenariat avec des collèges de la région).

Bibliographie

- [1] Brainweb simulated brain database. <http://www.bic.mni.mcgill.ca/brainweb/>, 2017. [Online ; accessed 20-December-2014].
- [2] http://casemed.case.edu/clerkships/neurology/web_neurorad/mri_basics.htm, 2017. [Online ; accessed 20-December-2014].
- [3] <http://recap-ide.blogspot.fr/2014/10/anatomie-du-systeme-nerveux.html>, 2017. [Online ; accessed 20-December-2014].
- [4] <https://fr.pinterest.com/explore/noyau-caudé>, 2017. [Online ; accessed 20-December-2014].
- [5] <https://northfieldneuroscience.wordpress.com>, 2017. [Online ; accessed 20-December-2014].
- [6] <http://www.cantechletter.com/2016/07/study-finds-software-glitch-40000-fmri-results/>, 2017. [Online ; accessed 20-December-2014].
- [7] Mm-whs17, 2017. [Online ; accessed 20-December-2014].
- [8] P. Aljabar, R. Heckemann, A. Hammers, J. V. Hajnal, and D. Rueckert. *Classifier Selection Strategies for Label Fusion Using Large Atlas Databases*, pages 523–531. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.
- [9] K. S. Arun, T. S. Huang, and S. D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(5) :698–700, Sept 1987.
- [10] J. Ashburner and K. J. Friston. Unified segmentation. *NeuroImage*, 26(3) :839 – 851, 2005.
- [11] B. Avants, C. Epstein, M. Grossman, and J. Gee. Symmetric diffeomorphic image registration with cross-correlation : Evaluating automated labeling of elderly and neurodegenerative brain. *Medical Image Analysis*, 12(1) :26 – 41, 2008.
- [12] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society*, 48(3) :48–259, 1986.
- [13] F. Bloch, W. W. Hansen, and M. Packard. Nuclear induction. *Phys. Rev.*, 69 :127–127, Feb 1946.
- [14] I. Bloch. Some aspects of dempster-shafer evidence theory for classification of multi-modality medical images taking partial volume effect into account. *Pattern Recognition Letters*, 17(8) :905 – 919, 1996.

- [15] I. Bloch. Fuzzy spatial relationships for image processing and interpretation : a review. *Image and Vision Computing*, 23(2) :89 – 110, 2005.
- [16] F. L. Bookstein. *Thin-Plate splines and the atlas problem for biomedical images*, pages 326–342. Springer Berlin Heidelberg, Berlin, Heidelberg, 1991.
- [17] Y. Boykov and M.-P. Jolly. *Interactive Organ Segmentation Using Graph Cuts*, pages 276–286. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000.
- [18] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9) :1124–1137, Sept 2004.
- [19] S. Bricq, C. Collet, and J. P. Armspach. Triplet markov chain for 3d mri brain segmentation using a probabilistic atlas. In *3rd IEEE International Symposium on Biomedical Imaging : Nano to Macro, 2006.*, pages 386–389, April 2006.
- [20] C. Broit. Aoptimal registration of deformed images. 1981.
- [21] A. Buades, B. Coll, and J. M. Morel. A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 60–65 vol. 2, June 2005.
- [22] M. Cabezas, A. Oliver, X. Lladó, J. Freixenet, and M. B. Cuadra. A review of atlas-based segmentation for magnetic resonance brain images. *Computer Methods and Programs in Biomedicine*, 104(3) :e158 – e177, 2011.
- [23] B. Caldaïrou, N. Passat, P. A. Habas, C. Studholme, and F. Rousseau. A non-local fuzzy segmentation method : Application to brain mri. *Pattern Recognition*, 44(9) :1916 – 1927, 2011.
- [24] H. Chen, X. Qi, L. Yu, Q. Dou, J. Qin, and P.-A. Heng. Dcan : Deep contour-aware networks for object instance segmentation from histology images. *Medical Image Analysis*, 36(Supplement C) :135 – 146, 2017.
- [25] K.-S. Chuang, H.-L. Tzeng, S. Chen, J. Wu, and T.-J. Chen. Fuzzy c-means clustering with spatial information for image segmentation. *Computerized Medical Imaging and Graphics*, 30(1) :9 – 15, 2006.
- [26] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger. *3D U-Net : Learning Dense Volumetric Segmentation from Sparse Annotation*, pages 424–432. Springer International Publishing, Cham, 2016.
- [27] D. L. Collins, A. P. Zijdenbos, V. Kollokian, J. G. Sled, N. J. Kabani, C. J. Holmes, and A. C. Evans. Design and construction of a realistic digital brain phantom. *IEEE Transactions on Medical Imaging*, 17(3) :463–468, June 1998.
- [28] O. Colliot. *Représentation, évaluation et utilisation de relations spatiales pour l'interprétation d'images. Application à la reconnaissance de structures anatomiques en imagerie médicale*. PhD thesis, Télécom ParisTech, 2003.
- [29] O. Colliot, O. Camara, and I. Bloch. Integration of fuzzy spatial relations in deformable models-application to brain mri segmentation. *Pattern Recognition*, 39(8) :1401 – 1414, 2006.
- [30] R. Damadian. Tumor detection by nuclear magnetic resonance. *Science*, 171(3976) :1151–1153, 1971.

