



**HAL**  
open science

## Contribution à la détection de changements dans des séquences IRM 3D multimodales

Marcel Bosc

► **To cite this version:**

Marcel Bosc. Contribution à la détection de changements dans des séquences IRM 3D multimodales. Neurosciences [q-bio.NC]. Université Louis Pasteur - Strasbourg I, 2003. Français. NNT: . tel-00005163

**HAL Id: tel-00005163**

**<https://theses.hal.science/tel-00005163>**

Submitted on 28 Feb 2004

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

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



N° d'ordre :

## · THÈSE ·

présentée pour obtenir le grade de  
Docteur de l'Université Louis Pasteur - Strasbourg I

École doctorale : Sciences pour l'ingénieur  
Discipline : Électronique, électrotechnique, automatique  
Spécialité : Traitement d'images et vision par ordinateur

# Contribution à la détection de changements dans des séquences IRM 3D multimodales

Soutenue publiquement  
le 17 décembre 2003  
par

Marcel BOSCH

Membres du jury :

M.	Karl	TOMBRE	Rapporteur externe
Mme.	Su	RUAN	Rapporteur externe
M.	Jean-François	DUFOURD	Rapporteur interne
M.	Lucien	RUMBACH	Examineur
M.	Fabrice	HEITZ	Directeur de thèse
M.	Jean-Paul	ARMSPACH	Directeur de thèse



# Remerciements

**Membres du Jury** Je remercie les membres du jury, Karl Tombre, Su Ruan, Jean-François Dufourd, Lucien Rumbach, Fabrice Heitz et Jean-Paul Armspach pour l'évaluation attentive de mon travail.

**Directeurs de thèse** Je tiens à remercier mes directeurs de thèse, Fabrice Heitz et Jean-Paul Armspach pour leur implication et leur soutien efficace durant ces trois années. Leurs compétences complémentaires et l'intérêt qu'ils ont portés à mon travail ont permis l'aboutissement de cette thèse. En particulier les nombreuses discussions animées et amicales avec Jean-Paul ont été enrichissantes.

**Contributions** Je tiens à remercier Torbjorn Vik pour les nombreuses discussions et suggestions ayant fortement contribué à ce travail. Je remercie Lucien Rumbach et Yves Chevalier pour leur participation au protocole d'évaluation du système de détection automatique de changements inter-images. Je remercie Izzie Namer et Daniel Gounot pour l'acquisition de la base d'images de Sclérose en Plaques sur laquelle repose ce travail.

**Manuscrit** Je remercie mes directeurs de thèse et ma compagne, Barbara Ripoché, pour leurs relectures attentives et les nombreuses corrections qu'ils ont apportées à l'ensemble de ce manuscrit. Je remercie de même Nicolas Passat, Daniel Gounot et Daniel Grucker pour leurs corrections. Je remercie aussi Valérie Wolff et Marie-Noëlle Metz-Lutz pour la relecture et leurs contributions aux aspects médicaux de l'introduction de ce manuscrit.

**Logiciel** Je remercie Torbjorn Vik pour ses importantes contributions à ImLib3D, le système logiciel sur lequel a été réalisé ce travail (voir Annexe-B). Je tiens à remercier tous les programmeurs ayant distribué leurs logiciels en OpenSource, sans lesquels ce travail aurait été impossible.

**Divers** Je tiens à remercier les différents membres de l'équipe ayant contribué par leurs avis, encouragements et discussions à enrichir ce travail. En particulier (par ordre alphabétique) Thomas Berst, Fabien Bouleau, Sylvain Faisan, Daniel Gounot, Paulo Loureiro, Olivier Musse, Vincent Noblet, Nicolas Passat, Samuel Sinapin, et Torbjorn Vik. Je remercie Daniel Grucker pour m'avoir accueilli dans son laboratoire. Je remercie aussi le personnel administratif de l'Institut de Physique Biologique. Je remercie bien évidemment mes parents et Barbara pour tous les efforts qu'ils ont déjà fait et aussi ceux à venir...

**Financement** Les entités suivantes ont participé au financement de ce travail : Région Alsace, Ligue Française contre la Sclérose En Plaques (LFSEP), Institut de Physique Biologique (UMR-7004), SERONO et BIOGEN.





# Table des matières

<b>I</b>	<b>Introduction</b>	<b>9</b>
<b>1</b>	<b>Présentation</b>	<b>11</b>
1.1	Problématique . . . . .	11
1.2	Organisation du manuscrit et contributions . . . . .	12
1.2.1	Organisation du manuscrit . . . . .	12
1.2.2	Contributions . . . . .	14
1.2.3	Illustrations . . . . .	14
<b>2</b>	<b>Introduction à l'imagerie cérébrale</b>	<b>17</b>
2.1	Imagerie par Résonance Magnétique . . . . .	17
2.1.1	Imagerie cérébrale et IRM . . . . .	17
2.1.2	Principe de l'IRM . . . . .	18
2.1.3	Modalités IRM . . . . .	19
2.1.4	Résolution des images IRM . . . . .	19
2.1.5	Artefacts IRM . . . . .	20
2.2	Anatomie cérébrale . . . . .	23
2.2.1	Description anatomique . . . . .	23
2.3	Pathologies cérébrales et leurs évolutions . . . . .	28
2.3.1	Présentation des pathologies cérébrales . . . . .	28
2.3.2	La sclérose en plaques . . . . .	35
<b>II</b>	<b>Segmentation du cortex</b>	<b>41</b>
<b>3</b>	<b>Problématique, état de l'art</b>	<b>43</b>
3.1	Introduction à la segmentation des structures cérébrales . . . . .	43
3.1.1	Applications . . . . .	44
3.1.2	Objectifs et propriétés souhaitables pour la segmentation . . . . .	46
3.1.3	Spécificités de la segmentation de l'IRM cérébrale . . . . .	47
3.2	État de l'art . . . . .	49
3.2.1	Méthodes de segmentation des structures non corticales . . . . .	49
3.2.2	Méthodes de segmentation des structures corticales . . . . .	49
3.2.3	Segmentation corticale : outils . . . . .	51
3.3	Conclusion . . . . .	54
<b>4</b>	<b>Description de notre approche</b>	<b>55</b>
4.1	Présentation . . . . .	55
4.2	Évolution de la segmentation . . . . .	58

4.2.1	Restriction de l'évolution aux frontières : contrainte $E_0$ . . . . .	60
4.2.2	Approximation de localité . . . . .	60
4.3	Contraintes : introduction . . . . .	61
4.4	Contraintes image : l'attache aux données . . . . .	62
4.4.1	Notations . . . . .	62
4.4.2	Processus d'acquisition . . . . .	63
4.4.3	Contrainte image à échelle variable . . . . .	65
4.4.4	Développement analytique de la variation d'énergie $\Delta E_l$ . . . . .	65
4.4.5	Correction de l'artefact RF apparaissant en IRM . . . . .	69
4.5	Contraintes <i>a priori</i> . . . . .	69
4.5.1	Contraintes : distribution relative . . . . .	70
4.5.2	Contraintes : épaisseur . . . . .	76
4.6	Pré-traitements et initialisation . . . . .	85
4.6.1	Estimation de l'intensité des labels . . . . .	85
4.6.2	Initialisation . . . . .	85
<b>5</b>	<b>Résultats, perspectives, et conclusion</b> . . . . .	<b>89</b>
5.1	Segmentation des sillons corticaux . . . . .	89
5.1.1	Construction du modèle des sillons corticaux . . . . .	89
5.1.2	Validation expérimentale . . . . .	91
5.1.3	Mise en œuvre pratique de la segmentation . . . . .	94
5.1.4	Convergence et rayon d'interaction . . . . .	94
5.1.5	Temps de calcul, encombrement mémoire et structures de données . . . . .	97
5.2	Perspectives . . . . .	99
5.2.1	Atlas statistique . . . . .	100
5.3	Conclusion . . . . .	102
<b>III</b>	<b>Détection de changements</b> . . . . .	<b>105</b>
<b>6</b>	<b>État de l'art de la détection de changements</b> . . . . .	<b>107</b>
6.1	Présentation . . . . .	107
6.2	Détection de changements en télédétection . . . . .	108
6.3	Détection de changements dans des séquences vidéo . . . . .	109
6.4	Détection de changements en imagerie cérébrale . . . . .	110
6.4.1	Recalage interimage . . . . .	112
6.4.2	Comparaison interimage . . . . .	113
6.5	Conclusion . . . . .	115
<b>7</b>	<b>Description de notre approche</b> . . . . .	<b>117</b>
7.1	Présentation des traitements . . . . .	117
7.2	Extraction du cerveau . . . . .	117
7.3	Repositionnement et correction de déformations . . . . .	119
7.3.1	Présentation . . . . .	119
7.3.2	Détermination des images de référence pour le recalage . . . . .	120
7.3.3	Recalage affine itératif robuste . . . . .	121
7.3.4	Recalage déformable . . . . .	121
7.3.5	Rééchantillonnage . . . . .	122

---

7.4	Correction des intensités et de l'artefact RF . . . . .	122
7.4.1	Normalisation d'intensité sur l'histogramme conjoint . . . . .	123
7.4.2	Artefact RF . . . . .	129
7.5	Modèle statistique pour la détection de changements . . . . .	129
7.5.1	Post-traitements . . . . .	131
<b>8</b>	<b>Protocole expérimental, résultats</b>	<b>133</b>
8.1	Patients . . . . .	134
8.2	Imagerie . . . . .	134
8.3	Évolutions simulées . . . . .	134
8.4	Évaluation expérimentale . . . . .	135
8.4.1	Protocole expérimental . . . . .	135
8.4.2	Résultats expérimentaux . . . . .	136
8.5	Discussion . . . . .	140
<b>9</b>	<b>Conclusions et perspectives</b>	<b>143</b>
9.1	Bilan et conclusions générales . . . . .	143
9.1.1	Bilan des contributions à la segmentation corticale . . . . .	143
9.1.2	Bilan des contributions à la détection de changements . . . . .	143
9.1.3	Conclusion . . . . .	144
9.2	Perspectives . . . . .	145
<b>IV</b>	<b>Annexes</b>	<b>147</b>
<b>A</b>	<b>Réflexions sur le développement logiciel et la recherche</b>	<b>149</b>
A.1	Reproductibilité et démarche scientifique . . . . .	149
A.2	Logiciels libres . . . . .	151
A.3	Freins à l'extension des logiciels libres dans la recherche . . . . .	153
<b>B</b>	<b>ImLib3D</b>	<b>155</b>
B.1	Fondements conceptuels . . . . .	156
B.1.1	Généricité . . . . .	156
B.1.2	Itérateurs . . . . .	157
B.1.3	Conception orientée objet . . . . .	158
B.2	La librairie ImLib3D . . . . .	158
B.2.1	Les fonctionnalités communes à toutes les images . . . . .	160
B.2.2	Les différentes classes d'images . . . . .	161
B.2.3	Itérateurs : le parcours des images . . . . .	163
B.2.4	Gestion des processeurs de traitement d'images . . . . .	164
B.2.5	Processeurs natifs . . . . .	164
B.2.6	Outils . . . . .	167
B.3	Le visualiseur . . . . .	167
B.3.1	Vues . . . . .	167
B.3.2	Autres fonctionnalités . . . . .	168
B.4	Interface ligne de commande . . . . .	169
B.5	Extensibilité d'ImLib3D . . . . .	169
B.5.1	Extensions des processeurs de traitement d'image . . . . .	169

B.5.2	Extensions du visualiseur . . . . .	169
B.6	Perspectives et conclusions . . . . .	170
<b>C</b>	<b>Segmentation : variations d'énergie des contraintes</b>	<b>171</b>
C.1	Approximation de la variation de distance en un point $\Delta^a D_k(x)$ . . . . .	171
C.2	Différentielle d'une distance . . . . .	171
<b>D</b>	<b>Publications de l'auteur</b>	<b>175</b>
	<b>Bibliographie</b>	<b>176</b>

# Première partie

## Introduction



# Chapitre 1

## Présentation

### 1.1 Problématique

L'imagerie médicale a profondément influencé à la fois la recherche médicale et la pratique clinique. Elle est aujourd'hui incontournable aussi bien pour l'établissement du diagnostic que pour la mise en place et le suivi d'un traitement thérapeutique. Elle fournit un volume croissant de données tridimensionnelles provenant de modalités d'acquisition différentes (IRM, scanner-X, médecine nucléaire, échographie). Ce volume croissant de données rend délicate et laborieuse la tâche d'interprétation par un expert. Le traitement d'images est un outil permettant une automatisation des tâches et va assister l'expert aussi bien dans l'analyse qualitative que quantitative des images. Dans ce mémoire, nous proposons des techniques automatiques de détection de changements dans des séquences d'images IRM cérébrales. Nous nous intéressons plus particulièrement aux changements d'intensité localisés (figure 1.1) survenant lors d'évolutions pathologiques telles que les évolutions de lésions en sclérose en plaques (SEP). Les applications médicales des techniques développées ici sont nombreuses : aide au diagnostic, suivi à long terme de l'évolution d'une pathologie, évaluation de l'efficacité thérapeutique d'un médicament, aide à la prise de décision en vue d'une intervention chirurgicale. Ce travail de recherche a été mené en étroite collaboration entre le LSIIT (ULP/UMR CNRS 7005) et l'Institut de Physique Biologique (ULP-Hôpitaux Universitaires / UMR CNRS 7004), au sein de l'équipe-projet multi-laboratoires « imagerie et Robotique Médicale et Chirurgicale » (EPML IRMC). Il a été soutenu par la Ligue Française Contre la Sclérose En Plaques (LFSEP), la société SERONO et la région Alsace.

La détection automatique et fiable de changements interimages rencontre d'importantes difficultés rendant impossible la comparaison directe d'images acquises successivement. La position des patients dans l'imageur n'est jamais identique et les paramètres d'acquisition peuvent varier sensiblement entre chaque examen, entraînant, entre autres, des modifications de contraste. La définition même de ce qui doit être détecté est souvent subjective. Dans le cadre spécifique de la détection de changements d'intensité de lésions, des déformations globales de structures anatomiques, telle que l'atrophie cérébrale, peuvent également perturber la comparaison directe des images.

Le travail présenté dans cette thèse est centré sur le développement d'outils de traitement d'images permettant de décider quels changements sont statistiquement significatifs ou non. Lorsque l'expert détermine visuellement des changements, il utilise des connaissances *a priori*, implicites, de haut niveau qui lui permettent de corriger certaines erreurs d'acquisition. Ainsi, il peut compenser visuellement des erreurs de repositionnement et utiliser ses connaissances



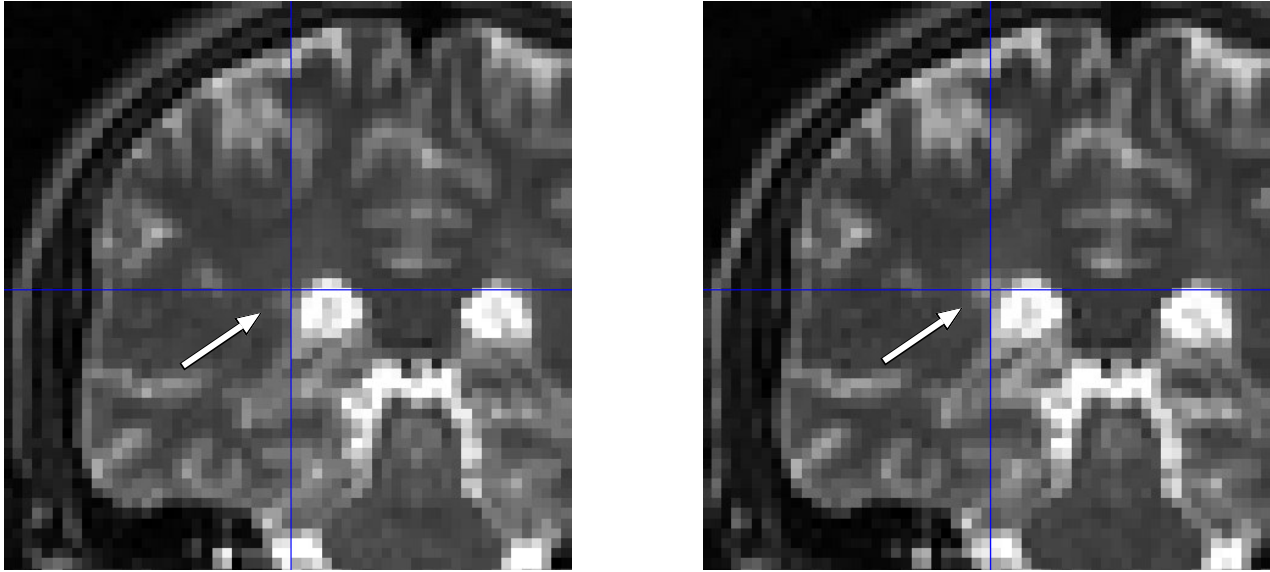


FIG. 1.1 – *Exemple d'une évolution de lésion de sclérose en plaque. Cette évolution a été détectée automatiquement par notre système, mais n'a pas été vue par l'expert (elle a été validée a posteriori). Les images sont agrandies, sans interpolation, pour faciliter la visualisation et mettre en évidence la résolution. Les images représentées ici, ont été recalées avec précision et divers artefacts d'acquisition ont été corrigés.*

anatomiques propres pour identifier et rejeter certains artefacts. Nous développons donc ici, des techniques automatiques d'identification et de correction des principaux artefacts (positionnement, déformations, variations d'intensité ...) et nous proposons une technique originale de segmentation du cortex, apportant les informations anatomiques permettant l'amélioration de la détection automatique. Les techniques de traitement d'images proposées ici ont été développées pour l'IRM cérébrale. Cependant, elles sont suffisamment générales pour s'appliquer à d'autres domaines.

Notre système de détection de changements a été évalué dans le cadre de l'étude de l'évolution de lésions de sclérose en plaques. Ses performances ont été déterminées sur une grande base d'images multimodales (plus de 200 images FLAIR, RARE et GE3D) de taille  $128^3$ . L'évaluation a été faite à l'aide d'un protocole impliquant deux experts (neurologues) et utilisant une analyse statistique de type COR<sup>1</sup>. Le système automatique a été jugé plus performant que l'expert humain.

## 1.2 Organisation du manuscrit et contributions

### 1.2.1 Organisation du manuscrit

Dans la **première partie** de ce manuscrit, nous présentons tout d'abord les éléments d'imagerie IRM et les aspects médicaux nécessaires à la compréhension de l'ensemble de ce travail. Nous décrivons les modalités d'acquisition IRM et les artefacts associés. Cette étape est importante pour la compréhension des imperfections pouvant apparaître et leur correction. Nous présentons ensuite des éléments sur l'anatomie cérébrale et nous décrivons l'apparence prise les différentes structures cérébrales dans les trois modalités IRM considérées. Puis, nous

<sup>1</sup>Caractéristique Opérationnelle du Récepteur

terminons par les pathologies cérébrales, leurs évolutions, et leur aspect en IRM. Les objectifs et les limites de notre approche sont situés par rapport à ce contexte applicatif.

Dans une **deuxième partie** nous décrivons une approche nouvelle de segmentation sous-voxel. Pour décider de la pertinence d'un changement observé, l'expert utilise des connaissances anatomiques. Dans notre système de détection automatique, ces connaissances sont obtenues en segmentant l'image du cerveau. La méthode de segmentation proposée est basée sur l'évolution d'une image de labels de très haute résolution. L'évolution se fait sous l'influence de contraintes statistiques multiples, exprimées dans un cadre de minimisation d'énergie. L'évolution de l'image de labels n'ayant lieu qu'à la frontière entre régions, notre approche est comparable à un système d'évolution de surfaces. Afin de s'adapter aux spécificités de chaque région cérébrale, les contraintes sont paramétrées à l'aide d'un atlas. Celui-ci, composé d'une image de référence et d'images de paramètres, est plaqué sur l'image à segmenter à l'aide d'un recalage déformable multi-échelles. Les contraintes sont classées en deux catégories : les contraintes image (attache aux données) et le modèle *a priori*. Plusieurs contraintes image, opérant simultanément à des échelles différentes, sont employées. Elles utilisent une description rigoureuse du processus d'acquisition, permettant ainsi d'atteindre à la fois une précision sous-voxel et une convergence globale (à grande échelle). Le modèle *a priori* est également composé de plusieurs contraintes : une contrainte de distribution relative qui donne la probabilité d'observer un label à une distance donnée d'un autre label et une contrainte d'épaisseur. Notre approche permet d'obtenir une segmentation de haute résolution à partir d'images IRM pouvant être de résolution inférieure. La performance du système de segmentation a été évaluée sur des images simulées et testée sur des images réelles.

La **troisième partie** présente l'ensemble de la chaîne de traitements conduisant à la détection de changements, ainsi que le protocole d'évaluation et les résultats. La chaîne de traitements est constituée d'une première étape de repositionnement et de correction des déformations. Toutes les images de la base sont alignées sur des références soigneusement choisies, d'abord à l'aide d'une méthode de recalage affine itératif robuste, puis à l'aide de recalage déformable. Au cours de la deuxième étape, les deux images à comparer subissent une correction d'intensité non-linéaire ainsi qu'une élimination d'erreurs résiduelles. La méthode de correction d'intensité que nous proposons permet d'établir une fonction de transfert d'intensité non-linéaire en optimisant un critère simple s'appuyant sur des informations de l'histogramme conjoint. Finalement, au cours de la dernière étape, une approche de détection statistique multimodale permet de décider quels changements sont significatifs. Les connaissances anatomiques fournies par la segmentation sont utilisées pour éliminer certaines détections aberrantes. L'ensemble de ces traitements est appliqué de manière entièrement automatique sur une base de plus de 200 images, de modalités différentes, démontrant ainsi la fiabilité des traitements. La validation du système a été menée à l'aide d'un protocole d'évaluation comprenant deux experts (neurologues). Le premier expert ainsi que le système automatique ont procédé indépendamment à un même travail de détection (l'expert opérant manuellement). Le second expert fait ensuite office d'arbitre pour comparer les résultats des deux procédés. L'analyse COR permet une vue synthétique de la performance du détecteur en donnant la probabilité de détection en fonction du nombre de fausses alarmes. Dans un cadre applicatif, les modifications détectées par le système automatique sont ordonnées par vraisemblance décroissante et présentées au neurologue dans un système de visualisation interactif. Ceci permet au médecin de conserver la décision finale, tout en parcourant efficacement et très rapidement les modifications détectées.

En **annexe** nous proposons quelques réflexions sur l'importance du développement logiciel

et de sa diffusion dans la recherche en traitement d'images. Nous présentons ensuite **ImLib3D**, une librairie C++ dédiée à la recherche en traitement d'images volumiques, que nous avons développée dans le cadre de cette recherche. ImLib3D propose à la fois un système de visualisation séparé et une librairie soigneusement conçue à l'aide d'une méthodologie orientée objet et utilisant des concepts modernes s'inspirant de la librairie standard du C++. L'objectif, dans la conception, a été de créer une librairie simple à utiliser par le chercheur, considéré comme le public cible. ImLib3D est distribuée librement (Open Source) et est placée dans un cadre de développement distribué coopératif (sourceforge.net).

En conclusion, nous avons élaboré un système complet et opérationnel de détection de changements dans lequel nous avons systématiquement analysé et traité les principaux artefacts gênant la détection.

### 1.2.2 Contributions

Les contributions scientifiques originales de cette recherche sont :

- un système de segmentation des plis corticaux basé sur un paradigme nouveau :
  - une approche multi-échelles d'intégration des informations image dans la segmentation modélisant rigoureusement les effets de volumes partiels (contrainte image) ;
  - deux contraintes (distribution relative et épaisseur) permettant d'apporter des informations *a priori* à la segmentation ;
- l'identification et l'analyse des problèmes de la détection et la proposition des solutions suivantes :
  - l'utilisation du recalage déformable comme étape de recalage ;
  - une approche non-linéaire de normalisation d'intensité ;
  - l'utilisation d'informations multimodales dans ce cadre spécifique de détection de changements ;
  - l'utilisation d'un cadre probabiliste pour la comparaison interimages
  - l'utilisation d'informations provenant de la segmentation des sillons corticaux pour la détection.

Un bilan plus détaillé de ces contributions est proposé dans la conclusion de ce manuscrit (§ 9.1, p. 143).

La principale contribution technique est un système logiciel complet, performant, librement disponible pour l'ensemble de la communauté de traitement d'images.

### 1.2.3 Illustrations

Toutes les images traitées dans ce travail de thèse sont des images volumiques, à l'exception de quelques cas particuliers, dans la partie II, explicitement signalés. Dans ce manuscrit, les images volumiques sont illustrées soit par une coupe unique, soit par une représentation multiplanaire. La représentation multiplanaire est composée de trois coupes perpendiculaires s'intersectant en un point indiqué par une mire (voir figure 1.2).