- [31] A. de Brébisson and G. Montana. Deep neural networks for anatomical brain segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 20–28, June 2015.
- [32] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, B*, 39(1) :1–38, 1977.
- [33] A. Ella and M. Keller. Construction of an mri 3d high resolution sheep brain template. *Magnetic Resonance Imaging*, 33(10) :1329 – 1337, 2015.
- [34] D. C. V. Essen. A population-average, landmark- and surface-based (pals) atlas of human cerebral cortex. *NeuroImage*, 28(3) :635 – 662, 2005.
- [35] A. X. Falcão and J. K. Udupa. A 3d generalization of user-steered live-wire segmentation. *Medical Image Analysis*, 4(4) :389 – 402, 2000.
- [36] B. Fischl. Freesurfer. *NeuroImage*, 62(2) :774 – 781, 2012.
- [37] B. Fischl, M. I. Sereno, and A. M. Dale. Cortical surface-based analysis. *NeuroImage*, 9(2) :195 – 207, 1999.
- [38] V. Fonov, A. C. Evans, K. Botteron, C. R. Almli, R. C. McKinstry, and D. L. Collins. Unbiased average age-appropriate atlases for pediatric studies. *NeuroImage*, 54(1) :313 – 327, 2011.
- [39] E. W. FORGY. Cluster analysis of multivariate data : efficiency vs. interpretability of classifications. *Biometrics*, 21 :768–769, 1965.
- [40] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1) :119 – 139, 1997.
- [41] H. Greenspan, A. Ruf, and J. Goldberger. Constrained gaussian mixture model framework for automatic segmentation of mr brain images. *IEEE Transactions on Medical Imaging*, 25(9) :1233–1245, Sept 2006.
- [42] P. A. Habas, K. Kim, J. M. Corbett-Detig, F. Rousseau, O. A. Glenn, A. J. Barkovich, and C. Studholme. A spatiotemporal atlas of mr intensity, tissue probability and shape of the fetal brain with application to segmentation. *NeuroImage*, 53(2) :460 – 470, 2010.
- [43] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A. Courville, Y. Bengio, C. Pal, P.-M. Jodoin, and H. Larochelle. Brain tumor segmentation with deep neural networks. *Medical Image Analysis*, 35 :18 – 31, 2017.
- [44] R. He, S. Datta, B. R. Sajja, and P. A. Narayana. Generalized fuzzy clustering for segmentation of multi-spectral magnetic resonance images. *Computerized Medical Imaging and Graphics*, 32(5) :353 – 366, 2008.
- [45] G. N. Hounsfield. Computerized transverse axial scanning (tomography) : Part 1. description of system. *The British Journal of Radiology*, 46(552) :1016–1022, 1973.
- [46] J. E. Iglesias and M. R. Sabuncu. Multi-atlas segmentation of biomedical images : A survey. *Medical Image Analysis*, 24(1) :205 – 219, 2015.
- [47] J. E. Iglesias, M. R. Sabuncu, and K. Van Leemput. *A Probabilistic, Non-parametric Framework for Inter-modality Label Fusion*, pages 576–583. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