Toutes les images IRM présentées dans ce mémoire, à l'exclusion de celles dont nous avons explicitement cité une autre origine, ont été acquises sur l'imageur (Tomikon S200 Bruker) de l'IPB (UMR CNRS 7004).

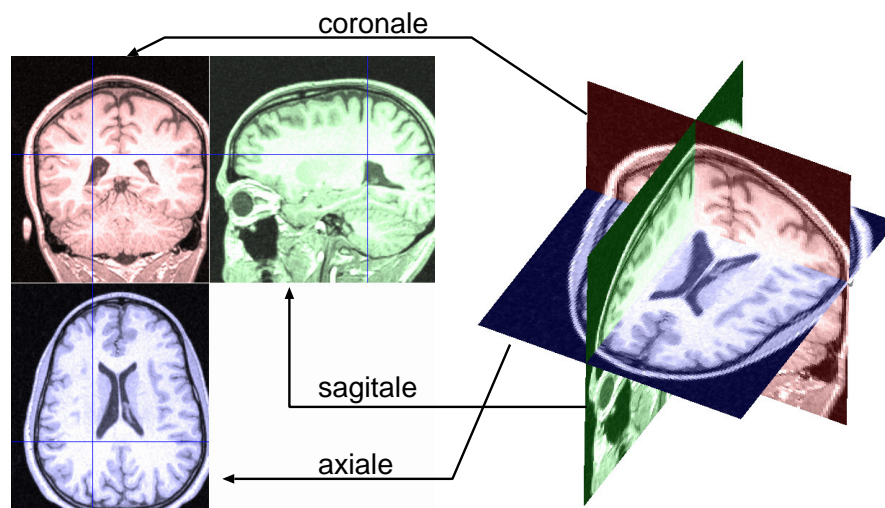


FIG. 1.2 – Représentation multiplanaire d'une image volumique, constituée de trois coupes perpendiculaires en un point indiqué par une mire. En imagerie médicale les trois plans de coupes portent les noms de coupe coronale, sagittale et axiale.



# Chapitre 2

## Introduction à l'imagerie cérébrale

Ce chapitre a pour objectif de présenter les notions d'imagerie cérébrale nécessaires à la compréhension de ce travail. Il s'adresse, pour l'essentiel, à un lecteur ayant des connaissances en traitement d'images, qui aborderait l'imagerie cérébrale pour la première fois. Pour permettre au lecteur intéressé d'approfondir ces notions, des références d'ouvrages pédagogiques sont fournies.

### 2.1 Imagerie par Résonance Magnétique

Nous proposons ici une brève introduction<sup>1</sup> à l'Imagerie par Résonance Magnétique (IRM<sup>2</sup>). Après avoir situé l'IRM par rapport aux techniques d'acquisition (§ 2.1.1) et brièvement présenté ses principes physiques (§ 2.1.2), nous aborderons les aspects de l'IRM importants du point de vue du traitement d'images. Dans ce sens, nous décrirons d'abord les différents contrastes pouvant être observés dans les images (§ 2.1.3), puis des considérations liées à leur résolution (§ 2.1.4), avant de finir par les artefacts pouvant les perturber (§ 2.1.5).

#### 2.1.1 Imagerie cérébrale et IRM

**Techniques d'acquisition d'images** Les principales techniques d'acquisition d'images cérébrales volumiques sont la tomographie par émission de positons<sup>3</sup> (TEP), la tomographie par émission mono-photonique<sup>4</sup> (TEMP), la tomographie par rayons-X<sup>5</sup> (CT), et l'IRM.

Les deux premières (TEP, TEMP) permettent de mesurer l'activité cérébrale à l'aide de traceurs radioactifs, avec une résolution spatiale assez faible ( $\simeq 6$  mm, 3-4 mm pour les appareils récents). Le scanner-X fournit des images de bonne résolution ( $\simeq 1$  mm), mais avec un contraste entre tissus cérébraux insuffisamment marqué.

La troisième technique (IRM) permet d'acquérir des images de natures très différentes, sur

---

<sup>1</sup>Parmi les nombreuses introductions à l'IRM à vocation pédagogique, le lecteur pourra aisément se référer à l'ouvrage de Hornak [44], disponible en ligne, avec de nombreuses illustrations et animations.

<sup>2</sup>L'appellation « Résonance Magnétique Nucléaire » (RMN) a été délaissée en faveur de l'IRM, plus rassurante.

<sup>3</sup>Acronyme anglais : PET

<sup>4</sup>Acronyme anglais : SPECT

<sup>5</sup>La tomographie par rayons-X est aussi appelé tomodensitométrie ou scanner-X. On utilisera son acronyme anglais CT : computed tomography. L'acronyme CAT est aussi utilisé dans la littérature.

un même appareil, selon les séquences<sup>6</sup> utilisées. L'IRM, contrairement aux techniques précédentes, n'expose pas le sujet à des rayonnements ionisants. L'intérêt de l'IRM est multiple<sup>7</sup>. L'IRM *anatomique* fournit des détails anatomiques avec une résolution de l'ordre du millimètre (§ 2.1.4) et permet de différencier les tissus mous du cerveau, ce qui justifie son choix dans cette étude portant sur les lésions de sclérose en plaques. Nous décrirons les séquences permettant la visualisation des structures anatomiques ci-après (§ 2.1.3). L'IRM *fonctionnelle* (IRMf) permet de visualiser des informations sur l'activité cérébrale à partir de séries temporelles d'images (plusieurs centaines d'images,  $\simeq 5$  s par image) de basse résolution ( $\simeq 4$  mm). L'IRM *angiographique* (ARM, VRM), en saturant les tissus fixes et en rehaussant le sang en mouvement, permet de visualiser l'arbre vasculaire cérébral sans l'injection d'un produit de contraste. L'IRM de *diffusion* détermine la direction de diffusion des molécules d'eau et par conséquent celle des fibres tissulaires. L'IRM par *transfert d'aimantation* fournit d'autres informations physiologiques liées aux échanges d'aimantation entre les tissus et l'eau. Un appareil IRM permet également d'analyser les propriétés chimiques d'une région donnée (spectroscopie) avec de nombreuses applications médicales.

## 2.1.2 Principe de l'IRM

Avant de présenter la formation d'une image en IRM, il est nécessaire d'expliquer ce qu'est le signal de Résonance Magnétique Nucléaire (RMN).

**Signal RMN** En présence d'un champ magnétique, une partie des spins des noyaux des atomes s'alignent sur ce champ. Il est alors possible de changer l'état du spin de certains de ces noyaux en les stimulant avec une impulsion radiofréquence (RF). L'impulsion doit être modulée à une fréquence de résonance précise (fréquence de Larmor) qui dépend de l'intensité du champ magnétique. Les spins ayant changé d'état retrouvent leur état d'équilibre en émettant un signal radiofréquence (voir figure 2.1-a). Ce signal porte des informations mesurables sur les propriétés physiques des matériaux l'ayant émis.

**Formation de l'image** Si, au lieu d'appliquer un champ magnétique uniforme, on applique un champ qui croît linéairement dans une des trois directions de l'espace (champ de gradient) alors la fréquence de résonance dépendra de la position dans l'espace. Ce principe permet de distinguer différentes positions spatiales<sup>8</sup>. Le signal RMN mesuré est composé de différentes fréquences, qui après transformée de Fourier, décrivent une projection spatiale de l'objet dans la direction des gradients (voir figure 2.1-b). On peut distinguer deux types d'acquisition volumique : l'acquisition coupe par coupe et l'acquisition de l'image entière dans le domaine de Fourier.

Une image IRM est donc obtenue en appliquant des gradients de champ et des séquences d'impulsions radiofréquence.

<sup>6</sup>Le terme séquence fait référence aux séquences d'impulsions radiofréquence générant un signal permettant la reconstruction d'une image IRM (§ 2.1.2).

<sup>7</sup>L'IRM est souvent préférée à d'autres techniques d'acquisition (contraste, résolution), cependant, il s'agit d'un examen moins accessible. Les délais actuels pour un examen IRM, sauf cas d'urgence, sont de l'ordre de 4 à 6 mois.

<sup>8</sup>Il sera exploité dans la sélection de coupes, dans l'encodage fréquentiel et indirectement dans l'encodage de phase.

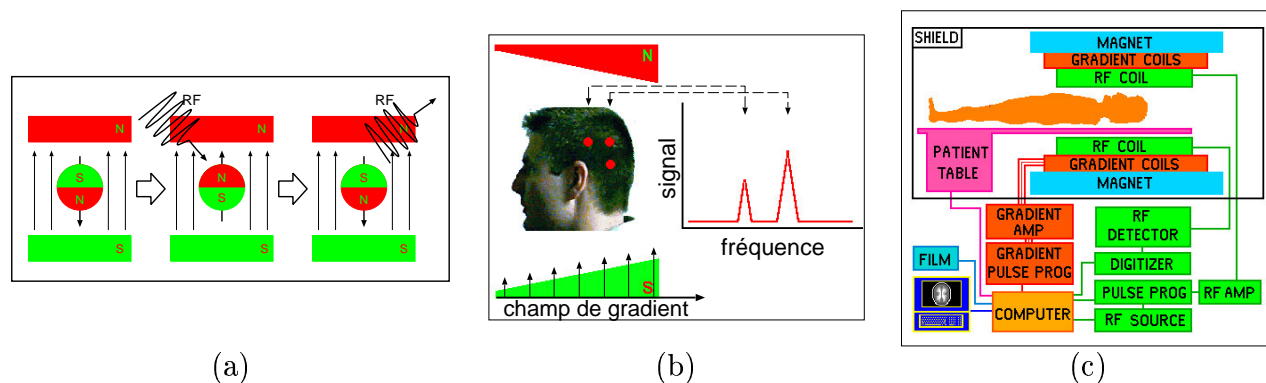


FIG. 2.1 – Principes de l'IRM. (a) Signal RMN : le spin d'un noyau aligné sur un champ magnétique peut être basculé par une impulsion radiofréquence. Il revient ensuite à son état initial en émettant un signal RF qui permet de mesurer des propriétés physiques de l'objet. (b) Formation de l'image : un champ de gradient permet de sélectionner une position dans une direction spatiale. (c) Vue d'ensemble d'un système d'acquisition IRM. (Illustrations (b) et (c) d'après Hornak [44], modifiée)

### 2.1.3 Modalités IRM

En variant ces séquences d'impulsions, on obtient des images portant des informations variées, appelées modalités<sup>9</sup>. Des informations sur trois propriétés physiques, entre autres, sont présentes dans le signal RMN : les temps de relaxations  $T_1$  et  $T_2$  et la densité de protons. En variant les séquences, on pourra privilégier certaines de ces informations par rapport à d'autres. Les images pondérées en  $T_1$  permettent de distinguer les tissus du cerveau, matière blanche (MB), matière grise (MG) et le liquide céphalo-rachidien (LCR) (figure 2.2-a). Les images pondérées en  $T_2$  mettent en valeur certaines anomalies (lésions SEP, par exemple : figure 2.2-b). Dans les images FLAIR (Fluid Attenuated Inversion Recovery) le signal du liquide est supprimé, permettant de distinguer des anomalies (lésions SEP) par rapport au liquide (figure 2.2-c).

Chaque constructeur propose des séquences optimisées pour leurs imageurs. L'imageur de notre équipe (Tomikon S200, Bruker) emploie des séquences spécifiques, nommées GE3D (pondérée en  $T_1$ , multi-coupes) RARE (pondérée en  $T_2$ , 3D). Il est aussi possible d'obtenir d'autres informations physiologiques en injectant au sujet un produit de contraste<sup>10</sup>.

### 2.1.4 Résolution des images IRM

Étant donné la faible amplitude ( $100 \mu\text{V}$ ) du signal RMN, la résolution d'une image IRM est limitée par la quantité de bruit dans le signal mesuré. Le rapport signal à bruit peut être amélioré en augmentant le temps d'acquisition (typiquement 3-5 mn). Cependant, celui-ci est limité par des considérations pratiques (durée de l'examen, inconfort pour le patient, coût) ainsi que par les mouvements du patient. En pratique clinique, les images sont souvent visualisées sur des films radiologiques. Dans ce cas, le clinicien ne peut visualiser qu'un nombre

<sup>9</sup>Le terme *modalité* est parfois utilisé pour distinguer les différents types d'imagerie (TEMP, IRM, scanner-X). Nous l'utiliserons aussi dans ce texte pour distinguer les différents types d'images IRM. Le terme *multimodal* sera utilisé pour désigner des techniques utilisant des informations provenant de plusieurs modalités (IRM) différentes. Dans la littérature, les termes *multi-spectral* ou *multi-paramétrique* sont aussi employés dans ce sens.

<sup>10</sup>Le Gadolinium est un produit de contraste paramagnétique couramment utilisé en IRM.



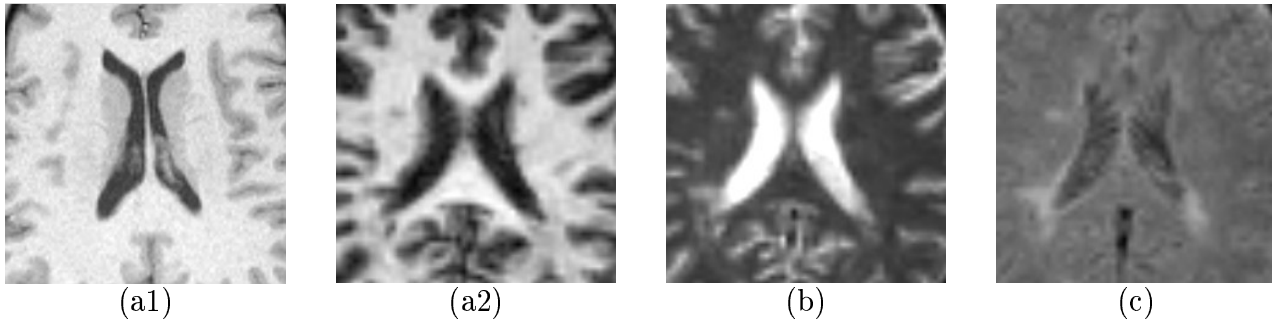


FIG. 2.2 – *Trois modalités d'IRM. (a) Image pondérée en  $T_1$ . Les trois composantes principales (LCR, MG, MB) peuvent être distinguées. (b) Image pondérée en  $T_2$ . La matière grise est difficile à distinguer de la MB. (c) Image FLAIR, le liquide est supprimé. Les images (a2, b, c) contiennent des lésions de sclérose en plaques, particulièrement visibles en FLAIR (c).*

limité de coupes. Il est alors possible d'acquérir chaque coupe avec une meilleure résolution ( $\geq 256 \times 256$ ). Remarquons que la gestion, en traitement d'images, de ces coupes épaisses peut poser des problèmes pratiques délicats<sup>11</sup>. L'acquisition d'images isotropes<sup>12</sup> peut se faire coupe par coupe (multi-coupes) ou bien directement dans le domaine de Fourier. Dans le premier cas, l'image de chaque coupe successive est calculée à partir d'un signal RMN provenant de l'ensemble de la coupe sélectionnée. Dans le deuxième cas, l'image est calculée à partir d'un signal RMN provenant de l'ensemble du cerveau simultanément. Ces deux approches donnent lieu à des artefacts ayant des aspects différents. En acquisition multi-coupes, isotropes ou non, différents profils de coupes<sup>13</sup> sont envisageables. Voici quelques exemples de résolutions utilisées en IRM :

- $2 \times 2 \times 2$  mm sur une matrice de  $128 \times 128 \times 128$  voxels : résolution de la base de donnée d'images de sclérose en plaques sur laquelle repose cette étude ;
- $1 \times 1 \times 1$  mm sur une matrice de  $256 \times 256 \times 256$  voxels : résolution d'appareils récents ;
- $.5 \times .5 \times 4$  mm sur une matrice de  $512 \times 512 \times 64$  voxels : utilisé en clinique pour préciser la morphologie ;

### 2.1.5 Artefacts IRM

Les erreurs (artefacts) d'acquisition en IRM sont de nature très différente de celles observées dans d'autres domaines du traitement d'image (image « photographique », vidéo). Nous décrivons ici les artefacts de mouvement, les inhomogénéités RF, les problèmes de shim et le bruit aléatoire, qui sont les principaux artefacts abordés dans cette thèse.

**Artefacts de mouvement** Les mouvements lors de l'acquisition, comme les mouvements des yeux, les pulsations du flux sanguin, les mouvements respiratoires, les mouvements de la tête, sont responsables d'artefacts. Ces artefacts sont très différents des flous de bougé qu'on observe en photographie ou en vidéo. Ils se traduisent par des phénomènes d'aspect très

<sup>11</sup>En particulier, sur des images très anisotropes, il est indispensable que les problèmes d'interpolation et d'échantillonnage (filtrage anti-repliement) soient soigneusement traités.

<sup>12</sup>Une image isotrope est une image ayant la même résolution dans toutes les directions.

<sup>13</sup>Le profil de coupe correspond à la manière dont les informations autour de la coupe sont intégrées dans la coupe. Une coupe parfaite correspond à un profil carré, mais n'est pas réalisable. Des profils gaussiens sont le plus souvent utilisés.

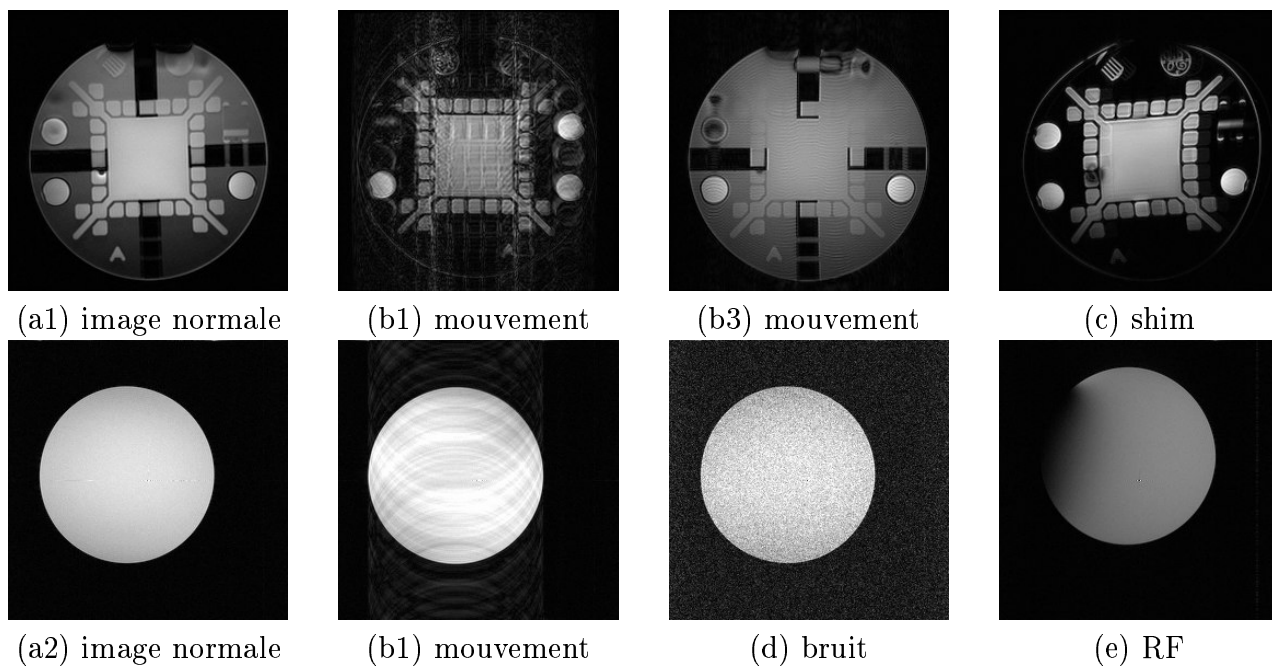


FIG. 2.3 – *Artefacts IRM classiques. (a) Image normale, ou « fantôme ». (b) De légers mouvements du fantôme peuvent provoquer des artefacts très distincts selon le moment de l'acquisition où ils ont lieu. (c) Si des distorsions du champ magnétique principal sont mal compensées (shim), des distorsions géométriques dans l'image peuvent apparaître. (d) Bruit aléatoire (e) artefact RF provoqué artificiellement en introduisant une plaque métallique qui produit une distorsion du champ RF (sources UCLA, University of Illinois).*

variable. Il s'agit souvent d'apparition d'images fantômes de la structure en mouvement (voir figures 2.3 et 2.4).

**Inhomogénéités RF** Des variations spatiales lentes (inhomogénéités) de l'intensité des images peuvent apparaître (figures 2.3 et 2.5) dans des images IRM [12]. Elles sont provoquées par des inhomogénéités dans le champ magnétique principal, des inhomogénéités dans le champ généré par les impulsions RF et dans le gain de l'antenne RF. Cet artefact n'est pas très gênant pour le clinicien, sauf dans des cas extrêmes, où le contraste entre tissus disparaît. En revanche, il peut poser d'importants problèmes à un système de traitement d'images automatique.

**Problèmes de shim, distorsions géométriques** L'homogénéité du champ magnétique principal est perturbée par la présence d'objets dont la susceptibilité magnétique n'est pas constante. Pour compenser ces inhomogénéités, des champs magnétiques supplémentaires (shim) sont ajoutés interactivement par l'opérateur. Des insuffisances dans cette procédure peuvent conduire à des distorsions géométriques de l'image (figure 2.3). Par ailleurs, des variations dans l'intensité des champs de gradient provoquent une compression ou une dilatation de l'image dans une direction spatiale (ce phénomène est parfois appelé « changement de taille de voxel »).