- [48] I. Isgum, M. Staring, A. Rutten, M. Prokop, M. A. Viergever, and B. van Ginneken. Multi-atlas-based segmentation with local decision fusion ; application to cardiac and aortic segmentation in ct scans. *IEEE Transactions on Medical Imaging*, 28(7) :1000–1010, July 2009.
- [49] E. Ising. Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik A Hadrons and Nuclei*, 31(1) :253–258, Feb. 1925.
- [50] H. Jia, P.-T. Yap, and D. Shen. Iterative multi-atlas-based multi-image segmentation with tree-based registration. *NeuroImage*, 59(1) :422 – 430, 2012.
- [51] Y. Kang, K. Engelke, and W. A. Kalender. Interactive 3d editing tools for image segmentation. *Medical Image Analysis*, 8(1) :35 – 46, 2004.
- [52] R. Kikinis, M. E. Shenton, D. V. Iosifescu, R. W. McCarley, P. Saiviroonporn, H. H. Hokama, A. Robatino, D. Metcalf, C. G. Wible, C. M. Portas, R. M. Donnino, and F. A. Jolesz. A digital brain atlas for surgical planning, model-driven segmentation, and teaching. *IEEE Transactions on Visualization and Computer Graphics*, 2(3) :232–241, Sep 1996.
- [53] R. Kindermann and J. L. Snell. *Markov Random Fields and Their Applications*. AMS, 1980.
- [54] S. Klein, M. Staring, and J. P. W. Pluim. Evaluation of optimization methods for nonrigid medical image registration using mutual information and b-splines. *IEEE Transactions on Image Processing*, 16(12) :2879–2890, Dec 2007.
- [55] M. Kuklisova-Murgasova, P. Aljabar, L. Srinivasan, S. J. Counsell, V. Doria, A. Serag, I. S. Gousias, J. P. Boardman, M. A. Rutherford, A. D. Edwards, J. V. Hajnal, and D. Rueckert. A dynamic 4d probabilistic atlas of the developing brain. *NeuroImage*, 54(4) :2750 – 2763, 2011.
- [56] M. Kuklisova-Murgasova, P. Aljabar, L. Srinivasan, S. J. Counsell, V. Doria, A. Serag, I. S. Gousias, J. P. Boardman, M. A. Rutherford, A. D. Edwards, J. V. Hajnal, and D. Rueckert. A dynamic 4d probabilistic atlas of the developing brain. *NeuroImage*, 54(4) :2750 – 2763, 2011.
- [57] B. A. Landman, S. K. Warfield, A. Hammers, A. Akhondi-asl, A. J. Asman, A. Ribbens, B. Lucas, B. B. Avants, C. Ledig, D. Ma, D. Rueckert, D. Vandermeulen, F. Maes, H. Holmes, H. Wang, J. Wang, J. Doshi, J. Kornegay, J. V. Hajnal, K. Gray, L. Collins, M. J. Cardoso, M. Lythgoe, M. Styner, M. Armand, M. Miller, P. Aljabar, P. Suetens, P. A. Yushkevich, P. Coupe, R. Wolz, and R. A. Heckemann. MICCAI 2012 Workshop on Multi-Atlas Labeling. 2012.
- [58] M. Lorenzo-Valdés, G. I. Sanchez-Ortiz, A. G. Elkington, R. H. Mohiaddin, and D. Rueckert. Segmentation of 4d cardiac mr images using a probabilistic atlas and the em algorithm. *Medical Image Analysis*, 8(3) :255 – 265, 2004.
- [59] J. M. Lötjönen, R. Wolz, J. R. Koikkalainen, L. Thurfjell, G. Waldemar, H. Soininen, and D. Rueckert. Fast and robust multi-atlas segmentation of brain magnetic resonance images. *NeuroImage*, 49(3) :2352 – 2365, 2010.
- [60] Y. Ma, P. Hof, S. Grant, S. Blackband, R. Bennett, L. Slatest, M. McGuigan, and H. Benveniste. A three-dimensional digital atlas database of the adult c57bl/6j mouse brain by magnetic resonance microscopy. *Neuroscience*, 135(4) :1203 – 1215, 2005.