**Bruit aléatoire** Diverses sources de bruit dans les systèmes électroniques (bruit thermique, grenaille ...) sont responsables d'un bruit aléatoire gaussien apparaissant dans le domaine de

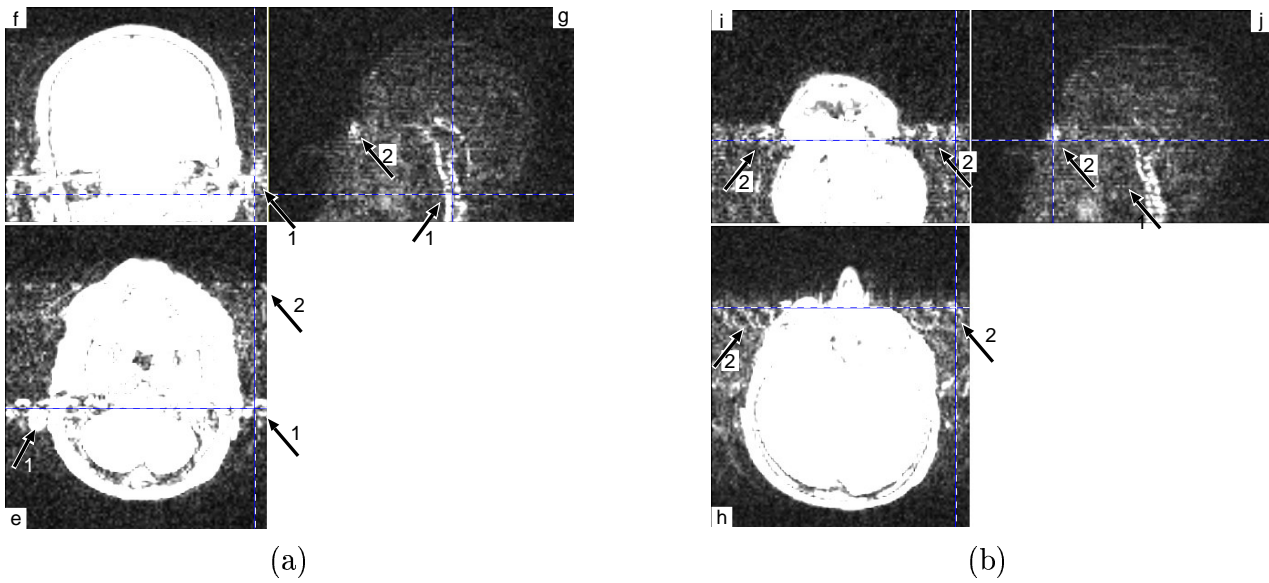


FIG. 2.4 – *Artefacts de mouvement IRM. Trois coupes (e, f, g) (h, i, j) en deux points différents (a, b) au travers d'un volume. Le contraste a été poussé lors de la visualisation. Les coupes (g) et (j) sont extérieures au cerveau. On voit clairement apparaître à l'extérieur de la tête des artefacts de mouvement dus aux pulsations dans les carotides (flèches-1) et aussi des artefacts dus aux mouvements des yeux (flèches-2). Ces artefacts, visibles ici à l'extérieur de la tête, se retrouvent sur toute l'image, y compris le cerveau où ils peuvent être gênants.*

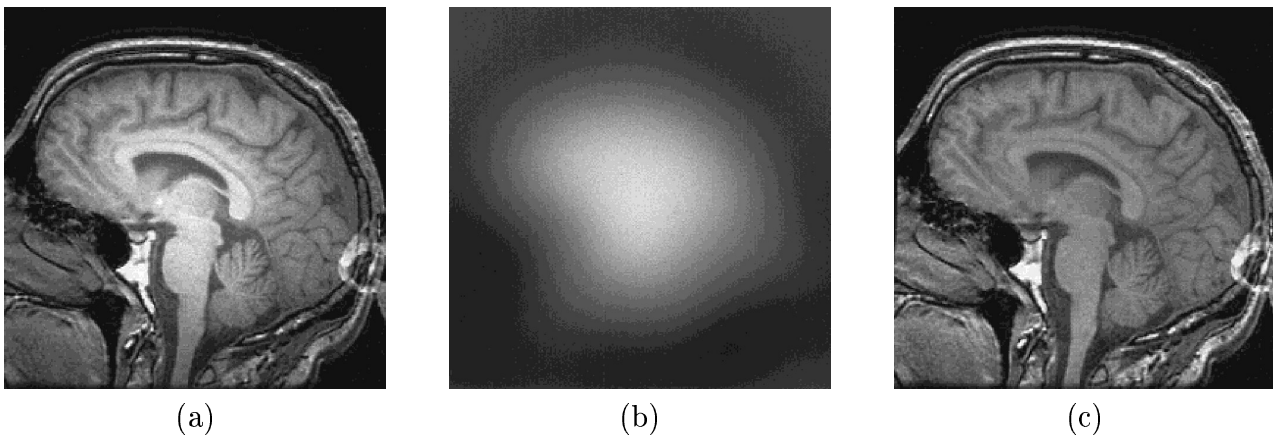


FIG. 2.5 – *Inhomogénéités RF. (a) image affectée par une inhomogénéité RF (b) l'artefact RF isolé (c) image sans l'artefact RF (D'après Cohen et al. [12], modifiée)*

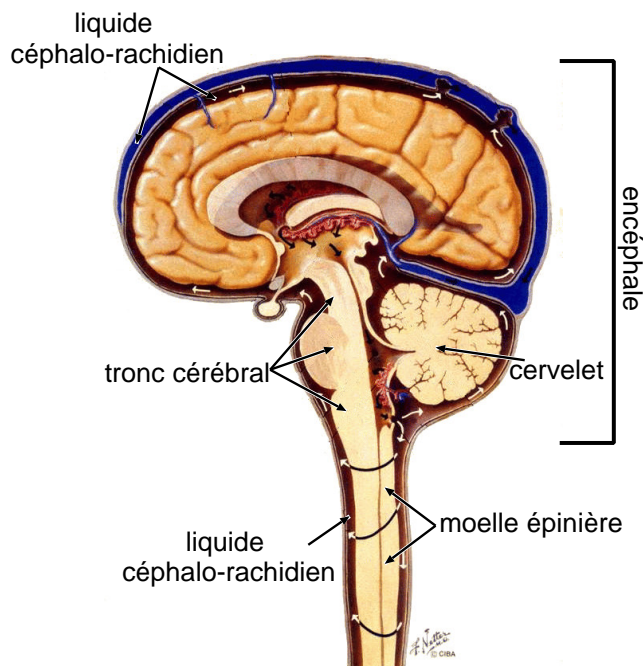


FIG. 2.6 – Le système nerveux central humain est constitué de l'encéphale et de la moelle épinière. L'ensemble flotte dans un coussin protecteur de liquide céphalo-rachidien. (Illustration d'après [81], modifiée)

Fourier de l'image acquise. L'image étant obtenue en calculant le module de la transformée de Fourier inverse, cette distribution gaussienne devient, dans l'image, une distribution de Rice [100] (figure 2.3-d). Dans les régions où l'intensité de l'image n'est pas proche de zéro, cette distribution est approximativement gaussienne. Dans les régions d'intensité proche de zéro, elle est proche d'une distribution de Raleigh.

## 2.2 Anatomie cérébrale

Nous décrivons ici brièvement les éléments anatomiques ainsi que la terminologie nécessaires à la compréhension de ce mémoire. Le lecteur qui souhaiterait approfondir pourra se référer aux nombreux ouvrages pédagogiques existants [84, 94], ou à des sources en ligne<sup>14</sup>.

### 2.2.1 Description anatomique

Le système nerveux central humain est constitué de l'encéphale et de la moelle épinière (figure 2.6). L'ensemble du système nerveux flotte dans un coussin protecteur de liquide céphalo-rachidien (LCR), maintenu dans une série de membranes appelées méninges. La partie supérieure de l'encéphale, où se déroulent la majorité des processus cognitifs et sensoriels, est divisée en deux : l'hémisphère droit et l'hémisphère gauche. Sous ces hémisphères, le tronc cérébral contrôle les fonctions vitales, comme la respiration, les pulsations cardiaques et les mouvements oculaires. Il constitue aussi le circuit intermédiaire où circulent les informations entre le cerveau et la moelle épinière.

<sup>14</sup>Virtual Hospital (<http://www.vh.org>) Section sur l'anatomie cérébrale : <http://www.vh.org/adult/provider/anatomy/BrainAnatomy/BrainAnatomy.html>

**Composition du cerveau : types de tissus** Le cerveau est essentiellement constitué de deux types de tissus différents : la matière blanche et la matière grise. La matière grise contient le corps des cellules nerveuses et peut être vue comme le lieu de traitement des données. La matière blanche est constituée de fibres de conduction, les axones des cellules nerveuses. Les axones, entourés de gaines de myéline de couleur blanchâtre, ont pour fonction de transmettre les données.

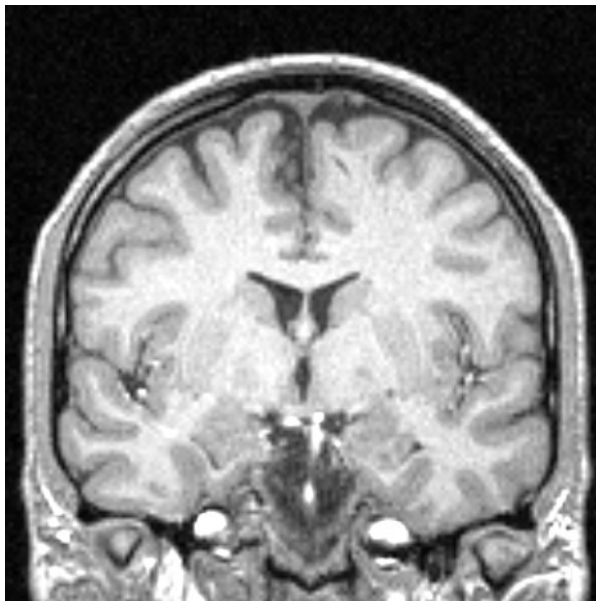
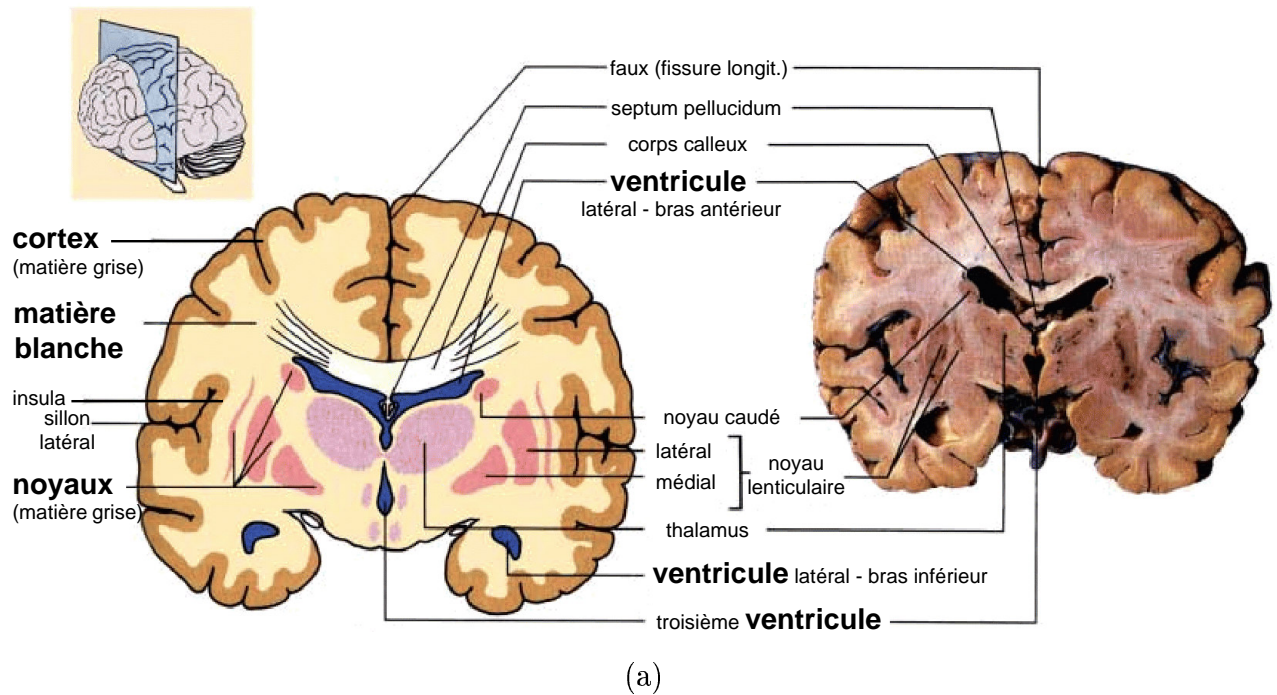
**Composition du cerveau : régions** La matière grise est répartie en deux types de structures : le cortex et les noyaux (figure 2.7). Le cortex a une structure bidimensionnelle. Il peut être vu comme une sphère, dont la surface aurait une épaisseur relativement constante (autour de 1,7 mm [68]), et qui aurait été repliée sur elle-même de manière très intriquée. Sur la face intérieure du cortex se trouve la matière blanche et sur sa surface extérieure circule le liquide céphalo-rachidien. La forme et la position des plis corticaux<sup>15</sup> sont très variables d'un individu à un autre. Les noyaux, aussi constitués de matière grise, sont des structures plus compactes se trouvant au centre du cerveau (figure 2.7). Nous utiliserons le terme structures internes pour désigner les noyaux gris (y compris le thalamus) et les ventricules. La taille des structures internes peut varier d'un individu à un autre (atrophie cérébrale avec dilatation des ventricules, par exemple), mais leur forme reste relativement stable.

**Couches externes au cerveau** Le cerveau est entouré d'une série de couches successives de tissus (figure 2.8) protecteurs : tout d'abord, de l'intérieur vers l'extérieur, les méninges, puis l'os et enfin le scalp (peau). Les méninges sont constituées de trois membranes successives (voir figure 2.8). La pie-mère, qui transporte les vaisseaux sanguins, est au contact direct du cerveau. Ensuite se trouve la membrane arachnoïdienne ressemblant à une toile d'araignée. Le LCR circule entre celle-ci et la pie-mère. La dernière membrane extérieure, plus épaisse et résistante est la dure-mère. Elle protège les couches précédentes, plus fines, de l'os. L'os, sombre en IRM, comporte en son centre des portions contenant de la moelle osseuse, claire en IRM. Une fine membrane, appelé le périoste, est située immédiatement à l'extérieur de l'os.

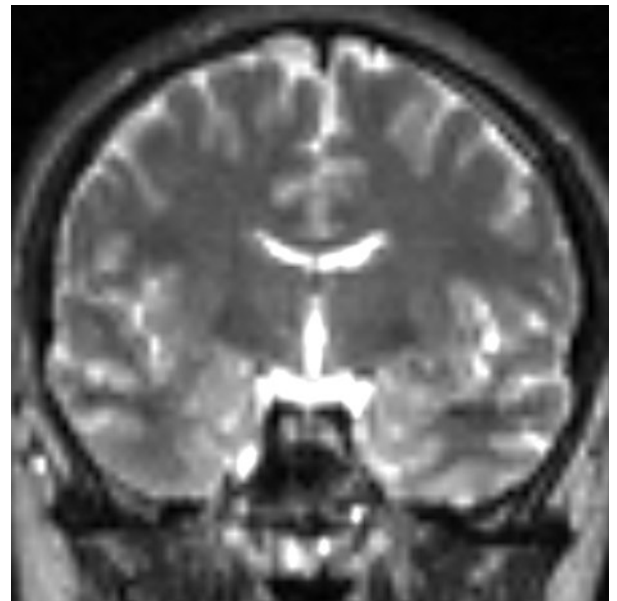
**Le LCR** Le liquide céphalo-rachidien est une substance limpide, ressemblant à de l'eau, produite au niveau des cavités du cerveau appelées ventricules. Il forme un coussin protecteur pour le système nerveux central et le nourrit. Il circule dans l'espace sous arachnoïdien, dans les sinus, au centre des sillons ainsi qu'autour de la moelle épinière (figure 2.6). En temps normal, il est continuellement absorbé et réapprovisionné, de manière à ce que sa quantité et sa pression soient maintenues constantes.

**Régions fonctionnelles** Le cortex peut être décomposé en régions ayant des fonctions spécifiques bien localisées. Certaines de ces régions sont, par exemple, consacrées au traitement du langage, d'autres à la vision, et d'autres encore aux perceptions sensorielles. La figure 2.9 donne un exemple de la décomposition des zones sensorielles. L'IRM fonctionnelle (IRMf), qui est une technique d'IRM rapide de basse résolution évoquée précédemment (§ 2.1.1), permet de cartographier les régions cérébrales. Lorsqu'un sujet effectue une tâche (cognitive, par exemple), l'irrigation augmente dans les zones cérébrales impliquées. Ce changement d'irrigation, appelé réponse hémodynamique, est mesurable en IRM. Le sujet alterne deux tâches (paradigme) pendant l'acquisition des images. Des techniques de traitement du signal permettent alors de retrouver un signal en crêteaux dans les zones d'activation.

<sup>15</sup>Nous emploierons aussi, indifféremment, les termes plis corticaux et sillons corticaux.



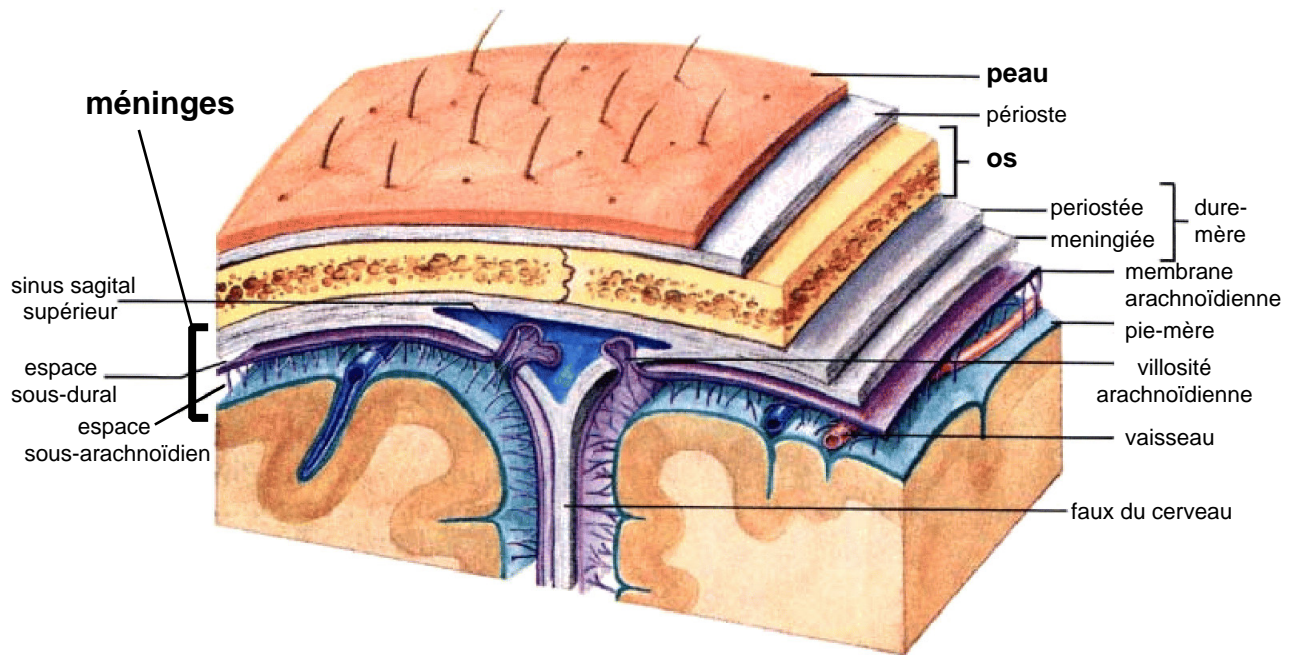
(b)



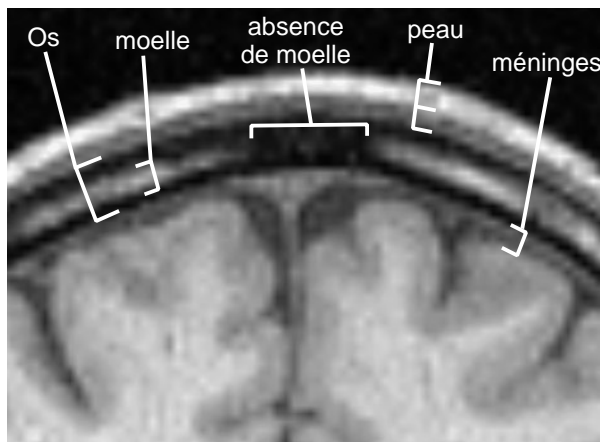
(c)

FIG. 2.7 – Principales structures du cerveau. (a) Le cerveau comporte essentiellement deux types de tissus : la matière blanche et la matière grise. Le liquide céphalo-rachidien (LCR) dans lequel baigne le cerveau remplit aussi des cavités dénommées ventricules. La matière grise se répartie en deux types de structures : les cortex (ou plis corticaux) et les noyaux gris. (D'après Marieb [69], modifiée) (b) Sur cette image IRM  $T_1$  de résolution 1 mm, on peut distinguer les différentes structures. (c) Sur des images  $T_2$ , (résolution : 2 mm) les contrastes sont inversés et la matière grise n'apparaît presque pas.





(a)



(b)



(c)

FIG. 2.8 – (a) Couches externes au cerveau. Les méninges sont constituées de trois membranes entourant le cerveau : la pie-mère, directement au contact du cerveau, la membrane arachnoïdienne et la dure-mère. A l'extérieur des méninges on trouve l'os et la peau. (Illustration d'après Marieb [69], modifiée) (b,c) Les couches externes sont difficiles à visualiser en IRM. (b) En haute résolution (1 mm) seul la moelle de l'os crânien produit un signal, les portions d'os sans moelle peuvent donc disparaître sur l'image. (c) En basse résolution (2 mm), il est très difficile de distinguer les différents éléments.

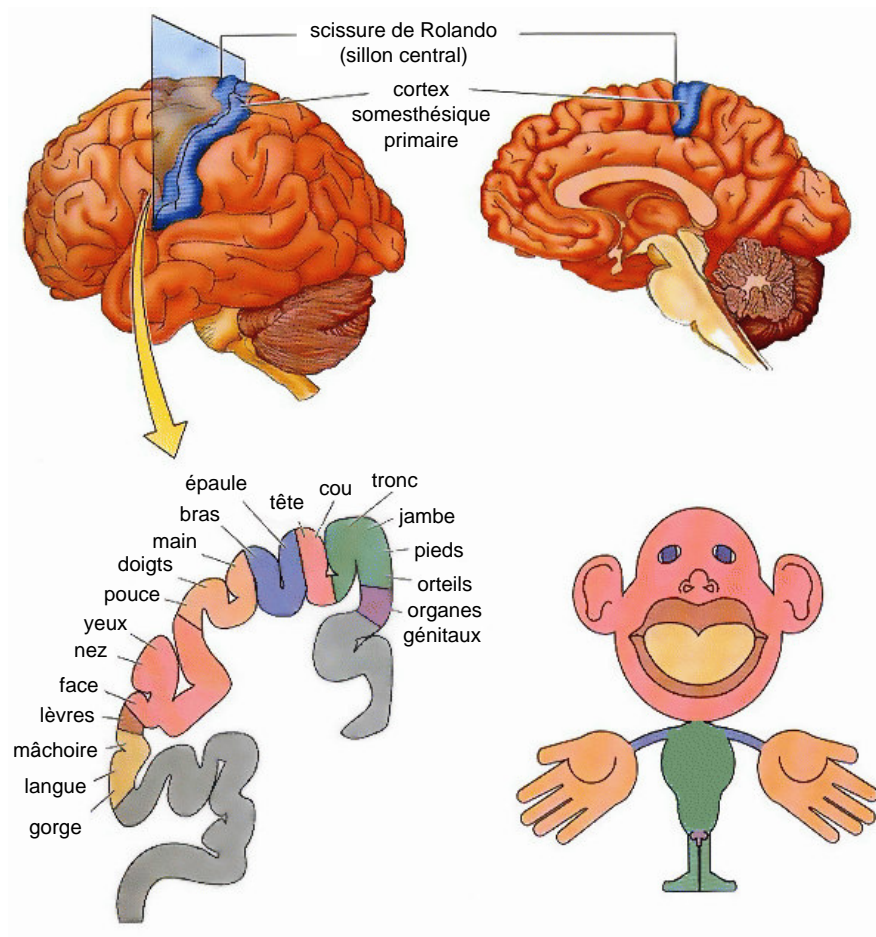


FIG. 2.9 – Le cortex peut être décomposé en régions ayant des fonctions spécifiques bien localisées. Cette figure décrit les zones sensorielles du cortex. Chaque région du corps correspond à une zone du cortex sensoriel. La correspondance entre la taille des zones sensorielles du cortex et les régions du corps est représentée schématiquement par le personnage en bas à droite. (source : driesen.com)



## 2.3 Pathologies cérébrales et leurs évolutions

Nous allons présenter brièvement certaines pathologies cérébrales, leurs évolutions et leurs manifestations en IRM. L'objectif de cette présentation est de donner au lecteur non spécialiste un aperçu des applications cliniques de l'IRM cérébrale et de situer notre travail par rapport à ces applications. Dans un deuxième temps, nous présenterons de manière plus détaillée la sclérose en plaques qui a servi de cadre d'évaluation aux techniques de détection de changements développées au cours de cette thèse.

### 2.3.1 Présentation des pathologies cérébrales

Les pathologies cérébrales peuvent être classées en six grandes catégories : les pathologies expansives, les pathologies vasculaires cérébrales, les pathologies dues à un traumatisme (accident), les pathologies démyélinisantes et inflammatoires les pathologies infectieuses et les autres pathologies telles que les pathologies dégénératives (maladie d'Alzheimer, maladie de Parkinson). Nous analyserons brièvement et successivement chacune de ces catégories, fournissant de nombreuses illustrations pour aider le lecteur à comprendre les problématiques de traitement d'image pouvant apparaître. Les illustrations proviennent d'un ensemble de sources disponibles sur Internet [57][49][25], souvent à vocation pédagogique, qui proposent des bases de données d'images.

**Pathologies expansives** Ces pathologies regroupent essentiellement les tumeurs. Leurs symptômes cliniques sont liées à la compression des tissus cérébraux avoisinants par la tumeur et par l'œdème l'entourant. Les tumeurs sont des amas de cellules qui prolifèrent de manière incontrôlée. On distingue les tumeurs bénignes des tumeurs malignes. Les premières ont une frontière bien délimitée et se développent localement. Les deuxièmes, les tumeurs malignes, ont des frontières mal délimitées et peuvent s'étendre dans les tissus environnants et éventuellement à d'autres sites (métastases). On dénombre, parmi les tumeurs courantes, les méningiomes, les gliomes et les métastases cérébrales. Les méningiomes sont des tumeurs bénignes, développées à partir des méninges (figure 2.10-e,f). Les gliomes sont des tumeurs développées à partir des cellules de la glie (figure 2.10-a,b). Les métastases cérébrales sont des localisations secondaires de tumeurs primitives développées au dépend d'autres organes. Les tumeurs cérébrales sont responsables de 2% des morts par cancer et ce quel que soit l'âge (10% de la mortalité par cancer pour les patients âgés entre 10 et 35 ans).

L'IRM permet de localiser une tumeur, de déterminer sa taille, et ses rapports avec les tissus avoisinants (effet de masse). Elle permet parfois de préjuger du type de tumeur. Une tumeur maligne sera extraite par exérèse<sup>16</sup>, si les risques liés à son extraction ne s'avèrent pas trop importants. Dans le cas contraire, le suivi par IRM peut être intéressant, par exemple, pour déterminer la réponse aux traitements (chimiothérapie, radiothérapie, ...). En IRM, l'évolution de la tumeur peut se manifester par un changement de sa taille, mais aussi par des changements d'intensité en son sein. Si la détection de changements peut être utile ici, le potentiel de notre approche n'est pas pleinement exploité. En effet, les tumeurs sont localisées sans difficulté par le neurologue et des techniques plus spécifiques mesurant les expansions / contractions locales sont bien adaptées à cette application. Habituellement les tumeurs sont visualisées par l'IRM, cependant des tumeurs de petites tailles telles que des micrométastases

---

<sup>16</sup>Exérèse : biopsie enlevant la totalité d'une lésion avec des berges en tissu sain, afin de faire simultanément le diagnostic anatomo-pathologique et le traitement radical.

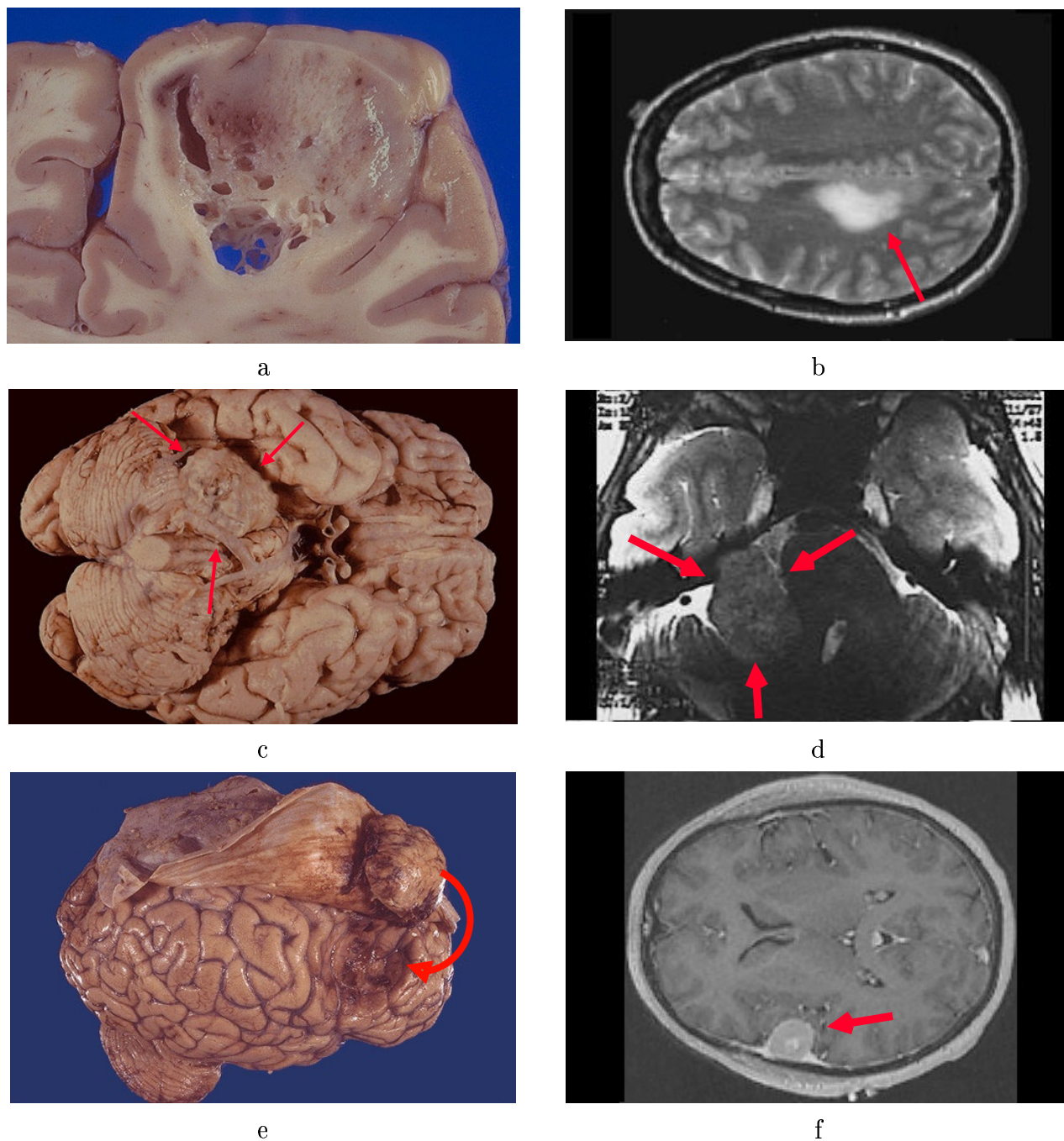


FIG. 2.10 – Trois exemples de tumeurs cérébrales. À gauche, des vues histologiques et des pièces anatomiques présentant des tumeurs, à droite, des vues en IRM. (a) Coupe d'un gliome. (b) IRM. Les bords de cette tumeur maligne sont diffus. (c) Neurinome de l'acoustique. (d) IRM d'un neurinome (e) Un méningiome, collé aux méninges, est soulevé, dévoilant une compression d'un hémisphère cérébral. (f) IRM. Notez les bords très nets de cette tumeur. (source : Web-Path [57], modifiée)

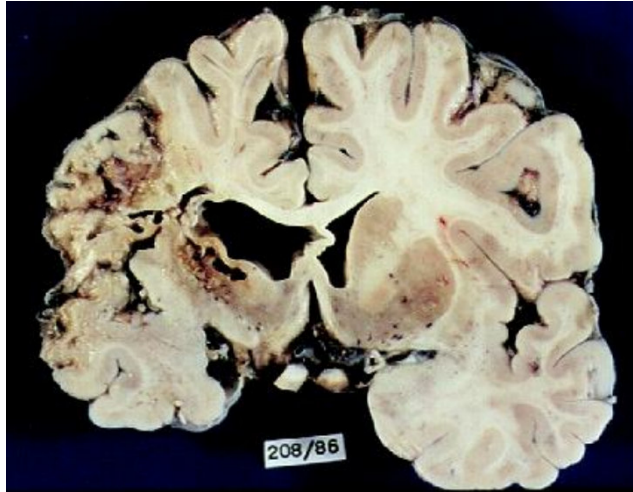


FIG. 2.11 – Coupe frontale d'une cerveau affecté par un infarctus de l'artère sylvienne droite. (source : V.Wolff, CHU-Strasbourg).

peuvent passer inaperçues à un stade très précoce. Dans cette indication notre technique de détection pourrait avoir un intérêt diagnostique mais aussi thérapeutique.

**Pathologies vasculaires cérébrales** Les maladies vasculaires cérébrales représentent la troisième cause de mortalité<sup>17</sup> des pays riches et la première cause d'handicap moteur [104]. Elles touchent une personne sur 1 000 avec 150 000 nouveaux cas par an. Elles se manifestent, le plus souvent, par des épisodes de déficit neurologique, appelés « attaques » ou accidents vasculaires cérébraux, dus à des anomalies dans l'irrigation du cerveau. Parmi ces anomalies, on distingue les infarctus (blocage de la circulation sanguine, 85% des attaques, voir figure 2.11) et les hémorragies (15%). Privé d'oxygène, le tissu cérébral est détruit en quelques minutes voire quelques heures. Nous décrivons ci-dessous, successivement, les deux catégories d'infarctus : les ischémies artérielles et les ischémies veineuses.

Pour les infarctus artériels à la phase aiguë, l'IRM est plus sensible que le scanner-X, dès les premières heures notamment grâce à la séquence de diffusion (voir figure 2.12). Les ischémies artérielles donnent rarement lieu à des examens IRM de suivi (figure 2.13), sauf apparition de nouveaux signes cliniques, pour déceler une nouvelle ischémie. L'intérêt d'un système de détection comme le nôtre, est donc limité dans le cas des ischémies artérielles.

Les thromboses veineuses (figure 2.14) sont caractérisées par un grand polymorphisme clinique. L'IRM cérébrale avec l'IRM angiographique en séquences veineuses (VRM) est actuellement l'examen de référence. Elle permet non seulement de visualiser le thrombus dans le sinus mais aussi de voir les conséquences de cette occlusion et de celles des veines corticales, avec parfois la présence d'un infarctus veineux. L'IRM est utilisée dans le but d'établir un diagnostic, mais aussi pour le suivi après traitement par des anticoagulants. La thrombose des veines corticales est plus difficile à visualiser du fait du petit calibre des veines, de leur nombre et de leur topographie variables. Un système de détection de changements pourrait assister le neurologue dans cette tâche

**Traumatismes** Les traumatismes du système nerveux sont une des principales causes de handicap et de mortalité chez l'homme jeune. Ils sont provoqués majoritairement par des acci-

<sup>17</sup>Après les maladies cardiovasculaires et les cancers.

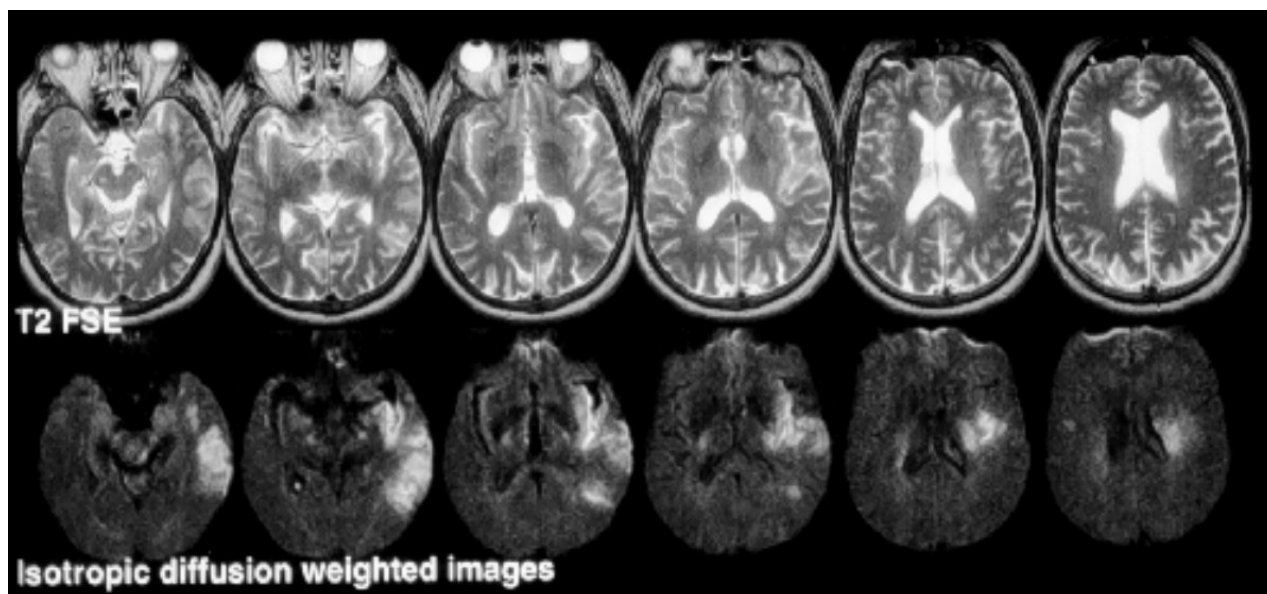


FIG. 2.12 – *Infarctus de l'artère sylvienne gauche à la phase aiguë. L'infarctus n'est pas visible en  $T_2$  mais apparaît clairement en IRM de diffusion. Les images présentent une série de coupes à des hauteurs successives (source : V. Wolff CHU-Strasbourg)*

dents de la route et des chutes. Deux types de pathologies peuvent être distingués : les conséquences immédiates du traumatisme (lésions, contusions, hémorragies) et ses conséquences à plus long terme (œdèmes, saignements, hypoxie). Les traumatismes ne sont pas toujours visibles en imagerie (que ce soit en IRM ou en scanner-X), 50% des patients dans le coma présentent des IRM et scanner-X normaux. L'IRM permet parfois de visualiser les conséquences d'un traumatisme plus facilement que le scanner-X.

**Pathologies démyélinisantes et inflammatoires** Dans cette catégorie nous nous intéresserons à la sclérose en plaques (SEP), une pathologie démyélinisante décrite de manière plus détaillée au paragraphe 2.3.2. Notre système de détection de changements a été appliqué au suivi de lésions de SEP et évalué dans ce cadre. Nous avons ainsi démontré l'intérêt de la détection de changements en tant qu'outil permettant d'assister le neurologue dans la localisation d'évolution de lésions.

**Pathologies infectieuses** Elles comprennent les encéphalites, les méningites et les abcès. Les encéphalites sont une inflammation, sans suppuration (production de pus), d'une partie plus ou moins étendue de l'encéphale, ayant à la fois des causes infectieuses et non infectieuses. Les encéphalites infectieuses sont souvent d'origine bactérienne (méningo-encéphalites, syphilis, tuberculose) ou virale (herpès, rage), mais peuvent aussi être provoquées par des mycoses et des parasites. Les encéphalites se présentent, en IRM, comme un hypersignal diffus localisé sur une grande région (figure 2.15-a). La maladie de Creutzfeld-Jakob est aussi considérée comme une encéphalite (figure 2.16). Les encéphalites non infectieuses sont souvent liées à d'autres pathologies (tumeur, accident vasculaire cérébral, SEP ...), mais peuvent aussi être dues à d'autres facteurs spécifiques comme l'intoxication au plomb (saturnisme) et la réaction à certains vaccins. Les méningites sont une inflammation des méninges, et les abcès sont des infections localisées présentant un contour très net en IRM-T1 (voir figure 2.15-b).

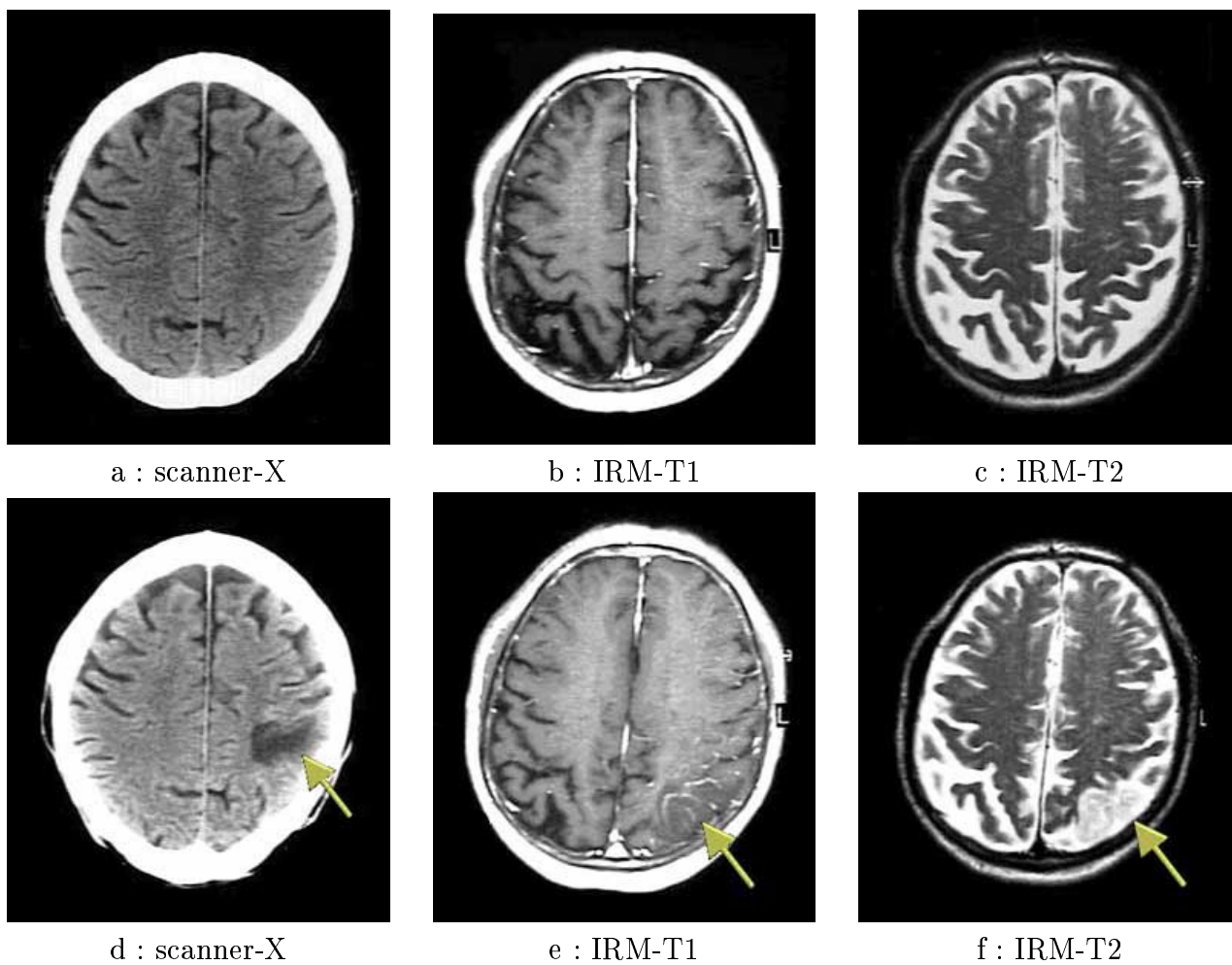


FIG. 2.13 – Évolution d'un infarctus : comparaison entre l'IRM (b,c,e,f) et le scanner-X (CT) (a,d). Dans un premier examen, en phase aiguë sur (a,b,c), l'infarctus n'est pas visible. Lors du deuxième examen (d,e,f), un mois plus tard, l'infarctus pariétal gauche est clairement visible au scanner-X et à l'IRM. Les images T1 sont prises avec un produit de contraste. (source : strokecenter.org)

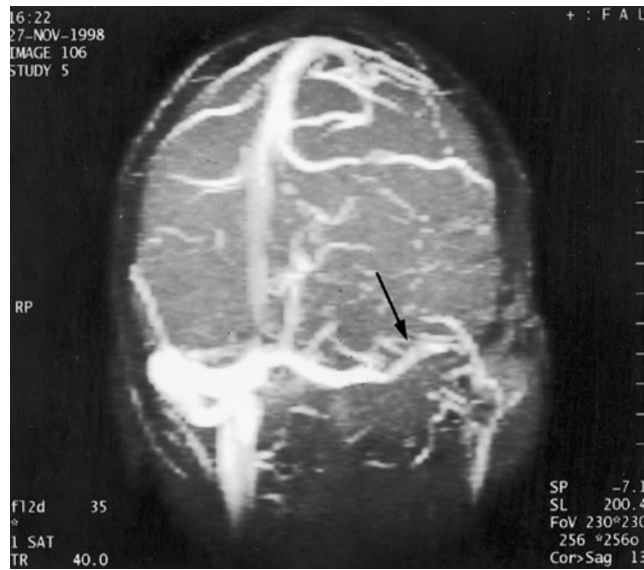


FIG. 2.14 – *IRM angiographique avec une séquence permettant de visualiser les veines. Une thrombose veineuse cérébrale du sinus latéral gauche est indiquée par la flèche. (source : V. Wolff, CHU-Strasbourg)*

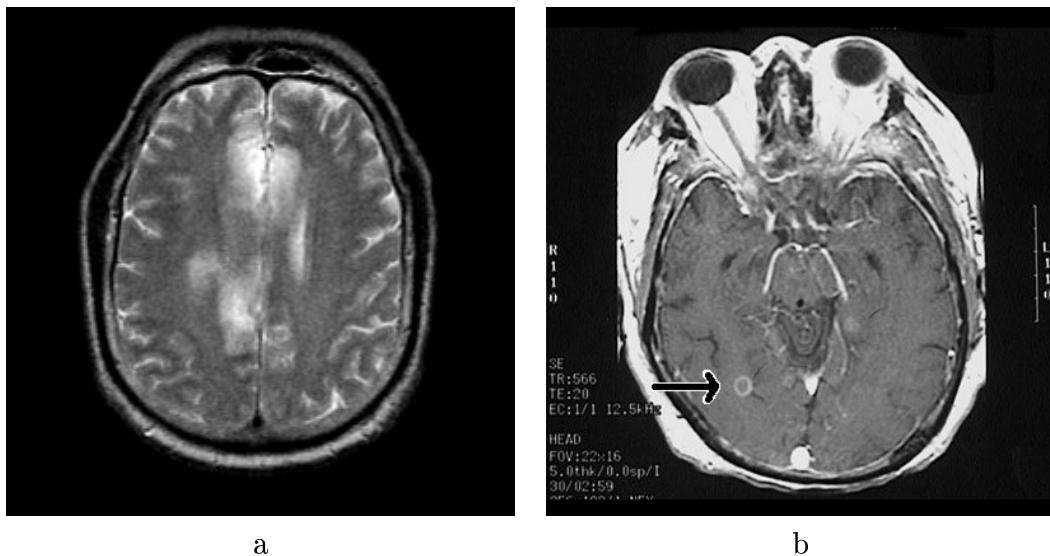


FIG. 2.15 – *Infections du cerveau : (a) L'herpès peut provoquer des encéphalites qui sont clairement visibles en IRM. Ces inflammations affectent une région entière du cerveau. (b) Les abcès sont, quant à eux, des infections très localisées entourées d'un contour brillant en IRM-T1*



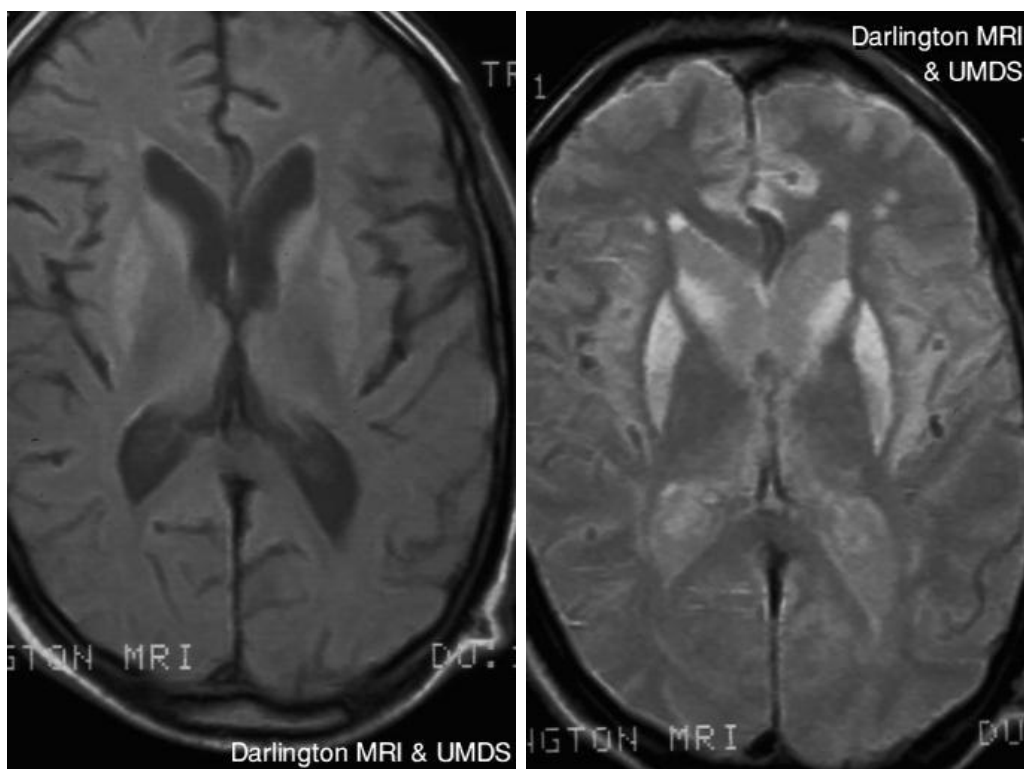


FIG. 2.16 – La maladie de Creutzfeld-Jakob est une maladie infectieuse très particulière dont l'agent infectieux est une protéine (PrPc) ayant une conformation anormale. La maladie se présente en IRM par un hypersignal (ici dans le putamen et les noyaux caudés). (source : SRS-X [25])

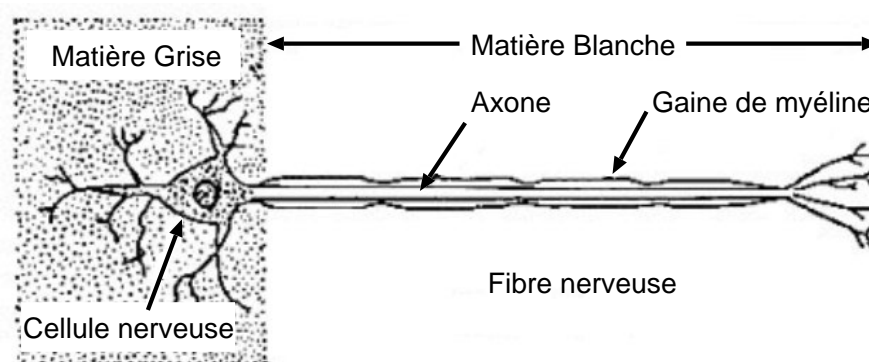


FIG. 2.17 – Structure schématique simplifiée d'un neurone. Dans la sclérose en plaques, la gaine de myéline entourant les axones peut être attaquée par le système immunitaire, provoquant des inflammations et des lésions (voir figure 2.18).