- [61] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens. Multimodality image registration by maximization of mutual information. *IEEE Transactions on Medical Imaging*, 16(2) :187–198, April 1997.
- [62] D. S. Marcus, T. H. Wang, J. Parker, J. G. Csernansky, J. C. Morris, and R. L. Buckner. Open access series of imaging studies (oasis) : Cross-sectional mri data in young, middle aged, nondemented, and demented older adults. *J. Cognitive Neuroscience*, 19(9) :1498–1507, Sept. 2007.
- [63] P. Matsakis and L. Wendling. A new way to represent the relative position between areal objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(7) :634–643, Jul 1999.
- [64] J. Mazziotta, A. Toga, A. Evans, P. Fox, J. Lancaster, K. Zilles, R. Woods, T. Paus, G. Simpson, B. Pike, C. Holmes, L. Collins, P. Thompson, D. MacDonald, M. Iacoboni, T. Schormann, K. Amunts, N. Palomero-Gallagher, S. Geyer, L. Parsons, K. Narr, N. Kabani, G. L. Goualher, D. Boomsma, T. Cannon, R. Kawashima, and B. Mazoyer. A probabilistic atlas and reference system for the human brain : International consortium for brain mapping (icbm). *Philosophical Transactions of the Royal Society of London B : Biological Sciences*, 356(1412) :1293–1322, 2001.
- [65] D. G. McLaren, K. J. Kosmatka, T. R. Oakes, C. D. Kroenke, S. G. Kohama, J. A. Matochik, D. K. Ingram, and S. C. Johnson. A population-average mri-based atlas collection of the rhesus macaque. *NeuroImage*, 45(1) :52 – 59, 2009.
- [66] O. Nempont, J. Atif, and I. Bloch. A constraint propagation approach to structural model based image segmentation and recognition. *Information Sciences*, 246 :1 – 27, 2013.
- [67] B. Nitzsche, S. Frey, L. D. Collins, J. Seeger, D. Lobsien, A. Dreyer, H. Kirsten, M. H. Stoffel, V. S. Fonov, and J. Boltze. A stereotaxic, population-averaged t1w ovine brain atlas including cerebral morphology and tissue volumes. May 2015.
- [68] L. G. Nyul, J. K. Udupa, and X. Zhang. New variants of a method of mri scale standardization. *IEEE Transactions on Medical Imaging*, 19(2) :143–150, Feb 2000.
- [69] T. Okada, R. Shimada, M. Hori, M. Nakamoto, Y.-W. Chen, H. Nakamura, and Y. Sato. Automated segmentation of the liver from 3d ct images using probabilistic atlas and multilevel statistical shape model. *Academic Radiology*, 15(11) :1390 – 1403, 2008.
- [70] D. L. Pham and J. L. Prince. *An Adaptive Fuzzy Segmentation Algorithm for Three-Dimensional Magnetic Resonance Images*, pages 140–153. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999.
- [71] A. Pitiot, E. Bardinet, P. Thompson, and G. Malandain. Piecewise Affine Registration of Biological Images for Volume Reconstruction. *Medical Image Analysis*, 10(3) :465–483, 2006.
- [72] K. M. Pohl, J. Fisher, W. E. L. Grimson, R. Kikinis, and W. M. Wells. A bayesian model for joint segmentation and registration. *NeuroImage*, 31(1) :228 – 239, 2006.
- [73] K. M. Pohl, J. Fisher, M. Shenton, R. W. McCarley, W. E. L. Grimson, R. Kikinis, and W. M. Wells. *Logarithm Odds Maps for Shape Representation*, pages 955–963. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