**Autres Pathologies** Les maladies neurodégénératives, comme Alzheimer ou Parkinson sont souvent associées à une atrophie cérébrale importante, distinctement visible en IRM. Dans le cas d'Alzheimer, le cerveau peut perdre jusqu'à 50% de son poids. Des techniques de détection de changements d'intensité, comme l'approche que nous proposons, ne sont pas adaptées à la mesure de ces atrophies globales. Cependant, d'autres approches spécifiques permettent de les mesurer avec précision, avec des applications aussi en SEP. Ces aspects seront décrits ultérieurement (§ 6.4.1, p. 112).

### 2.3.2 La sclérose en plaques

Nous présentons ici les principales caractéristiques de la sclérose en plaques (SEP). Des ouvrages [71] et de nombreuses ressources en ligne<sup>18,19,20</sup>, très complètes, fournissent à la fois une information pédagogique et des liens vers des documents plus précis, permettant au lecteur intéressé d'approfondir ses connaissances sur le sujet.

**Présentation de la pathologie** La sclérose en plaques touche environ 2,5 millions de personnes (1 personne sur 1 000 en Europe), affectant plutôt les femmes (50% de plus). Dans 70% des cas les premiers symptômes apparaissent entre 20 et 40 ans. C'est une maladie, incurable à ce jour, qui peut être fortement incapacitante, mais qui ne modifie pas significativement l'espérance de vie des malades. Il existe, aujourd'hui, des traitements permettant d'amoinrir les symptômes, même si les effets à long terme sur la progression de la pathologie restent sujet à débat.

**Causes** Les causes exactes de la SEP sont encore incertaines. Des réactions auto-immunitaires détruisant les gaines de myéline (démýélinisation) entourant les nerfs (figure 2.17) sont impliquées dans le processus pathologique. Les signaux circulant sur ces gaines sont alors ralentis voire bloqués. Les inflammations forment des lésions localisées qui se durcissent

<sup>18</sup><http://www.msif.org>

<sup>19</sup><http://www.nationalmssociety.org>

<sup>20</sup><http://www.lfsep.asso.fr>



(sclérosent) progressivement. Les lésions de SEP peuvent avoir des tailles très variables (voir figure 2.18, voir ci-dessous pour des exemples de lésions en IRM).

**Symptômes** La SEP se manifeste souvent par des poussées correspondant à des épisodes de démyélinisation. Les poussées sont espacées de manière variable dans le temps. Entre chaque poussée, le patient peut récupérer tout ou partie de ses capacités perdues. Les signes cliniques sont très variables et dépendent de la localisation de la démyélinisation dans le système nerveux, en particulier, si les lésions sont localisées dans la moelle épinière. Parmi les exemples de symptômes, on peut citer entre autres : une fatigue anormale, des troubles aiguës de la vue, des problèmes de coordination et d'équilibre, des troubles de l'élocution, des tremblements, des dysfonctionnements de la vessie, etc. L'évolution de la sévérité des symptômes au cours du temps permet de classer la SEP en quatre catégories [67] : rémittante (Relapsing Remitting), progressive secondaire (Secondary Progressive), rémittante progressive (Progressive Relapsing) et progressive primaire. Cette évolution est illustrée schématiquement à la figure 2.19.

**Le diagnostic** Le diagnostic de la SEP n'est pas immédiat. Les manifestations cliniques peuvent être confondues avec celles d'autres pathologies cérébrales. Les lésions apparaissant en IRM ne permettent pas non plus, à elles seules, d'écarter d'autres pathologies. Par ailleurs, environ 5% des patients atteints de SEP n'ont pas de lésions visibles en IRM (même si, à terme, ces lésions seront visibles dans la vaste majorité des cas). Pour systématiser le diagnostic, un protocole précis a été élaboré par un panel international [72] proposant un ensemble de critères (signes cliniques, lésions IRM avec ou sans produit de contraste, analyses sur des prélèvements de LCR, critères temporels, localisation des lésions) qui, dans certaines combinaisons, permettent de diagnostiquer la SEP. Ce protocole a confirmé l'importance du rôle de l'IRM dans le diagnostic de la SEP.

**Les traitements** S'il n'existe pas de traitements permettant de guérir la SEP, en revanche, certains traitements permettent de modifier le cours de la maladie, en amoindrissant la fréquence et la sévérité des poussées. Le choix du type de traitement se fait suivant la catégorie de l'évolution (figure 2.19) de la maladie. L'efficacité de certains de ces traitements reste controversée : leurs effets secondaires étant parfois mal tolérés et leur coût important (entre €10 000 à €15 000 par an). Les poussées font l'objet d'un traitement spécifique par de fortes doses de cortico-stéroïdes. Ce traitement permet de raccourcir la durée de la poussée et d'atténuer ses effets.

### La sclérose en plaques en IRM

Les protocoles standardisés ont confirmé l'importance de l'IRM dans le diagnostic et le suivi de la SEP [72]. Différentes séquences (modalités) peuvent être utilisées pour les visualiser [115]. Des séquences comme la FLAIR, présentent l'avantage d'avoir des temps d'acquisition plus courts et d'annuler le signal du LCR, permettant ainsi de mieux distinguer les lésions périventriculaires (voir figures 2.20 et 2.21). L'utilisation d'un produit de contraste paramagnétique (gadolinium) permet de mettre en évidence les lésions actives.

Si l'IRM permet d'observer les lésions, il existe une faible corrélation entre la charge lésionnelle totale et les signes cliniques (sauf pour les lésions de la moelle épinière pour lesquelles la corrélation entre la charge lésionnelle et le handicap est forte). Cette décorrélation entre la charge lésionnelle et les signes cliniques semble indiquer que la matière blanche, apparemment

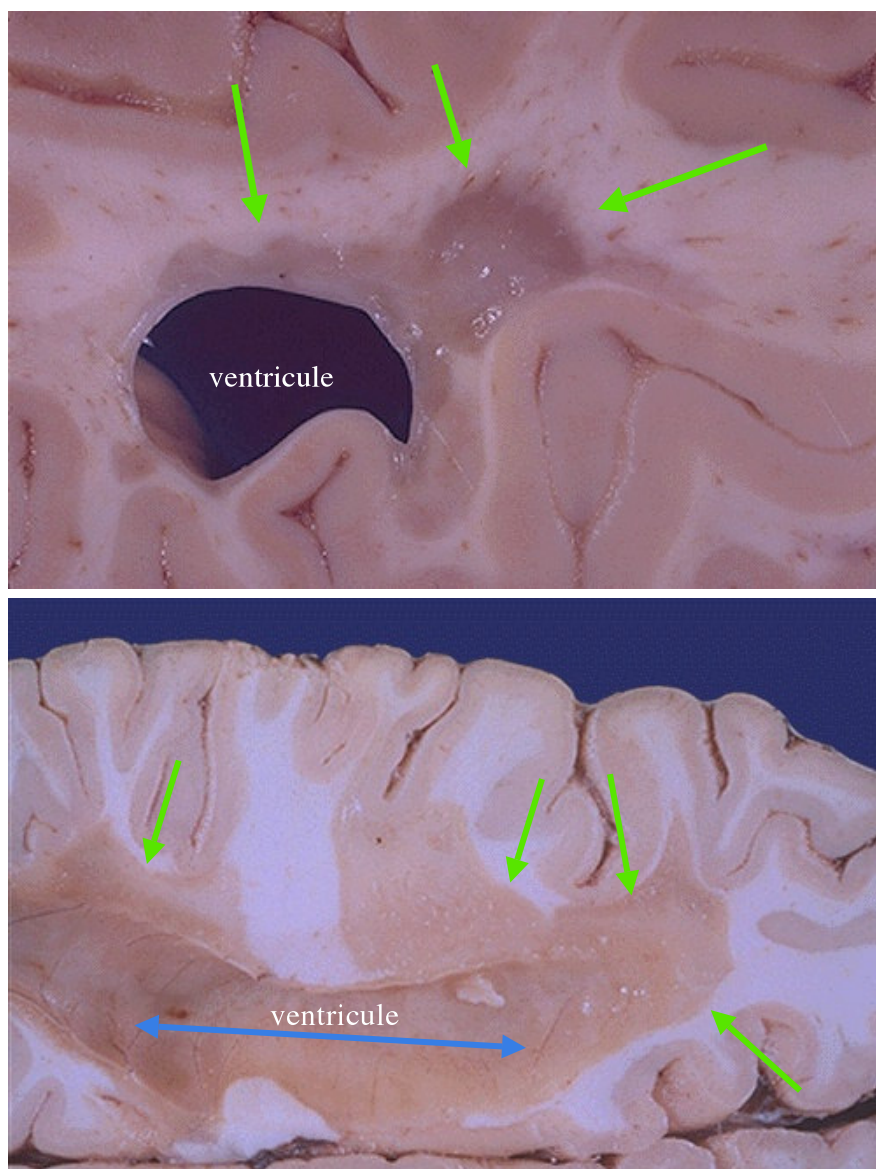


FIG. 2.18 – Coupes histologiques de grandes lésions (flèches vertes) périventriculaires de sclérose en plaques. Après un épisode de démyélinisation certaines lésions se durcissent (se sclérosent) (source : WebPath[57], modifiée).

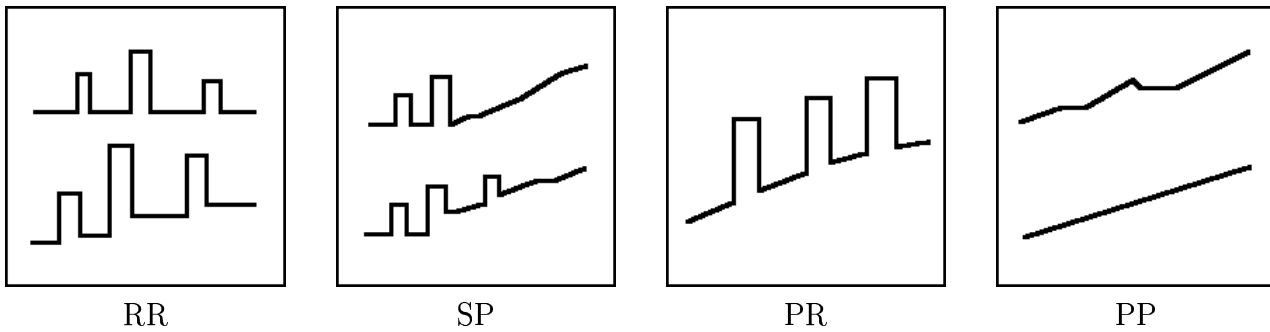


FIG. 2.19 – Évolution clinique schématisée des quatre formes de sclérose en plaques : *RR* rémittante, *SP* progressive secondaire, *PR* progressive rémittante, *PP* progressive primaire. Les pics représentent des poussées pouvant être suivies d'une rémission totale ou partielle. (source : *mult-sclerosis.org*)

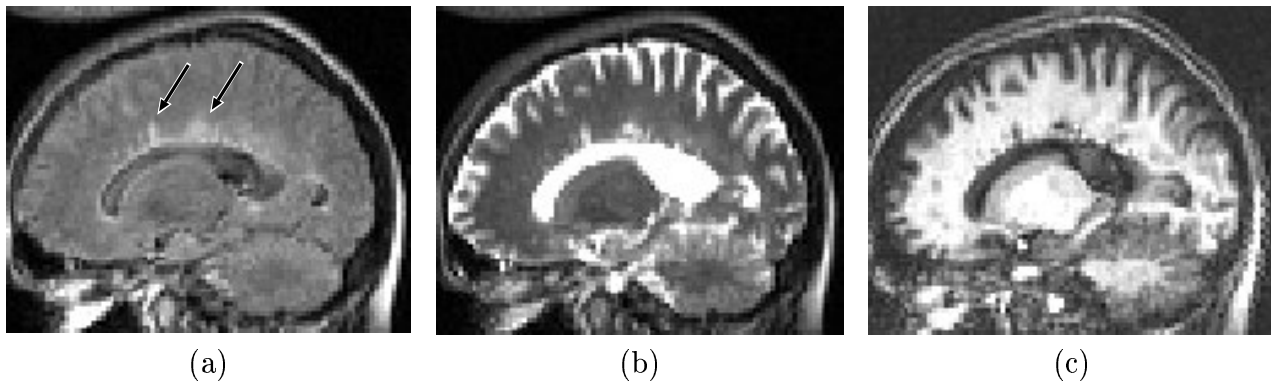


FIG. 2.20 – Lésions périventriculaires de sclérose en plaques dans trois modalités IRM différentes : (a) *FLAIR*, (b) *RARE* ( $T_2$ ), (c) *GE3D* ( $T_1$ ). Le contraste du liquide est annulé en *FLAIR*, permettant ainsi de bien distinguer les lésions des ventricules, essentiellement constitués de liquide. En *RARE* (centre) et *GE3D* (droite) les lésions ont le même contraste que le liquide et sont plus difficiles à distinguer, et ce, d'autant plus que les lésions sont petites et proches des ventricules.

normale pourrait être affectée [50]. D'autres techniques d'imagerie comme l'imagerie par transfert d'aimantation, l'imagerie de diffusion, et la spectroscopie par résonance magnétique (voir § 2.1.1, p. 17) sont utilisées actuellement pour explorer cette hypothèse [115].

Les manifestations de la SEP en IRM ne se limitent pas aux seules lésions. Lorsque la démyélinisation est complète, elle peut conduire à la destruction de l'axone. Ces lésions destructrices sont appelées trous noirs et ne sont pas rehaussées par le produit de contraste. La perte neuronale conduit à long terme à une atrophie généralisée du cerveau qui peut être observée en IRM et quantifiée à l'aide de techniques de traitement d'images [102] (voir aussi § 6.4.1, p. 112).

**Description et suivi de lésions de SEP** Les lésions de SEP ont des tailles et des formes très variées. Certaines lésions sont adjacentes aux ventricules (périventriculaires, figure 2.21). La majorité est répartie dans la matière blanche (figure 2.24), parfois près des sillons corticaux (figure 2.23). Certaines lésions de petite taille sont à la limite de la résolution d'acquisition (figure 2.22). En IRM, l'évolution des petites lésions se manifeste d'abord par une apparition brusque (hyper-signal), qui s'estompe ensuite progressivement.

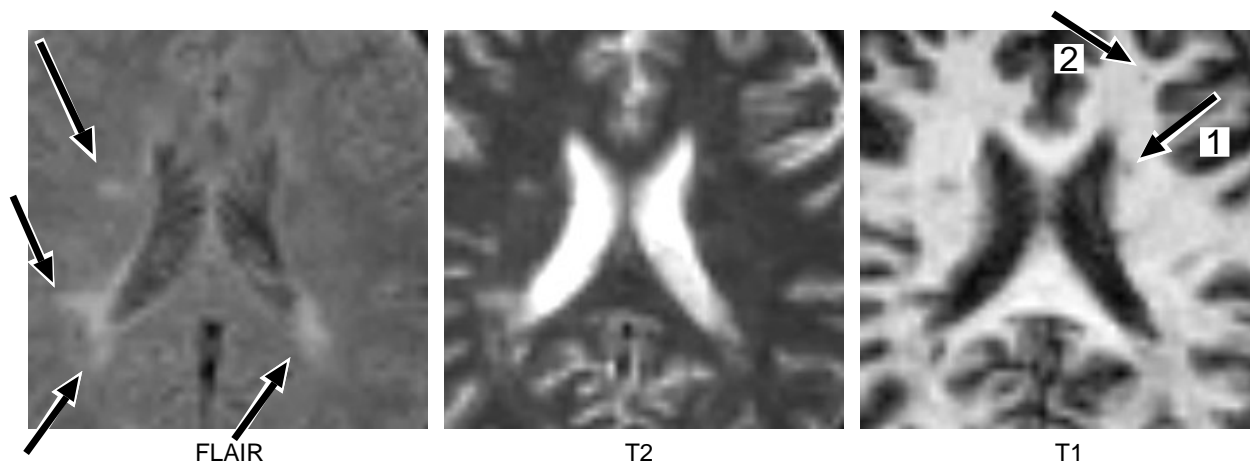


FIG. 2.21 – Agrandissement sur des lésions de sclérose en plaques en trois modalités IRM différentes. Plusieurs tailles, formes et localisations de lésions sont visibles. Les lésions 1 et 2 sont montrées en agrandissement à la figure 2.22. Ces deux lésions sont difficiles à voir en FLAIR, montrant la complémentarité de ces modalités.

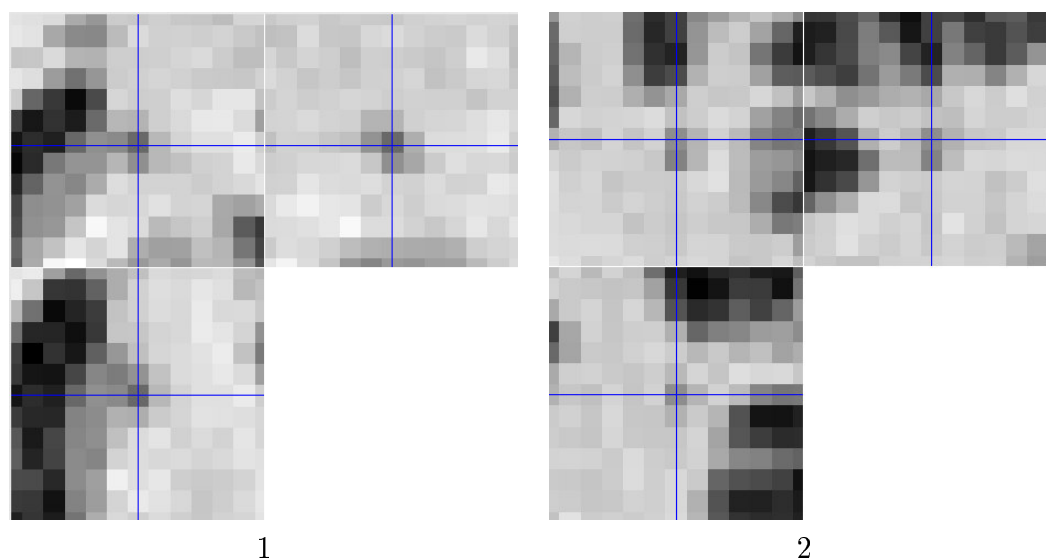


FIG. 2.22 – Trois coupes, fortement agrandies, au travers des lésions 1 et 2 de la figure 2.21. Les images sont affichées sans interpolation, pour mettre en évidence la taille des voxels. La lésion 1, d'environ 8 voxels, est typique des lésions dont l'évolution sera suivie dans ce travail. La lésion 2, plus petite, ne fait qu'environ 2 voxels (1 voxel=8 mm<sup>3</sup>).

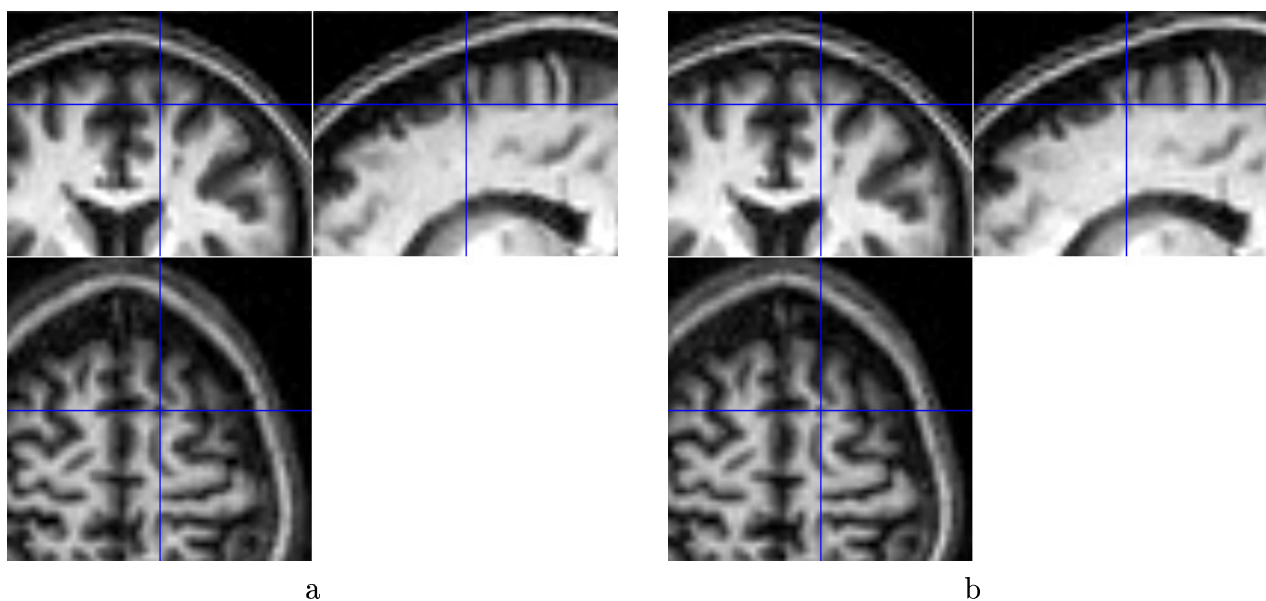


FIG. 2.23 – Évolution d'une lésion de SEP proche des sillons (IRM-T1). Les images sont prises à deux mois d'intervalle et ont été recalées et normalisées (voir partie III). Ce type de lésion est difficile à repérer, dans cette modalité (T1), de par sa ressemblance à un sillon. Pour vérifier qu'il s'agit bien d'une lésion, on peut parcourir le volume l'entourant et constater que la tache sombre n'est pas rattachée aux sillons. Cette évolution représente une modification relativement importante qui sera aisément détectée par notre système automatique.

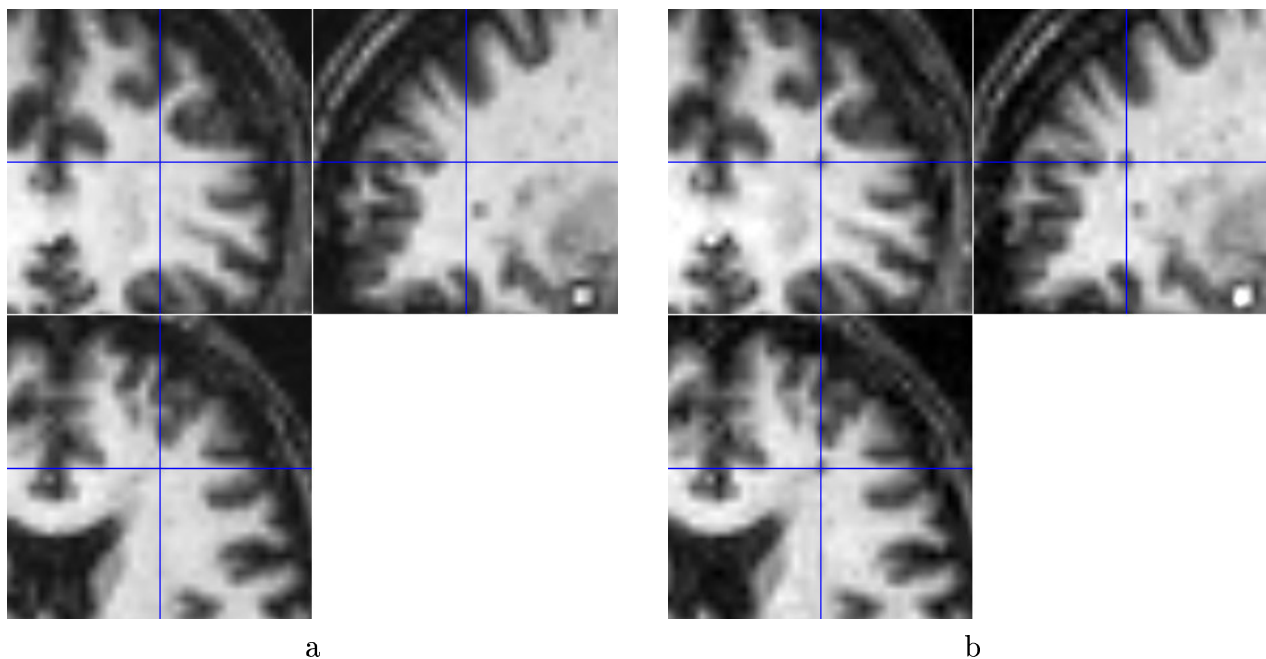


FIG. 2.24 – Évolution (apparition) d'une lésion de SEP. Cette lésion est assez petite (4 voxels) mais est très contrastée, ce qui est fréquent pour des nouvelles lésions (actives).

Deuxième partie

Segmentation du cortex



# Chapitre 3

## Problématique, état de l'art

Avant d'aborder la détection de changements, nous allons décrire, dans cette deuxième partie du mémoire, une approche de segmentation des sillons corticaux. Comme cela a été décrit en introduction (§ 1.1, p. 11), la segmentation de sillons corticaux apportera des connaissances anatomiques à la détection, permettant d'améliorer ses performances (comme l'attestent les résultats présentés § 8.4.2, p. 136). En effet, les lésions de sclérose en plaques, dont on souhaite détecter l'évolution, apparaissent dans la matière blanche (et éventuellement dans la matière grise). Les plis corticaux et la matière blanche environnante constituent une partie importante du volume du cerveau. Leur segmentation permettra donc de délimiter les régions où il est possible d'observer des évolutions de lésions et d'éliminer les détections dans les régions ne pouvant pas les contenir.

### 3.1 Introduction à la segmentation des structures cérébrales

Segmenter une image, c'est construire une représentation sémantique de la réalité physique à partir de cette image. Plus spécifiquement, une segmentation d'un cerveau permet d'associer une structure anatomique à chaque position de l'espace. La définition exacte de ce qui est considéré comme une structure anatomique varie selon les approches : dans certaines, il s'agira simplement de distinguer les tissus cérébraux en trois grandes classes (MB, MG, LCR), d'autres se spécialiseront sur des structures spécifiques (noyaux gris, corps calleux, ventricules, plis corticaux, etc. voir § 2.2.1, p. 24). La forme que prend l'association entre la structure anatomique et la position de l'espace dépend des approches considérées : certaines associeront un label (correspondant à une structure) à chaque point de l'image, d'autres détermineront des surfaces de séparation (explicites ou implicites) entre structures, et d'autres encore indiqueront des degrés d'appartenance à chaque structure en chaque point.

Nous commencerons cette introduction en présentant les applications de la segmentation cérébrale (§ 3.1.1). Nous en déduirons au paragraphe 3.1.2 les objectifs que peuvent se fixer les approches de segmentation. Puis, nous présenterons § 3.1.3 les spécificités de la segmentation d'images IRM cérébrales et les difficultés associées.

Dans un deuxième temps, au paragraphe 3.2, nous allons présenter le travail existant dans le domaine de la segmentation cérébrale. Nous nous concentrerons sur les méthodes permettant de segmenter les plis corticaux, puis nous décrirons brièvement, au paragraphe 3.2.3, les différents outils couramment employés en segmentation corticale.



### 3.1.1 Applications

Segmenter, ou délimiter, les structures cérébrales est une étape fondamentale pour l'analyse quantitative d'images cérébrales<sup>1</sup>. Les principales applications, que nous présentons ici, sont la morphométrie, la cartographie fonctionnelle, et la visualisation.

**Morphométrie** La mesure quantitative des positions, formes et tailles de structures cérébrales, appelée morphométrie, nécessite la segmentation préalable de ces structures. Elle peut permettre d'identifier des anomalies et de suivre la progression de pathologies<sup>2</sup>. La segmentation de l'enveloppe extérieure du cerveau permet de mesurer l'atrophie globale du cerveau associée à des pathologies, telle que la SEP (voir introduction sur la SEP au paragraphe 2.3.2, et aussi § 6.4.1, p. 112) et la maladie d'Alzheimer [22]. La segmentation des tissus cérébraux (MB/MG/LCR) permet de comparer leur volume total, et de suivre l'évolution de certaines pathologies ainsi que les phénomènes liés au vieillissement [39, 92, 63]. La segmentation de structures internes, comme le corps calleux, permet d'étudier leur atrophie due à la SEP [101]. La segmentation de l'hippocampe et l'analyse de son volume sont employées en épilepsie [34]. La segmentation précise du cortex permet l'analyse de son épaisseur, avec des applications en épilepsie [4]. Des mesures quantitatives sur des sujets sains ou malades ont aussi contribué à la compréhension de pathologies comme l'autisme, Alzheimer ou la schizophrénie (références dans [126]).

**Cartographie fonctionnelle** L'essentiel de l'activité fonctionnelle mesurable (IRMf, EEG, MEG) est situé dans le cortex dont la topologie est essentiellement bidimensionnelle (voir § 2.2, p. 23). La segmentation de la structure bidimensionnelle des plis corticaux est alors nécessaire pour localiser les signaux, pour les cartographier et pour les visualiser. La construction d'images fonctionnelles EEG et MEG, à partir des signaux électriques (EEG) ou magnétiques (MEG) nécessite de résoudre un problème inverse qui est fondamentalement mal posé. La segmentation du cortex à partir d'images anatomiques (IRM  $T_1$ ) du même patient fournit les connaissances *a priori* qui permettent de contraindre l'espace des solutions et de résoudre le problème inverse [13, 97]. En IRMf, l'activité mesurée sur des positions tridimensionnelles doit être cartographiée sur la surface des plis corticaux qui est une structure bidimensionnelle. Ceci nécessite de segmenter le cortex, de le déplier (gonfler en une sphère), et d'établir un repère de référence [27, 17](voir figure 3.1). Pour ce type d'applications, le résultat de la segmentation du cortex doit avoir une topologie bidimensionnelle [59].

**Visualisation** La visualisation de structures anatomiques (par exemple en chirurgie assistée [125], figure 3.2), se fait le plus naturellement à l'aide de la visualisation de leurs surfaces extérieures<sup>3</sup>. Il est alors nécessaire de segmenter ces structures. Une représentation précise et réaliste de la surface extérieure du cortex est souhaitable dans la préparation de la chirurgie cérébrale [42], en particulier, il est important de pouvoir y localiser visuellement les

<sup>1</sup>Nous recommandons l'article de Worth *et al*, 1997 [126] pour une présentation très intéressante des applications et objectifs de la segmentation cérébrale, même si les travaux récents n'y apparaissent pas.

<sup>2</sup>Néanmoins, l'identification de certains changements pathologiques peut se faire sans segmentation, à l'aide de techniques de détection de changements, comme celles présentées dans ce travail (troisième partie). Une quantification précise de la variation du volume d'une structure spécifique (bien délimitée) nécessite toutefois une segmentation préalable de cette structure.

<sup>3</sup>Des techniques de visualisation non surfaciques existent, mais sont nettement moins employées. Elles ne permettent souvent pas la visualisation en temps réel, ou exigent un matériel spécialisé.

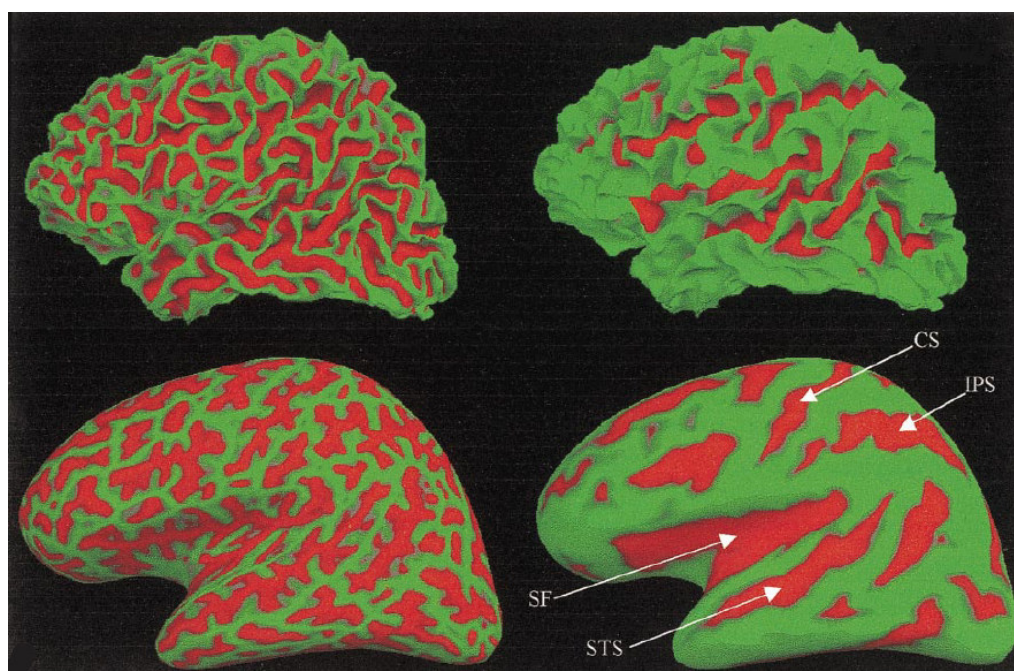


FIG. 3.1 – La segmentation du cortex permet de cartographier l'activité cérébrale sur la structure bidimensionnelle du cortex. Fischl et al. « gonflent » le cortex segmenté (en haut) comme un ballon (en bas) pour permettre la visualisation de l'activité située sur les plis profonds et pour établir une paramétrisation de la surface. (illustration d'après Fischl et al. [27])

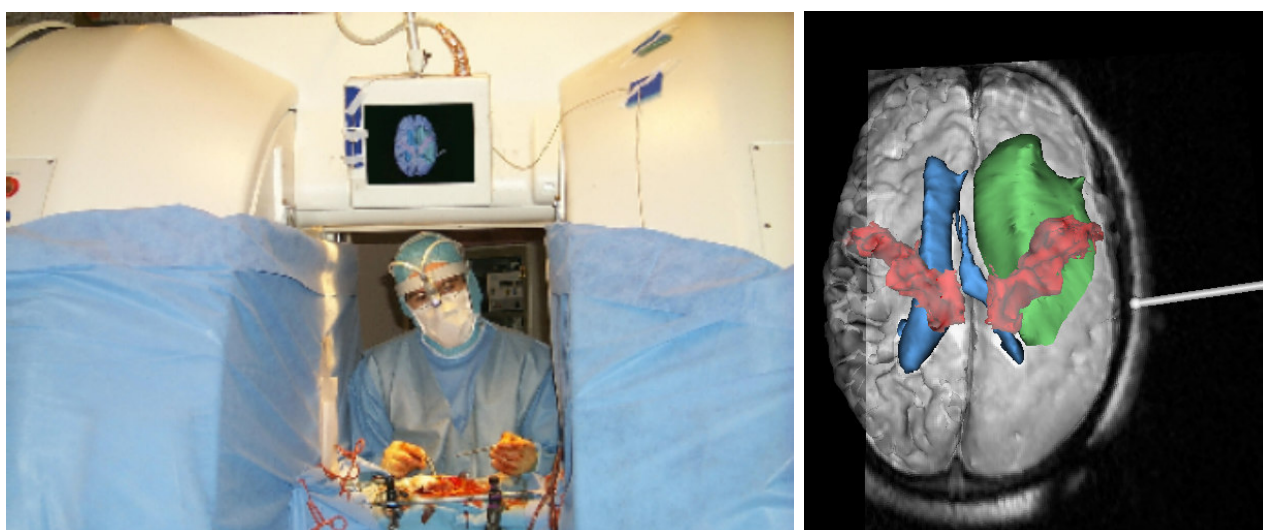


FIG. 3.2 – Une des applications de la segmentation de structures cérébrales est la visualisation. Cette illustration, tirée de Warfield et al. [125], montre un système de chirurgie assistée. L'acte opératoire se déroule au centre d'un appareil IRM et le chirurgien peut visualiser en temps réel les structures cérébrales segmentées.

zones fonctionnelles préalablement cartographiées pour planifier l'opération et éviter les zones fonctionnelles critiques.

**Autres applications** La segmentation des structures cérébrales permet d'obtenir une information sémantique qui est la base d'un grand nombre de traitements. Par exemple, dans notre utilisation, la segmentation du cortex permet de distinguer des évolutions de lésions de modifications parasites dans des structures, comme le LCR, qui ne peuvent pas contenir de lésions (voir § 8.4.2, p. 136). La segmentation permet aussi la construction d'atlas anatomiques. Une segmentation complète du cerveau peut être utilisée pour construire des simulateurs [10, 15] d'images IRM. Dans une autre application de notre laboratoire (IPB, UMR 7400), la segmentation des couches extérieures du cerveau et du crâne est utilisée pour fournir une initialisation pour la reconstruction de données de tomographie optique [113].

### 3.1.2 Objectifs et propriétés souhaitables pour la segmentation

Les applications décrites précédemment permettent de définir des objectifs et des propriétés souhaitables pour un système de segmentation cérébrale. Ces propriétés dépendent, bien entendu, de l'application envisagée. Nous ne développerons ici, que celles qui nous paraissent mériter des précisions particulières. D'autres, comme la précision des résultats ou la robustesse des traitements, nous paraissent suffisamment bien définies.

**Automatisation** L'automatisation complète est un objectif important. La segmentation purement manuelle d'un volume 3D est extrêmement laborieuse. Worth *et al.* [126] citent une étude sur la schizophrénie où des chercheurs ont segmenté 50 images par an et par chercheur travaillant à temps plein. Dans notre laboratoire nous avons constaté que la segmentation manuelle détaillée des structures internes, sur une seule image, peut prendre deux semaines à un anatomiste. Des méthodes assistées [59, 42, 97] permettent d'atteindre différents degrés d'automatisation. Dans [59], la correction d'erreurs topologiques, après la segmentation automatique du cortex, nécessite 30 mn. d'interaction par image. Dans [42], la segmentation semi-automatique de l'enveloppe extérieure du cerveau prend un peu moins d'une heure. Dans tous les cas, l'intervention humaine reste une source potentielle d'erreurs : au cours de longues séances d'interaction, l'opérateur humain change son comportement. La fatigue le fera commettre des erreurs, et en même temps l'habitude lui permettra d'aller plus vite. Le résultat est un manque de fiabilité et de reproductibilité<sup>4</sup>. Un autre point à considérer dans l'automatisation, est le réglage de paramètres ou de seuils nécessaires au fonctionnement des méthodes. Ces paramètres sont parfois difficiles à évaluer par un utilisateur non spécialiste et peuvent requérir une expérimentation comprenant des phases d'essai, parfois longues. Si la segmentation manuelle ou semi-automatique peut être acceptable dans certaines applications, comme pour la création d'un atlas de référence, l'automatisation complète de la segmentation paraît indispensable pour le traitement de grandes bases d'images (c'est le cas dans notre application).

**Types de structures segmentées** Les applications d'une méthode de segmentation sont limitées par le type de structures qu'elles sont capables de distinguer. Il est souhaitable qu'elle

<sup>4</sup>L'automatisation complète n'est cependant pas suffisante pour garantir la reproductibilité. Une méthode automatique peut être sensible à certaines perturbations de l'acquisition et donner des résultats non reproductibles, avec des biais systématiques, mais imprédictibles.

puisse segmenter le plus grand volume possible (idéalement, l'ensemble de la tête), tout en distinguant les différentes structures dans le détail. Les approches région (i.e. qui associent une structure anatomique à chaque voxel, voir § 3.2.2) fournissent en général des segmentations de l'ensemble du cerveau en trois classes (MB/MG/LCR). Les approches surfaciques (§ 3.2.3) sont souvent orientées vers l'extraction détaillée d'une structure particulière.

**Vitesse de calcul** L'importance de la vitesse de calcul dépend de l'application. Les applications interactives, telles que la chirurgie assistée (IRM intra-opérative, réalité augmentée [125], figure 3.2), nécessitent des approches très rapides, comme le recalage affine rapide d'un atlas présegmenté, la simulation simple de déformations bio-mécaniques [125], ou la segmentation à partir du seul histogramme. Cependant, beaucoup d'applications autorisent une attente plus longue (30 h dans MacDonald *et al.* [68]). Si l'analyse ou le diagnostic ne sont pas urgents, un délai plus long est acceptable, à condition que le traitement soit entièrement automatique et fiable<sup>5</sup>. Des temps de calcul longs peuvent cependant être gênants dans une phase d'expérimentation et de réglage.

### 3.1.3 Spécificités de la segmentation de l'IRM cérébrale

La segmentation de l'IRM cérébrale présente des particularités par rapport à d'autres domaines d'application de la segmentation, comme, par exemple, la segmentation de cartes routières, ou la segmentation des visages. Ces particularités sont liées aux spécificités de l'objet à segmenter (l'anatomie du cerveau a été présentée au paragraphe 2.2) et au processus d'acquisition IRM (présenté au paragraphe 2.1).

**Prédictibilité des structures anatomiques** La position et la disposition relative de certaines structures à segmenter est assez prédictible, particulièrement pour les structures internes (noyaux gris, corps calleux, ventricules, etc.), et les formes externes (crâne). Ceci permet d'utiliser des connaissances *a priori* fortes pour segmenter ces structures (ces méthodes sont décrites au paragraphe § 3.2). En revanche, des structures comme les plis corticaux présentent une très forte variabilité anatomique [111] ce qui complique l'apport de connaissances *a priori*.

**Information d'intensité insuffisante** L'intensité observée sur les images IRM n'apporte qu'une information incomplète sur les structures anatomiques sous-jacentes. Plusieurs structures voisines peuvent avoir la même intensité : en IRM  $T_2$  il n'y a pas de différence de contraste entre la MG et la MB. En IRM  $T_1$ , certaines structures, comme le corps calleux, n'ont pas de bords précisément délimités. Inversement, une structure unique, comme le crâne (partie osseuse) peut avoir différentes intensités (moelle / os). Des régions considérées comme homogènes, telles que la MB peuvent présenter des variations d'intensité [70] selon leur structure microscopique (concentration en fibres). Dans un autre exemple, l'intensité IRM lors de la transition entre la MG (corps cellulaire des neurones) et la MB (axones myélinisés) est complexe, et peut être variable [132]. Les informations anatomiques, qui ne sont pas présentes dans l'intensité de l'image, doivent donc être intégrées dans le système de segmentation : c'est un aspect important des connaissances *a priori*. Une autre approche, permettant de palier à l'in-

---

<sup>5</sup>En pratique, un calcul qui dure plus d'une nuit devient difficilement acceptable. Néanmoins, sur le long terme, ces considérations doivent être examinés en tenant compte de l'évolution exponentielle de la puissance de calcul (« loi de Moore » : doublement tous les 18 mois [76]).

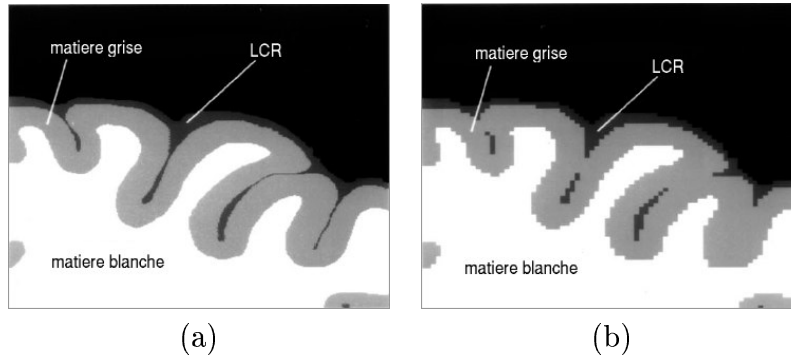


FIG. 3.3 – *En segmentation cérébrale, les problèmes de résolution peuvent être particulièrement importants pour de fines structures (LCR dans les sillons  $< 1$  mm, et même MG dans le cortex  $\simeq 2$  mm) (a) représentation simplifiée du cortex en haute résolution (b) cortex échantillonné : des portions de LCR au centre des sillons ont disparu (illustration d'après Xu et al. [127])*

suffisance des informations présentes dans une image, est l'utilisation simultanée de plusieurs modalités.

**Erreurs/Artefacts d'acquisition** Les principaux artefacts apparaissant en IRM ont été présentés au paragraphe 2.1.5, p. 20. Lors de la segmentation, celui l'artefact le plus souvent cité est l'artefact d'inhomogénéité de gain de l'antenne RF qui se manifeste par une variation spatialement lente de l'intensité des tissus sur l'image. La plupart des méthodes de segmentation tiennent compte de cet artefact, soit en l'éliminant par un pré-traitement [17], soit en le modélisant explicitement au cours de la segmentation. Il peut être modélisé comme un champ multiplicatif (voir 2.1.5) unique, ou bien, plus précisément avec des champs multiplicatifs distincts pour chaque classe de tissus [70]. L'autre problème d'acquisition couramment cité, est le bruit blanc additif (décrit au paragraphe 2.1.5, p. 21). Ce problème n'est pas spécifique à la segmentation IRM.

**Volumes partiels** Le terme « volume partiel » est utilisé dans le milieu de l'imagerie médicale pour désigner les problèmes dus à l'échantillonnage (ou résolution) de l'image. En effet, si on considère une portion d'image à la frontière de deux tissus différents, l'échantillonnage (filtrage anti-repliement + échantillonnage proprement dit) donnera des voxels qui portent une information provenant de plusieurs voxels voisins. Le terme « volume partiel » évoque une vision d'un voxel qui « contiendrait » un « mélange » de plusieurs tissus. Ceci correspond à une approximation dans laquelle le filtrage anti-repliement est un filtre porte dans le domaine spatial<sup>6</sup>. Cette approximation, aussi dénommée « mixel », est pratique, et est utilisée par plusieurs approches de la segmentation [95]. En segmentation cérébrale, les problèmes de résolution peuvent être particulièrement importants pour de fines structures (LCR dans les sillons  $< 1$  mm, et même MG dans le cortex  $\simeq 2$  mm, voir figure 3.3).

<sup>6</sup>La réalité de l'échantillonnage est beaucoup plus complexe et dépend du mode d'acquisition. Lorsque les images sont acquises en multi-coupes, le filtrage dans la direction perpendiculaire à la coupe est défini par un « profil de coupe », souvent gaussien. À l'intérieur d'une coupe (ou sur le volume entier pour les acquisitions 3D), l'acquisition se fait dans le domaine de Fourier et le filtre peut être proche d'un passe bas idéal (sinc dans le domaine spatial).

**Pratiques d'acquisition** L'automatisation exige que les méthodes de segmentation soient capables de gérer des images de sources différentes. Lors de la pratique clinique de l'IRM, les images sont souvent acquises en coupes épaisses ; la résolution à l'intérieur de chaque coupe étant élevée. La gestion de ces images anisotropes peut être délicate lors de la segmentation. Par ailleurs, les neurologues limitent aussi l'acquisition à une portion du cerveau, ce qui peut être gênant pour des méthodes utilisant des informations *a priori* sur l'ensemble du cerveau (recalage sur un atlas, histogramme, etc.).

**Perturbations pathologiques** Les modifications pathologiques, comme des déformations parfois très importantes (tumeurs, § 2.3.1, p. 28) peuvent perturber certaines approches de la segmentation. L'utilisation de connaissances *a priori* sur les formes pour guider la segmentation peut poser des problèmes si ces perturbations ne sont pas modélisées. Par exemple, la segmentation des ventricules, à l'aide de modèles déformables basés sur un apprentissage des formes, pourrait être perturbé si les ventricules sont fortement atrophiés à cause d'une pathologie (SEP).

## 3.2 État de l'art

Il existe une vaste littérature sur la segmentation d'images cérébrales. En 2002, Suri *et al.* [106] ont dénombré plus de 450 publications consacrées à la seule segmentation corticale. Nous ne proposons donc ici qu'une vue d'ensemble synthétique permettant de situer notre travail, et nous renvoyons le lecteur qui souhaiterait approfondir cette question vers des articles de synthèse [106, 90, 73].

Après une brève description des méthodes de segmentation des structures non corticales, nous nous focaliserons sur les méthodes de segmentation du cortex. Nous présenterons brièvement les outils qui se retrouvent sous diverses formes dans la littérature.

### 3.2.1 Méthodes de segmentation des structures non corticales

La moindre variabilité anatomique des structures non corticales du cerveau, et la subjectivité (information *a priori* ne se trouvant pas dans l'image) de leur définition, font de la segmentation de ces structures un domaine relativement distinct de la segmentation corticale. La connaissance *a priori* peut être apportée par un atlas [128, 8, 78] sous diverses formes (voir § 3.2.3) ou par une description statistique de la surface des objets construite à partir d'une base de structures présegmentées. La modélisation statistique peut reposer sur des propriétés géométriques locales et globales [98], sur l'analyse en composantes principales [117, 83], ou sur d'autres techniques de représentation des formes et distances entre structures [128].

### 3.2.2 Méthodes de segmentation des structures corticales

Dans leur article de synthèse Suri *et al.* [106] proposent une classification des méthodes de segmentation corticale (voir figure 3.4) qui divise les approches en trois grandes catégories : les approches région<sup>7</sup>, celles qui modélisent la surface de séparation entre les structures, et celles qui sont à cheval sur ces deux points de vue. Si cette classification permet une vision synthétique, elle est néanmoins, par nature, réductrice. Beaucoup de méthodes utilisent divers outils

---

<sup>7</sup>Les approches région sont celles qui associent une structure anatomique à chaque voxel, par opposition aux approches contour (ici le contour est une surface) qui délimitent la séparation entre structures anatomiques.