- [74] M. Poon, G. Hamarneh, and R. Abugharbieh. Efficient interactive 3d livewire segmentation of complex objects with arbitrary topology. *Computerized Medical Imaging and Graphics*, 32(8) :639 – 650, 2008.
- [75] E. M. Purcell, H. C. Torrey, and R. V. Pound. Resonance absorption by nuclear magnetic moments in a solid. *Phys. Rev.*, 69 :37–38, Jan 1946.
- [76] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2) :257–286, Feb 1989.
- [77] A. Roche, G. Malandain, X. Pennec, and N. Ayache. *The correlation ratio as a new similarity measure for multimodal image registration*, pages 1115–1124. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.
- [78] O. Ronneberger, P. Fischer, and T. Brox. *U-Net : Convolutional Networks for Biomedical Image Segmentation*, pages 234–241. Springer International Publishing, Cham, 2015.
- [79] R. K. Roy. A primer on the taguchi method, 1990.
- [80] D. Rueckert and J. A. Schnabel. *Medical Image Registration*, pages 131–154. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
- [81] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes. Nonrigid registration using free-form deformations : application to breast mr images. *IEEE Transactions on Medical Imaging*, 18(8) :712–721, Aug 1999.
- [82] Z. S. Saad, R. C. Reynolds, B. Argall, S. Japee, and R. W. Cox. Suma : an interface for surface-based intra- and inter-subject analysis with afni. In *2004 2nd IEEE International Symposium on Biomedical Imaging : Nano to Macro (IEEE Cat No. 04EX821)*, pages 1510–1513 Vol. 2, April 2004.
- [83] M. R. Sabuncu, B. T. T. Yeo, K. V. Leemput, B. Fischl, and P. Golland. A generative model for image segmentation based on label fusion. *IEEE Transactions on Medical Imaging*, 29(10) :1714–1729, Oct 2010.
- [84] B. Scherrer. *Segmentation des tissus et structures sur les IRM cérébrales : agents markoviens locaux et coopératifs et formulation bayésienne*. PhD thesis, 2008. Thèse de doctorat dirigée par Garbay, Catherine et Dojat, Michel Informatique et mathématiques appliquées Grenoble INPG 2008.
- [85] B. Scherrer, F. Forbes, C. Garbay, and M. Dojat. Distributed local mrf models for tissue and structure brain segmentation. *IEEE Transactions on Medical Imaging*, 28(8) :1278–1295, Aug 2009.
- [86] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, 1976.
- [87] D. W. Shattuck, M. Mirza, V. Adisetiyo, C. Hojatkashani, G. Salamon, K. L. Narr, R. A. Poldrack, R. M. Bilder, and A. W. Toga. Construction of a 3d probabilistic atlas of human cortical structures. *NeuroImage*, 39(3) :1064 – 1080, 2008.
- [88] F. Shi, P.-T. Yap, Y. Fan, J. H. Gilmore, W. Lin, and D. Shen. Construction of multi-region-multi-reference atlases for neonatal brain mri segmentation. *NeuroImage*, 51(2) :684 – 693, 2010.