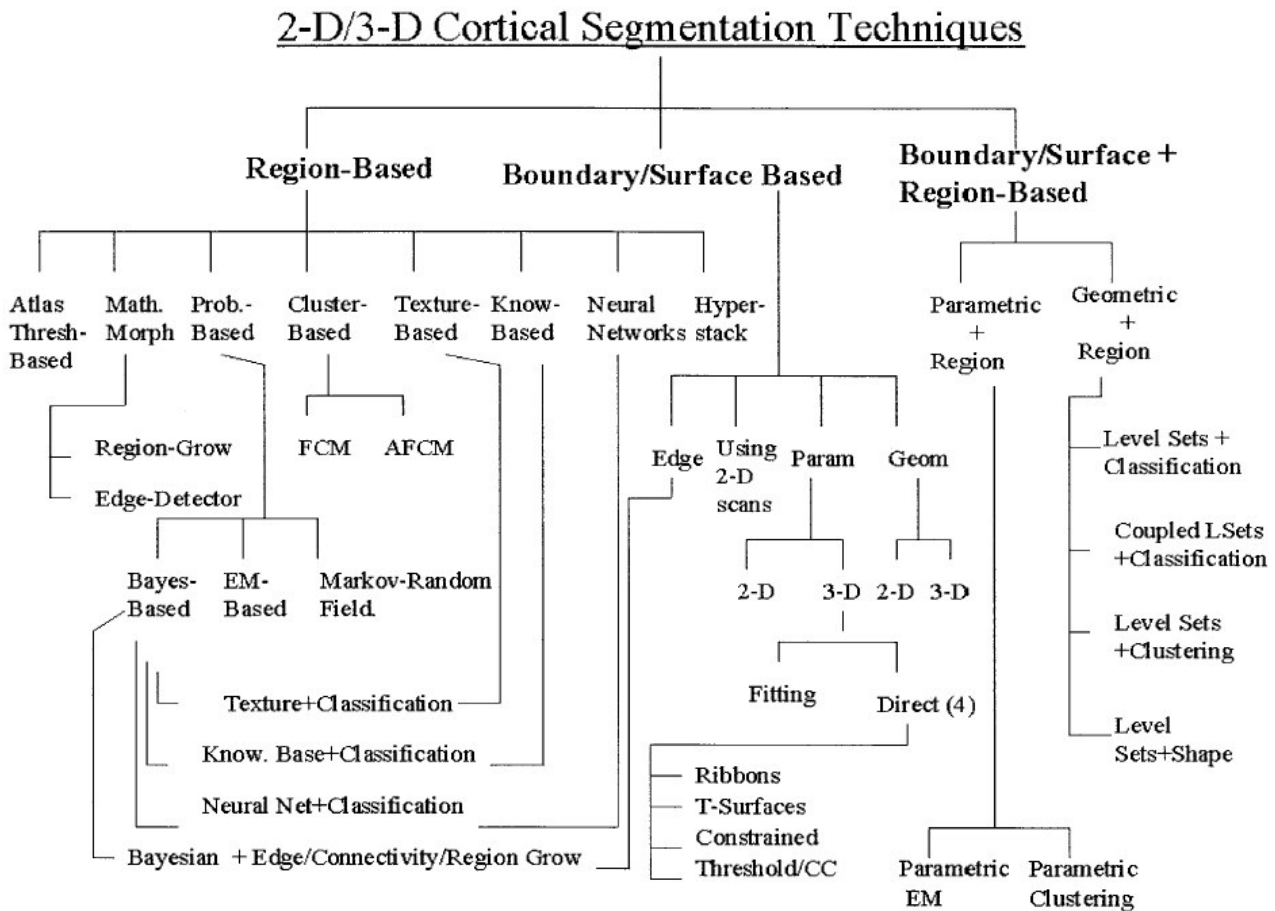


FIG. 3.4 – Classification des méthodes de segmentation corticales proposée par Suri et al. [106]. La principale distinction entre approches est faite par rapport au support de la segmentation : les approches région, celles qui modélisent la surface de séparation entre les structures et celles effectuant une synthèse entre ces deux points de vue (illustration tirée de Suri et al. [106])

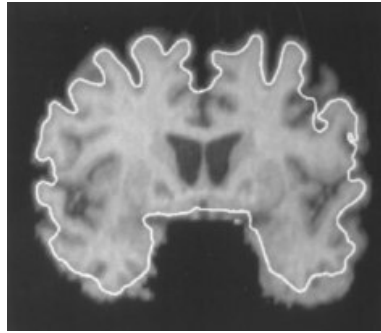


FIG. 3.5 – *Les méthodes d'évolution de surfaces peuvent se bloquer dans de minima locaux, et n'atteignent pas toujours le fond des plis corticaux (tiré de Xu et al. [127])*

provenant de catégories différentes. Les méthodes de segmentation corticale doivent effectuer des tâches qui peuvent, ou non, être menées par étapes (ou modules) séparés (enchaînement de traitements). Parmi ces tâches, distinguons les suivantes : extraction de l'enveloppe externe du cerveau, correction des artefacts RF, débruitage, estimation de paramètres d'intensité, correction de topologie de la surface corticale [59]. Une approche modulaire [68, 17] d'enchaînement de traitements a pour avantage de créer des outils réutilisables, aux fonctionnalités bien délimitées et évaluables. Cependant, enchaîner des traitements implique aussi de ne plus pouvoir remettre en cause leurs erreurs à des étapes ultérieures.

### 3.2.3 Segmentation corticale : outils

Nous avons choisi de présenter brièvement ici un ensemble d'outils qui apparaissent fréquemment dans différentes approches de segmentation corticale.

**Atlas** En utilisant différentes formes de recalage [112, 110], des informations associées à une image de référence (atlas) peuvent être projetées sur l'image à segmenter (état de l'art dans [111]). Ce type d'outils peut être utilisé pour projeter directement une carte de segmentation pour des structures à faible variabilité anatomique [78, 125, 14, 108], ou bien pour projeter d'autres types d'informations (probabilités de présence de types de tissus [70, 2, 121], modèle statistique [124]).

**Évolution de surface et modèles déformables** Au lieu d'associer un label à un point de l'image, de nombreuses méthodes de segmentation [68, 17, 27, 127, 20, 119] (état de l'art [73]) cherchent à reconstruire la surface séparant les structures anatomiques (paradigme des snakes [52]). La surface évolue alors sous l'effet de contraintes internes (régularité de la surface) ou externes (contraintes image ou autres), se subdivisant (raffinement) éventuellement selon les besoins. Ces contraintes peuvent être exprimées comme des forces [17, 27, 127, 20, 119] ou dans une formulation énergétique [68] (voir formulations énergétique, ci-dessous). Le paradigme de l'évolution de surface, permet, par construction, d'imposer une segmentation résultante composée de zones homogènes, régulières et bien délimitées. Des problèmes d'auto-intersection peuvent apparaître et ne sont solubles qu'au prix d'un effort calculatoire important [68]. L'évolution de ces surfaces peut se bloquer (minimum local d'énergie, voir figure 3.5) dans certains cas, empêchant la surface d'atteindre les plis corticaux profonds [127]. Lorsque l'objet à segmenter est bruité (occlusions, trous) ou lorsque l'initialisation est très éloignée du résultat souhaité, les surfaces déformables ont des difficultés à adapter leur structure locale



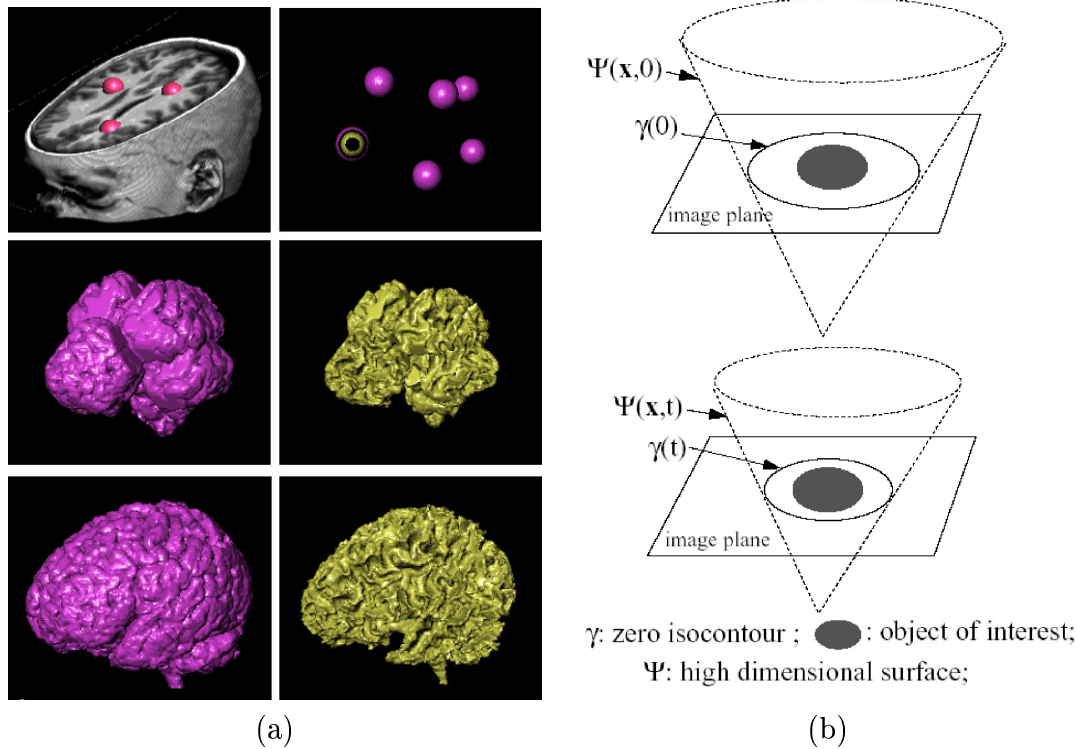


FIG. 3.6 – Méthode des surfaces couplées présentée par Zeng et al. [130]. Les surfaces  $\gamma$  ont une représentation implicite  $\Psi(x, t) = 0$  évoluant à partir d'une initialisation avec des sphères concentriques. (a)-gauche : évolution de l'interface MB/MG, (a)-droite : évolution (couplée) de l'interface MG/LCR. (b) illustration de la représentation implicite d'une surface convergeant vers un objet (illustrations d'après Zeng et al. [129]).

(changements de topologie), conduisant à des blocages. Pour estimer la complexité calculatoire et comparer la résolution des résultats avec ceux obtenus par les approches région, on peut considérer le nombre de triangles nécessaires. MacDonald *et al.* [68] obtiennent des surfaces séparant la MB de la MG dans le cortex se composant d'environ 82 000 triangles dont les sommets sont espacés en moyenne de 1 mm. En termes de précision ceci est comparable à des voxels sub-millimétriques (environ 0,5 mm) dans une approche région.

**Ensembles de niveau** (*level-set*) Pour résoudre les difficultés évoquées précédemment concernant les représentations explicites (polygonales) des surfaces, d'autres représentations, plus souples peuvent être souhaitables. Une solution théorique élégante [96] est d'utiliser une représentation implicite des surfaces. La surface est alors décrite par le niveau 0 de l'hypersurface  $\Psi(x, t)$  (où  $\Psi$  est une fonction de  $\mathbb{R}^3 \times \mathbb{R}$  dans  $\mathbb{R}$ ), et l'évolution de la surface est déduite de l'évolution de  $\Psi(t)$  qui est elle-même gouvernée par une équation différentielle. Ce type de représentation permet d'obtenir des surfaces dont la topologie s'adapte naturellement au cours de son évolution. Zeng *et al.* [130] ont utilisé une paire de surfaces couplées pour représenter les interfaces entre MG/MB et MG/LCR (voir figure 3.6). En pratique, la méthode alterne, à chaque itération, entre une représentation polygonale (déduite de  $\Psi$  par une extraction d'iso-surface : *marching cubes*), et une représentation « image » où  $\Psi$  est échantillonnée sur une grille discrète de points. D'autres travaux de segmentation corticale employant ce type d'approche sont plus orientés sur les développements théoriques [36, 37, 38], permettant d'éviter

des problèmes calculatoires délicats.

**Formulation énergétique** Le problème de segmentation est fréquemment [68, 52, 73] posé comme un problème de minimisation d'une fonction de coût (énergie). En pondérant les différents termes additifs qui composent l'énergie, cette formulation permet de faire un compromis entre divers facteurs souhaitables de la segmentation (souvent contraintes image / connaissances *a priori*). Ce type de formulation porte de fortes analogies avec les formulations probabilistes (ou bayésiennes, ci-dessous) intervenant dans la résolution des problèmes mal posés. Dans notre approche, nous avons adopté cette formulation (§ 4.2, p. 58).

**Formulations probabilistes et bayésiennes** Un paradigme très utilisé dans la segmentation en trois classes (MB/MG/LCR) sur des approches région, est de traduire le problème de segmentation dans un cadre probabiliste. On estime alors à la fois les paramètres décrivant le modèle probabiliste de l'intensité (le plus souvent gaussien) de chaque classe (ou mélange de classes [95]), et la classe de chaque voxel de l'image (ou la probabilité d'être dans cette classe). Les approches les plus simples estiment les paramètres d'intensité à partir du seul histogramme de l'image, en tenant compte éventuellement de l'artefact RF, puis calculent la segmentation en seillant l'image. Elles n'utilisent donc pas les informations spatiales de l'image, sauf pour la correction RF, ce qui est une limitation sévère. De nombreuses approches, plus sophistiquées, incluent des contraintes de cohérence spatiale sur la segmentation obtenue (par exemple par champs de Markov, voir ci-dessous). Les formulations bayésiennes permettent de spécifier des connaissances *a priori*, comme celle de la cohérence spatiale, en modélisant la probabilité d'observer une segmentation donnée *a priori*, indépendamment de l'image à segmenter. Le problème de segmentation ne s'exprime alors plus de manière analytique et doit utiliser une procédure itérative pour converger vers une solution optimale.

**Champs de Markov** Les champs de Markov (Geman *et al.* [31]) permettent de contraindre la cohérence d'une segmentation en modélisant la probabilité d'observer les différentes configurations de labels possibles dans le voisinage immédiat de chaque voxel. Ils sont fréquemment [131, 95, 70, 121, 97] utilisés comme modèles *a priori* dans les formulations bayésiennes. Les approches se distinguent par la manière dont les paramètres d'intensité sont estimés (une fois pour toutes, ou ajustés itérativement), par les classes modélisées (trois classes, mixels) et par le type de modélisation markovienne (champs de Markov caché, ou non). Remarquons que la formulation probabiliste des champs de Markov peut se traduire en une formulation énergétique [31].

**Mélanges de gaussiennes** [39] La structure particulière du cerveau (division MB/MG/LCR) permet de modéliser l'histogramme de l'image observée comme une somme de contributions gaussiennes. Certaines méthodes se limitent à trois classes, d'autres tiennent compte des problèmes de volumes partiels en ajoutant des classes intermédiaires [95, 39]. Ce type de modèle des intensités est souvent utilisé conjointement à une formulation probabiliste ou bayésienne.

**Logique floue** Certaines approches [89, 127] déterminent une segmentation où la structure anatomique associée à chaque point de l'image n'est définie que par un degré d'appartenance. Ce type de représentation permet de différer des décisions (i.e. ne pas avoir à trancher sur les classes d'appartenance) jusqu'à ce que plus d'informations soient disponibles (convergence).

En pratique, une décision doit souvent être prise à la fin de l'algorithme, et il n'est pas certain que la segmentation non floue finale ait les propriétés souhaitées.

**Filtrage non-linéaire** Des méthodes de filtrage non-linéaire, souvent formulées comme un processus de diffusion [32], permettent de lisser (débruiter) des images en préservant (ou renforçant) les contours[82, 103]. Ces techniques peuvent être utilisées dans une phase de pré-traitement [97, 128].

### 3.3 Conclusion

Dans ce chapitre nous avons décrit les principaux enjeux, et les grandes lignes des méthodes existantes de segmentation cérébrale. Dans le chapitre suivant, nous proposons une nouvelle approche à la segmentation. Nous y décrirons les avantages de notre approche, et nous la situerons par rapport aux approches décrites ici.

# Chapitre 4

## Description de notre approche

Dans ce chapitre nous allons proposer une nouvelle approche de segmentation. Après l'avoir présentée dans ses grandes lignes (§ 4.1), nous expliquerons comment la segmentation évolue itérativement (§ 4.2, p. 58), avant de décrire les différentes contraintes qui influencent son évolution (§ 4.4, p. 62 et § 4.5, p. 69). Au chapitre suivant (chapitre 5, p. 89), notre approche sera appliquée à la segmentation des sillons corticaux, et évaluée expérimentalement dans ce cadre.

### 4.1 Présentation

**Motivations** Segmenter automatiquement le cerveau de manière fiable et robuste est un problème délicat. Pourtant, c'est une tâche que l'expert pourrait effectuer avec une certaine précision, s'il disposait d'un temps illimité. L'expert a une connaissance *a priori* très précise de ce qu'il souhaite segmenter. Il connaît la forme générale et la disposition des objets qu'il doit segmenter. Il peut faire appel à des principes logiques de « bon sens ». Ce type de connaissances est intégré, à des degrés différents, et souvent de manière très implicite, dans toutes les méthodes évoquées au chapitre précédent. Nous avons cherché ici, à construire un système de segmentation qui soit capable d'intégrer de manière explicite des connaissances *a priori* complexes. Les techniques d'acquisition 3D, et leurs artefacts associés, évoluent rapidement. Nous pensons donc qu'il est important de bien séparer la modélisation des connaissances *a priori* et la compensation des erreurs d'acquisition. Nous avons cherché à construire un modèle du processus d'acquisition qui soit adaptable et qui permette une segmentation sous-voxel. Un autre objectif majeur de ce système de segmentation est de pouvoir représenter les détails fins (sous-voxel) de l'objet qui paraissent indispensables pour construire une segmentation anatomiquement cohérente.

La méthode que nous présentons possède des analogies avec certains aspects des approches utilisant des surfaces déformables. Cependant, notre approche discrète permet de s'affranchir des problèmes liés aux représentations paramétriques ou implicites [130] des surfaces (voir § 3.2.3, p. 51).

**Représentation de la segmentation** Dans notre approche, la segmentation courante est représentée par une carte de labels  $L : \Omega \rightarrow \{1 \dots n\}$  associant à chaque point de l'espace un label parmi les  $n$  classes possibles (voir figure 4.1). Le support  $\Omega \in \mathbb{N}^3$  de la carte de labels  $L$  n'est pas celui de l'image à segmenter  $I_0$ . En effet, la carte de labels  $L$  peut être choisie d'une résolution supérieure à celle de l'image à segmenter. L'ajout de connaissances *a priori*

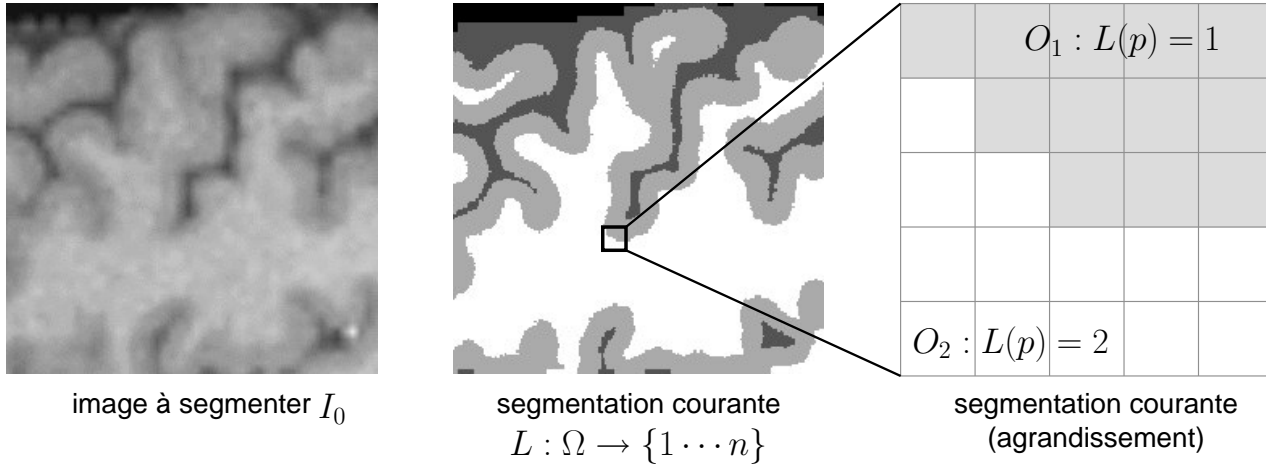


FIG. 4.1 – La segmentation courante est représentée par une image de labels  $L$  pouvant prendre des valeurs entre 1 et  $n$ . À droite est montré un agrandissement d'une portion de  $L$ . L'ensemble des points de label  $k$  constitue un objet appelé  $O_k$ . La résolution de  $L$  peut être supérieure à celle de l'image à segmenter (à gauche).

fortes permet de segmenter des détails plus fins que ceux présents dans l'image. Ainsi, dans des images IRM basse résolution ( $2 \text{ mm}^3$ ), le LCR au centre des plis corticaux n'est presque pas visible. Mais la connaissance de la distribution particulière des labels <sup>1</sup> permet de contraindre la segmentation et de retrouver les portions (sous-voxel) de LCR. L'image de labels  $L$  cherche à représenter la réalité physique. La résolution de  $L$  ne dépend donc pas de la résolution de l'image qu'on segmente, mais de la taille des structures que l'on souhaite pouvoir représenter<sup>2</sup>.

**Évolution de la segmentation** La carte de segmentation  $L$  évolue de manière à minimiser une énergie qui est constituée d'une somme de termes appelés contraintes (voir figure 4.2). On impose à la segmentation  $L$  de n'évoluer qu'aux frontières entre labels différents, ce qui rend notre approche comparable à une approche par évolution de surfaces. Ce cadre d'évolution de la segmentation sera décrit au paragraphe 4.2.

**Contraintes** Les contraintes, évoquées précédemment peuvent être conceptuellement classées en deux catégories<sup>3</sup> : les contraintes image, et les contraintes *a priori*. Plusieurs contraintes image opérant simultanément à des échelles différentes sont employées, permettant d'atteindre à la fois une précision sous-voxel et une convergence globale (à grande échelle). Elles utilisent un modèle réaliste du processus d'acquisition, permettant de comparer la segmentation courante  $L$  avec l'image à segmenter  $I_0$ , et d'en déduire les variations d'énergie associées à une évolution de la segmentation  $L$ . Les contraintes image sont décrites au paragraphe 4.4. Les contraintes *a priori* représentent les connaissances dont on dispose sur les propriétés des objets à segmenter. Elles comprennent une contrainte de distribution relative, qui donne la probabilité d'observer un label à une distance donnée d'un autre label et une contrainte d'épaisseur. Ces contraintes ne dépendent pas de l'image à segmenter. Elles sont décrites au paragraphe

<sup>1</sup>Le LCR est entouré d'une épaisseur relativement constante de MG.

<sup>2</sup>Par exemple, si l'image acquise est anisotrope, la segmentation ne le sera pas : la *résolution* de la segmentation sera la même dans toutes les directions. La *précision* de la segmentation obtenue pourra être inférieure dans la direction où la résolution de l'image acquise est inférieure.

<sup>3</sup>Ces catégories sont parfois appelées « attache aux données » et « terme de régularisation ».

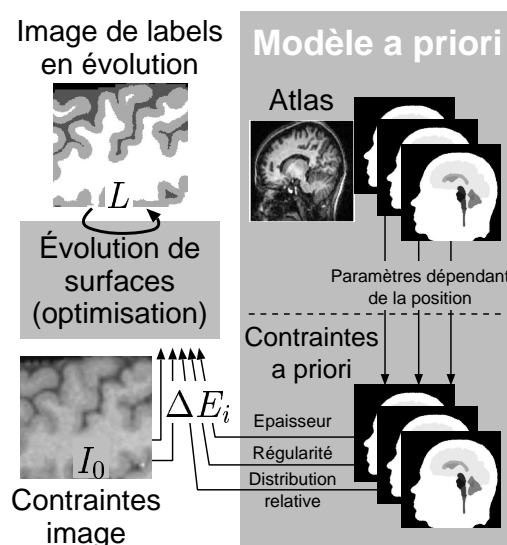


FIG. 4.2 – Vue d'ensemble du processus de segmentation proposé. Une image de labels  $L$  évolue sous l'effet de contraintes image et de contraintes a priori. Les contraintes peuvent être paramétrées à l'aide d'un atlas.