- [89] T. Sørensen. *A Method of Establishing Groups of Equal Amplitude in Plant Sociology Based on Similarity of Species Content and Its Application to Analyses of the Vegetation on Danish Commons*. Biologiske Skrifter // Det Kongelige Danske Videnskabernes Selskab. I kommission hos E. Munksgaard, 1948.
- [90] A. Sotiras, C. Davatzikos, and N. Paragios. Deformable medical image registration : A survey. *IEEE Transactions on Medical Imaging*, 32(7) :1153–1190, July 2013.
- [91] C. Studholme, C. Drapaca, B. Iordanova, and V. Cardenas. Deformation-based mapping of volume change from serial brain mri in the presence of local tissue contrast change. *IEEE Transactions on Medical Imaging*, 25(5) :626–639, May 2006.
- [92] W. H. Sweet. The uses of nuclear disintegration in the diagnosis and treatment of brain tumor. *New England Journal of Medicine*, 245(23) :875–878, 1951.
- [93] J. Talairach and P. Tournoux. *Co-planar stereotaxic atlas of the human brain. 3-Dimensional proportional system : an approach to cerebral imaging*. Thieme, New York, 1988.
- [94] J.-P. Thirion. Image matching as a diffusion process : an analogy with Maxwell’s demons. *Medical Image Analysis*, 2(3) :243–260, 1998.
- [95] E. M. van Rikxoort, I. Isgum, Y. Arzhaeva, M. Staring, S. Klein, M. A. Viergever, J. P. Pluim, and B. van Ginneken. Adaptive local multi-atlas segmentation : Application to the heart and the caudate nucleus. *Medical Image Analysis*, 14(1) :39 – 49, 2010.
- [96] P. Viola and W. M. Wells III. Alignment by maximization of mutual information. *International Journal of Computer Vision*, 24(2) :137–154, Sep 1997.
- [97] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2) :260–269, April 1967.
- [98] H. Wang, S. R. Das, J. W. Suh, M. Altinay, J. Pluta, C. Craige, B. B. Avants, and P. A. Yushkevich. A learning-based wrapper method to correct systematic errors in automatic image segmentation : Consistently improved performance in hippocampus, cortex and brain segmentation. *NeuroImage*, 55(3) :968–985, 2011.
- [99] H. Wang, J. W. Suh, S. R. Das, J. B. Pluta, C. Craige, and P. A. Yushkevich. Multi-atlas segmentation with joint label fusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(3) :611–623, March 2013.
- [100] X. Wang and J. M. Keller. Human-based spatial relationship generalization through neural/fuzzy approaches. *Fuzzy Sets and Systems*, 101(1) :5 – 20, 1999.
- [101] W. M. Wells, P. Viola, H. Atsumi, S. Nakajima, and R. Kikinis. Multi-modal volume registration by maximization of mutual information. *Medical Image Analysis*, 1(1) :35 – 51, 1996.
- [102] R. Wolz, P. Aljabar, J. V. Hajnal, A. Hammers, and D. Rueckert. Leap : Learning embeddings for atlas propagation. *NeuroImage*, 49(2) :1316 – 1325, 2010.
- [103] C. O.-d.-S. Xavier Artaechevarria, Arrate Muñoz-Barrutia. Efficient classifier generation and weighted voting for atlas-based segmentation : two small steps faster and closer to the combination oracle. *Proc.SPIE*, 6914 :6914 – 6914 – 9, 2008.

- [104] Y. Y. Boykov and M.-p. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. 1 :105–112, 07 2001.
- [105] F. Yang and F. Kruggel. A graph matching approach for labeling brain sulci using location, orientation, and shape. *Neurocomputing*, 73(1) :179 – 190, 2009.
- [106] P. A. Yushkevich, J. Piven, H. C. Hazlett, R. G. Smith, S. Ho, J. C. Gee, and G. Gerig. User-guided 3d active contour segmentation of anatomical structures : Significantly improved efficiency and reliability. *NeuroImage*, 31(3) :1116 – 1128, 2006.
- [107] D.-Q. Zhang and S.-C. Chen. A novel kernelized fuzzy c-means algorithm with application in medical image segmentation. *Artificial Intelligence in Medicine*, 32(1) :37 – 50, 2004.
- [108] W. Zhang, R. Li, H. Deng, L. Wang, W. Lin, S. Ji, and D. Shen. Deep convolutional neural networks for multi-modality iso-intense infant brain image segmentation. *NeuroImage*, 108 :214 – 224, 2015.
- [109] Y. Zhang, M. Brady, and S. Smith. Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. *IEEE Transactions on Medical Imaging*, 20(1) :45–57, Jan 2001.
- [110] X. Zhuang, K. S. Rhode, R. S. Razavi, D. J. Hawkes, and S. Ourselin. A registration-based propagation framework for automatic whole heart segmentation of cardiac mri. *IEEE Transactions on Medical Imaging*, 29(9) :1612–1625, Sept 2010.
- [111] X. Zhuang and J. Shen. Multi-scale patch and multi-modality atlases for whole heart segmentation of mri. *Medical Image Analysis*, 31(Supplement C) :77 – 87, 2016.

BIBLIOGRAPHIE

Résumé :

Cette thèse présente une nouvelle méthode de segmentation interactive et incrémentale d'images médicales 3D. Pour cela, nous proposons une nouvelle manière, plus locale, de modéliser les connaissances *a priori* décrivant les structures à identifier. Plusieurs atlas locaux sont ainsi construits décrivant chacun uniquement une des structures d'intérêt. Nous proposons d'exploiter un graphe attribué pour lier les structures anatomiques les unes avec les autres. Les nœuds du graphe correspondent alors aux structures locales à segmenter et les arcs représentent les relations spatiales existantes entre ces régions. Ces relations spatiales ont pour objectif de permettre le positionnement des atlas locaux au sein de l'image entière. Elles sont définies sous forme d'intervalles de distance relative. Cette représentation renferme les connaissances *a priori* associées à chaque région de manière indépendante afin de pouvoir être utilisées individuellement. Lors de la segmentation, le graphe est utilisé suivant un processus incrémental permettant de réaliser des segmentations partielles et rapides tout en choisissant l'ordre de segmentation des différentes régions. L'utilisateur peut aussi intervenir sur le positionnement des atlas locaux afin d'améliorer la qualité de la segmentation obtenue. Une classification des voxels par champ de Markov a été proposée pour produire les segmentations locales. Enfin, notre méthode locale englobe un post-traitement afin de corriger les erreurs systématiques que notre méthode de segmentation peut produire. De nombreuses expérimentations ont permis de mettre en avant les avantages et les inconvénients de notre méthode et ont également démontré son efficacité par rapport aux méthodes de l'état de l'art en offrant une qualité comparable dans des délais plus courts.

Mots clés : Segmentation interactive, Segmentation incrémentale, Atlas probabilistes, Champs de Markov Cachés, Atlas locaux, graphe de connaissance *a priori*

Abstract :

This research work describes a new interactive and incremental method for the segmentation of 3D medical images. The *a priori* information associated to the anatomical structure to analyze is learned in a local way. Several local atlases, each one describing only one anatomical structure are constructed from a training dataset. An attributed graph is used to link the structures between each other. The nodes of the graph depict the anatomical structures, and the edges depict the spatial relationships between those regions. The spatial relationships aim to position the local atlases inside the whole image. They are defined as relative distances between the borders of the regions. This graph representation contains the *a priori* information associated with each region independently in order to be used individually. During the segmentation process, the graph is used in an incremental way allowing fast and partial segmentation. The user can also interact during the local atlas positioning in order to improve the segmentation quality. A voxel classification by an hidden Markov random field is employed to provide the local segmentations. We also propose a post-processing step in order to correct the systematic errors that a segmentation can achieve. Several experiments have been performed to evaluate and show the main advantages and drawbacks of our method. A comparison with the state of the art methods has shown that the segmentation quality is similar but faster.

Keywords : Interactive segmentation, Incremental segmentation, probabilistic atlas, Markov Random Field, Local atlas, Graph of *a priori* information