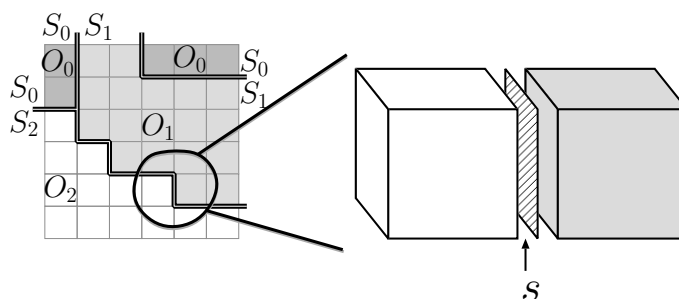


FIG. 4.3 – L'élément de surface  $s$  séparant deux voxels voisins de label différents est un carré. L'ensemble des carrés dont une des faces donne sur un voxel de label  $i$ , forment la surface frontière  $S_i$  de ce label  $i$ .

4.5, p. 69.

**Pré-traitement et initialisation** Pour estimer les paramètres du modèle d'acquisition nécessaires aux contraintes image, nous utilisons une fonction de transfert d'intensité calculée à partir de l'image à segmenter et d'une image de référence. De plus, notre approche de segmentation étant itérative, elle nécessite une initialisation  $L_0$ . Nous montrons (§ 4.6.2, p. 85) que notre approche n'est pas sensible à cette initialisation.

**Atlas statistique** Afin de s'adapter aux spécificités de chaque région cérébrale, les contraintes sont paramétrées à l'aide d'un atlas. Cet atlas, composé d'une image de référence et d'images de paramètres, est plaqué sur l'image à segmenter à l'aide d'un recalage déformable multi-échelles. Cet atlas statistique est décrit au paragraphe 5.2.1, p. 100 et les différents paramètres susceptibles d'y être représentés seront explicitement précisés lors des descriptions de chaque contrainte.

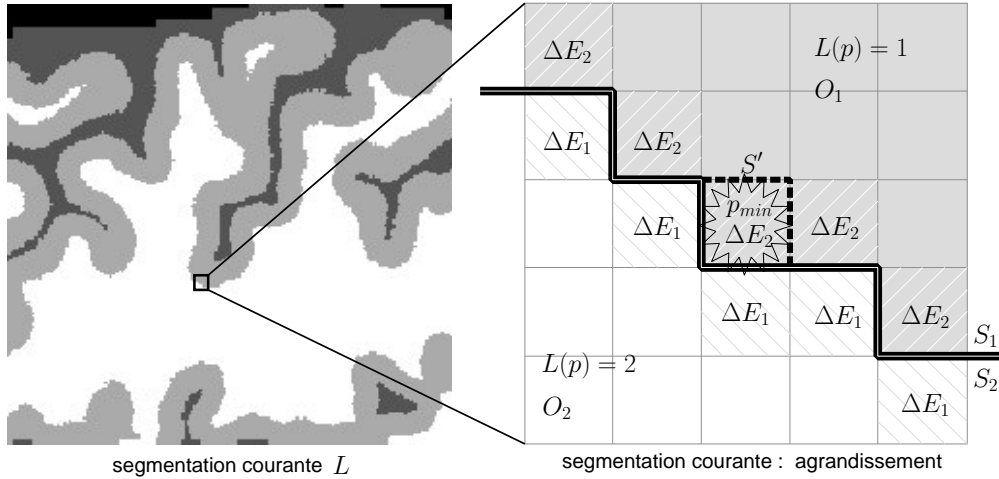


FIG. 4.4 – En chaque point de la frontière entre labels, la variation d'énergie qui aurait lieu si ce point changeait de label, est calculée. Le point  $p_{min}$  qui provoque la plus grande diminution d'énergie est choisi. Le label en ce point est alors changé. Ceci modifie donc la surface frontière entre les labels  $S'$ .

**Notations** Nous fournissons dans le tableau 4.1 un résumé de certaines notations qui seront employées dans ce chapitre, avec une brève explication, un renvoi vers le paragraphe du texte où chaque notation est introduite, et éventuellement vers une figure illustrant cette notation.

## 4.2 Évolution de la segmentation

Comme cela a été esquissé précédemment, la segmentation  $L$  évolue itérativement de manière à diminuer une énergie. Cette énergie  $E(L)$  associée à l'image de labels  $L$  est composée d'une somme de termes :

$$E(L) \triangleq \sum \alpha_i E_i(L) \quad (4.1)$$

Chaque terme  $\alpha_i E_i(L)$  est appelé contrainte. Une description détaillée de chaque type de contrainte sera fournie aux paragraphes 4.4 et 4.5, p. 69.

À chaque itération un label est changé en un point de la segmentation  $L$  (voir § 4.2.2). Pour basculer un label dans  $L$ , nous allons, en chaque point  $p \in \Omega$ , calculer la variation d'énergie qui aurait lieu si on changeait de label en ce point. Nous choisirons ensuite le point  $p_{min}$  pour lequel l'énergie diminue le plus (voir figure 4.4).

Définissons plus précisément la variation d'énergie : si l'image de labels  $L$  change au point  $p$  en passant du label  $l_0$  au label  $l$ , la variation d'énergie  $\Delta E_l(p)$  est alors

$$\Delta E_l(p) = E(L + (l - l_0)\delta_{[p]}) - E(L) \quad (4.2)$$

On définit de la même manière le changement d'énergie  $\Delta E_{i,l}(p)$  associé à chaque contrainte  $i$ . Notons que pour un label  $l$  donné,  $p \rightarrow \Delta E_l(p)$  définit une image  $\Delta E_l$  de support  $\Omega$ . D'après les considérations qui suivent (§ 4.2.1, p. 60),  $\Delta E_l$  est une image creuse, c'est-à-dire, n'ayant des valeurs non nulles qu'en un nombre limité de points. Cette propriété sera exploitée pour réduire l'encombrement mémoire et la charge calculatoire.

L'énergie ainsi définie, on est assuré qu'à chaque itération, l'énergie globale  $E(L)$  diminue. Lorsqu'il n'existe plus de points dont la modification diminuerait l'énergie ( $\forall p \in \Omega \Delta E(p) \geq 0$ ), la segmentation s'arrête.

Sym.	Signification	Renvoi	Fig.
$I_0$	$I_0 : \mathbb{R}^3 \rightarrow \mathbb{R}$ . L'image que l'on souhaite segmenter. Notez que la résolution d'acquisition est en général inférieure à celle de la segmentation courante $L$ .	4.2, p. 58, 4.4.2, p. 63	4.7
$I$	$I : \mathbb{R}^3 \rightarrow \mathbb{R}$ . Image reconstruite (prédite) à partir de la segmentation courante $L$ . Cette image peut être comparée en toute rigueur à l'image acquise $I_0$ .	4.4.2, p. 63	4.7
$\Omega$	$\Omega \subset \mathbb{N}^3$ . Support, en haute résolution de l'image de labels $L$ .	4.2, p. 58	4.1
$L$	$L : \Omega \rightarrow \{1 \dots n\}$ . Segmentation courante en haute résolution. C'est cette image qu'on fait évoluer à chaque itération. Cette image associé un label à chaque point du support $\Omega$ .	4.2, p. 58	4.1
$O_k$	$O_k = \{p \in \Omega   L(p) = i\}$ . « Objet » (pouvant être non connexe) correspondant au label $i$ . C'est-à-dire, l'ensemble des points où $L(p)$ vaut $i$ .	4.2, p. 58	4.1
$s$	$s \subset \mathbb{R}^3$ . Élément de surface : un carré qui sépare deux voxels voisins ayant des valeurs des labels différents. Les deux faces de $s$ sont les voxels $s_{p0}$ et $s_{p1}$	4.2, p. 58	4.3
$S_i$	$S_i = \{s   L(s_{p0}) = i \text{ ou } L(s_{p1}) = i\}$ . La surface formée par l'ensemble des éléments de surfaces voisinant le label $i$ .	4.5, p. 70, 4.2, p. 58	4.3, 4.4
$R$	$R : \mathbb{R}^3 \rightarrow \mathbb{N}$ . Fonction tridimensionnelle modélisant le cerveau réel. À chaque point est associé un tissu dont l'intensité sur une image IRM est déterminée à partir de la fonction $V_0$ . La fonction $R$ ne peut pas être déterminée, il s'agit d'une abstraction servant à modéliser le processus d'acquisition.	4.4.2, p. 63	4.7
$V_0$	$V_0 : \mathbb{N} \rightarrow \mathbb{R}$ . Correspondance entre une structure anatomique et une valeur d'intensité sur une image IRM. La fonction $V_0$ ne peut pas être déterminée, il s'agit d'une abstraction servant à modéliser le processus d'acquisition.	4.4.2, p. 63, 4.6.1, p. 85	4.7, 4.28
$V$	$V : \mathbb{N} \rightarrow \mathbb{R}$ correspondance entre un label et une intensité. La fonction $V$ est estimée.	4.4.2, p. 63, 4.6.1, p. 85	4.7, 4.28
$N^i(p)$	$N^i(p) \subset \mathbb{N}^3$ . Ensemble des points voisins d'un point $p$ . L'indice $i$ désigne le type de voisinage utilisé. Dans le cadre de la segmentation, on n'utilisera que les voisinages $N^6$ et $N^{26}$ .		
$M^i(p)$	$M^i(p) : \mathbb{N}^3 \rightarrow \mathbb{R}$ . Coefficients définissant la distance de chanfrein associée à un voisinage $N^i(p)$ .		
$\triangleq$	On utilisera le symbole $\triangleq$ au lieu de $=$ pour désigner une définition. Ceci permet de distinguer plus aisément les définitions de résultats d'un raisonnement ou d'un calcul.		
$\delta_p$	Distribution de Dirac tridimensionnelle centrée en $p \in \mathbb{R}^3$		
$\delta_{[p]}$	$\delta_{[p]} : \mathbb{N}^3 \rightarrow \{0, 1\}$ Impulsion unité en $p : \delta_{[p]}(p) = 1$ et $\delta_{[p]}(x) = 0$ pour $x \neq p$ .		

TAB. 4.1 – Récapitulatif des notations employées. Les deux dernières colonnes fournissent des renvois vers les paragraphes du texte et les figures où la notion est définie et illustrée.



### 4.2.1 Restriction de l'évolution aux frontières : contrainte $E_0$

Dans le cadre exposé précédemment, le changement peut avoir lieu en tout point  $p \in \Omega$ . Nous allons introduire une contrainte qui limitera les changements aux frontières<sup>4</sup> entre deux labels. Soit  $E_0$  définie par :

$$E_0(L) \triangleq \begin{cases} \infty & \text{si } \exists p \in \Omega \mid \forall p' \in N^6(p) : L(p') \neq L(p) \\ 0 & \text{sinon} \end{cases} \quad (4.3)$$

$E_0$  interdit toute configuration de  $L$  où on trouverait un voxel isolé ayant un label différent de tous ses voisins. En particulier, si  $L$  venait à changer en un point ne se trouvant pas à une frontière, alors on aurait un voxel isolé et l'énergie  $E_0$  serait infinie  $E_0(L) = \infty$ . On considérera désormais que les évolutions ne s'effectuent qu'aux frontières et, pour alléger les notations, on omettra d'écrire le terme  $E_0$  dans la définition de l'énergie globale (équation 4.1).

**Intérêt de la restriction aux frontières** Si les labels pouvaient basculer en n'importe quelle position, on perdrait rapidement la cohérence des surfaces séparant les différentes régions. Un des intérêts majeurs des méthodes basées sur l'évolution de surfaces (voir § 3.2.3, p. 51) est de fortement contraindre l'espace des solutions possibles en imposant des régions aux frontières/interfaces bien définies. En imposant cette contrainte  $E_0$  dans notre modèle, nous nous plaçons partiellement dans le paradigme de l'évolution de surfaces, sans certains de ses désavantages (auto-intersection, problèmes de changement de topologie, problèmes calculatoires etc., voir l'introduction § 4.1). Un deuxième intérêt, très important en pratique, est que dans ce cadre nous n'avons à calculer la variation d'énergie qu'aux frontières, et non sur tout  $\Omega$ . Ceci permet de réduire considérablement la charge calculatoire et l'encombrement mémoire.

### 4.2.2 Approximation de localité

Dans le cadre exposé précédemment, on ne bascule de label qu'en un seul point à chaque itération. Nous allons assouplir cette règle pour permettre d'en basculer en plusieurs points à chaque itération.

Le basculement d'un voxel peut changer les propriétés (épaisseur, forme) d'un objet sur des points distants. Par conséquent, ce basculement peut modifier la contribution à l'énergie d'un point distant. On fait alors l'hypothèse que les contributions à la variation d'énergie en un point  $p_{min}$  ne sont dues qu'à des labels se trouvant à moins d'une distance  $r$ , appelée rayon d'interaction. Après le basculement de  $p_{min}$ , les variations d'énergie calculées dans la sphère de rayon  $r$  ne sont donc plus valables. Basculer un label en un point  $p_1$  à l'intérieur de cette sphère pourrait faire augmenter l'énergie. À l'inverse, la variation d'énergie qui avait été calculée en un point  $p_2$  à l'extérieur de cette sphère est toujours valable (voir figure 4.5). Le basculement de labels en des points à l'extérieur de cette sphère ne fera donc pas augmenter l'énergie. On peut ainsi basculer plusieurs labels en une seule itération.

L'approche pour la minimisation suivante en découle : l'ensemble des positions où une diminution d'énergie est possible, est trié en ordre de  $\Delta E$  croissant, c'est-à-dire en mettant en premier, les points où ont lieu les plus grandes diminutions d'énergie. Ces candidats sont ensuite parcourus, et ne sont pris en compte que s'ils ne se trouvent pas à l'intérieur d'une

<sup>4</sup>La frontière de  $L$  est l'ensemble de points  $p \in \Omega$  ayant un label différent qu'un de leurs voisins (6-voisinage) :  $\{p \in \Omega : \exists p' \in N^6(p) \mid L(p') \neq L(p)\}$ .

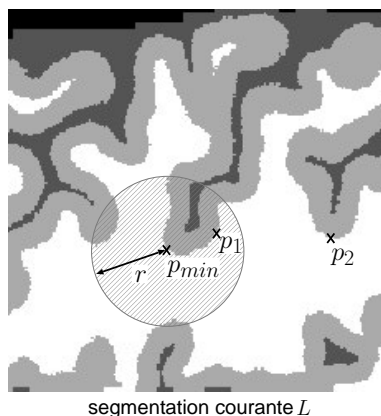


FIG. 4.5 – *Approximation de localité* : On suppose que les contributions à la variation d'énergie en un point  $p_{min}$  sont dues à des labels se trouvant à moins d'une distance  $r$ . Après le basculement de  $p_{min}$  les variations d'énergies calculées dans la sphère de rayon  $r$  ne sont donc plus valables. Basculer un label en un point  $p_1$  à l'intérieur de cette sphère pourrait faire augmenter l'énergie. À l'inverse, la variation d'énergie qui avait été calculée en un point  $p_2$  à l'extérieur de cette sphère est toujours valable. On peut ainsi basculer plusieurs labels en une seule itération.

sphère associée à un des candidats précédemment pris en compte. Cette approche garantit que l'énergie diminue effectivement à chaque itération. Notons que cette approche n'est pas strictement équivalente à l'approche « naïve » qui consisterait à ne changer qu'un seul label à chaque itération. La nouvelle approche ne garantit pas que la diminution soit maximale à chaque basculement. Cependant, l'approximation est correcte, et l'approche « naïve » n'est pas envisageable en pratique.

Le rayon d'interaction  $r$  est déterminé en considérant la définition des différentes contraintes et en envisageant les configurations plausibles. Le choix de  $r$  est important en termes de performances. Un rayon d'interaction trop grand diminuera le nombre de basculements par itération et donc augmentera le nombre d'itérations nécessaires à la convergence. Un rayon de convergence  $r$  trop faible se manifeste par différents phénomènes et particulièrement par une oscillation des surfaces autour de leur position d'équilibre.

En pratique, le choix du rayon d'interaction permettra de réguler le nombre de voxels qui pourront basculer à chaque itération et donc de réguler la vitesse d'évolution de la segmentation (voir aussi § 5.1.4, p. 94). Pour accélérer la convergence, on pourra initialement utiliser un faible rayon d'interaction. On acceptera alors qu'en certains points, des changements puissent provoquer une augmentation de l'énergie. Ensuite, pour affiner les résultats, on augmentera progressivement le rayon d'interaction  $r$  au cours de la segmentation. On obtiendra ainsi une convergence rapide dans les premières itérations, puis un résultat progressivement plus précis.

## 4.3 Contraintes : introduction

Dans les paragraphes qui suivent, nous allons définir les différents types de contraintes. Chaque type de contrainte est défini par l'expression<sup>5</sup> de son énergie  $E$ . Plusieurs termes d'un même type de contrainte peuvent figurer simultanément dans l'énergie globale (figure 4.6).

<sup>5</sup>Pour simplifier les notations, on omet désormais l'indice  $i$  de chaque contrainte  $E_i$  donné à l'équation 4.1.

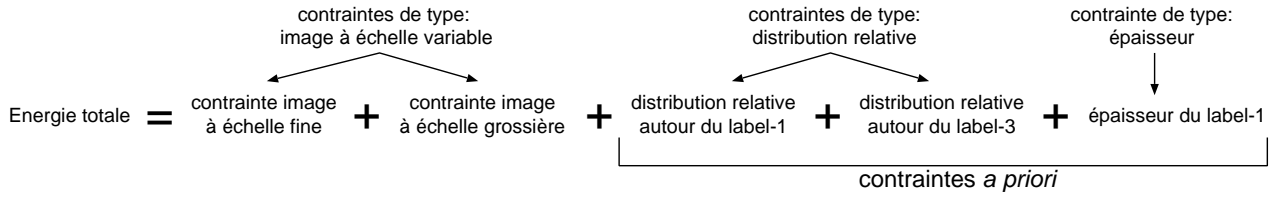


FIG. 4.6 – Plusieurs termes d’un même type de contrainte peuvent figurer simultanément dans l’énergie globale. Dans cet exemple, deux contraintes image sont utilisées à des échelles différentes. Deux contraintes de distribution relative, autour de labels différents, sont employées.

Une fois l’énergie d’une contrainte définie, il est nécessaire, pour déterminer son influence, de calculer la variation d’énergie associée. Cette variation est celle qui aurait lieu si un label de la carte de segmentation  $L$  changeait d’une valeur  $l_0$  à une valeur  $l$  en un point. L’expression de la variation  $\Delta E_l(p)$  est donc déterminée par une simple soustraction ( $E$  après changement moins  $E$  avant changement) à partir de la définition de l’énergie (équation 4.1). Pour toutes les contraintes envisagées, l’énergie est définie comme une somme d’un ensemble de contributions. Pour la contrainte image et la contrainte de distribution relative, il s’agit de sommes sur l’ensemble de l’image (où carte de label). Pour la contrainte d’épaisseur, il s’agit d’une somme sur l’ensemble de la surface frontière. Calculer directement la variation d’énergie par soustraction, implique alors de recalculer ces sommes<sup>6</sup>. Or, ce calcul de la variation d’énergie doit être fait pour chaque point  $p$  de la frontière, avant de pouvoir décider en quels points il est possible de basculer des labels. Des approximations, ou des développements analytiques, sont donc nécessaires pour permettre le calcul de  $\Delta E_l(p)$  sur l’ensemble des points de la frontière, en un temps raisonnable.

Pour chaque contrainte, nous donnerons des exemples d’évolution sur des objets simples bidimensionnels. Ces évolutions permettent à la fois d’illustrer le fonctionnement de ces contraintes et de valider leur fonctionnement, séparément d’autres contraintes.

Pour chacune de ces contraintes, nous indiquerons également les paramètres associés qui sont susceptibles d’être introduits dans l’atlas.

## 4.4 Contraintes image : l’attache aux données

Le premier type de contraintes que nous allons décrire, représente l’attache aux données. Sans ces contraintes, la segmentation obtenue ne dépendrait pas de l’image source. Pour parvenir à une précision sous-voxel, il est indispensable de modéliser de manière rigoureuse le processus d’acquisition.

### 4.4.1 Notations

Les images évoquées ici sont des images continues de  $\mathbb{R}^3 \rightarrow \mathbb{R}$ . Les images discrètes associées peuvent être déduites par échantillonnage à une résolution  $r$  donnée :  $I_{i,j,k}^r = I\left(\frac{i}{r}, \frac{j}{r}, \frac{k}{r}\right)$ . Il y a équivalence entre la représentation  $I^r$  discrète et la représentation  $I$  continue si  $I$  respecte les conditions de Shannon pour la résolution  $r$ . Les raisonnements suivants auront lieu dans le domaine continu, ce qui évitera d’évoquer artificiellement la résolution des images.

<sup>6</sup>D’autres quantités, comme les cartes de distance, doivent aussi être recalculées.

Les traitements informatiques, qui se déroulent dans le domaine discret, seront explicitement indiqués.

Rappelons que la carte de labels (segmentation courante)  $L$  a un support discret  $\Omega \subset \mathbb{N}^3$ . La fonction  $V : \mathbb{N} \rightarrow \mathbb{R}$  associant une intensité à un label (voir ci-dessous) sera appliquée sur la carte de segmentation. Toutefois, cette opération donne la fonction  $V(L) : \Omega \rightarrow \mathbb{R}$  avec  $V(L)(p) \triangleq V(L(p))$  pour tout  $p \in \Omega$ . La fonction  $V(L)$  est prolongée dans le domaine continu

$$V(L)^c \triangleq \Pi * \sum_{p \in \Omega} V(L(p)) \delta_p \quad (4.4)$$

où  $\Pi : \mathbb{R}^3 \rightarrow \mathbb{R}$  est la fonction porte tridimensionnelle. Exprimé plus simplement,  $V(L)^c$  est une image formée de petits cubes correspondant à chaque voxel.

#### 4.4.2 Processus d'acquisition

**Motivations** De nombreux systèmes de segmentation comparent directement des valeurs d'intensité<sup>7</sup> à l'intensité de voxels de l'image acquise  $I_0$ . En d'autres termes, elles comparent explicitement, ou implicitement, une image segmentée discrète  $V(L)$  à l'image acquise  $I_0$ . C'est le cas, par exemple, de toutes les approches travaillant sur l'histogramme de l'image. Cette manière de faire est erronée car elle ne tient pas compte du processus d'acquisition de l'image. En effet, il s'agit d'une approximation qui n'est valable que s'il n'y a pas de variation brusque dans le signal continu représentant la réalité sous-jacente de l'image acquise  $I_0$ . Or, lorsqu'on segmente une image, on postule implicitement que la réalité sous-jacente est composée de plages homogènes, donc que celle-ci est constante par morceaux. Par conséquent, on trouvera toujours des brusques variations dans la réalité sous-jacente des images à segmenter. Quelle que soit la résolution choisie, cette approche conduira à des voxels inclassifiables, conduisant au phénomène de « volumes partiels »<sup>8</sup>.

**Modélisation du processus d'acquisition** Nous allons donc modéliser plus rigoureusement le processus d'acquisition. La réalité sous-jacente (le cerveau réel) est représentée par une image  $R : \mathbb{R}^3 \rightarrow \mathbb{N}$  qui, à chaque position de l'espace continu associe un label. L'acquisition est modélisée en plusieurs étapes : d'abord en associant une intensité à un label à l'aide de la fonction  $V_0$ . En IRM, par exemple, cette fonction  $V_0$  est déterminée par la modalité utilisée. Ensuite, on modélise la formation de l'image par un filtre  $f_0$ , qui correspond à la réponse impulsionnelle du système d'acquisition (voir figure 4.7). Ce filtre  $f_0$  comprend le filtrage passe-bas associé à toute représentation discrète<sup>9</sup>, à travers l'échantillonnage. Notons que ce modèle reste simplificateur par rapport à la réalité, particulièrement dans le cas de l'IRM. En effet, l'acquisition IRM est parfois effectuée entièrement dans le domaine de Fourier, ou, dans d'autre cas, par coupes successives dans le domaine de Fourier (voir l'introduction § 2.1, p. 17). Cependant, le modèle d'acquisition proposé est à la fois général et suffisamment souple pour tenir compte des effets essentiels liés à diverses techniques d'acquisition. Des aspects plus

<sup>7</sup>Par exemple, segmenter une image en la seuillant revient à comparer une valeur d'intensité (le seuil) aux valeurs des voxels de l'image.

<sup>8</sup>Le terme « volume partiel », utilisé dans le milieu médical, évoque une vision d'un voxel « contenant » un « mélange » de plusieurs tissus. Cette vision est erronée en termes de traitement du signal. Il s'agit d'une approximation dans laquelle le filtrage anti-repliement serait effectué à l'aide d'un filtre porte (*sinc* dans le domaine fréquentiel).

<sup>9</sup>Même si nous utilisons une *notation* dans le domaine continu, l'acquisition produit une image discrète, qui aura subi un filtrage passe-bas lié à l'échantillonnage.

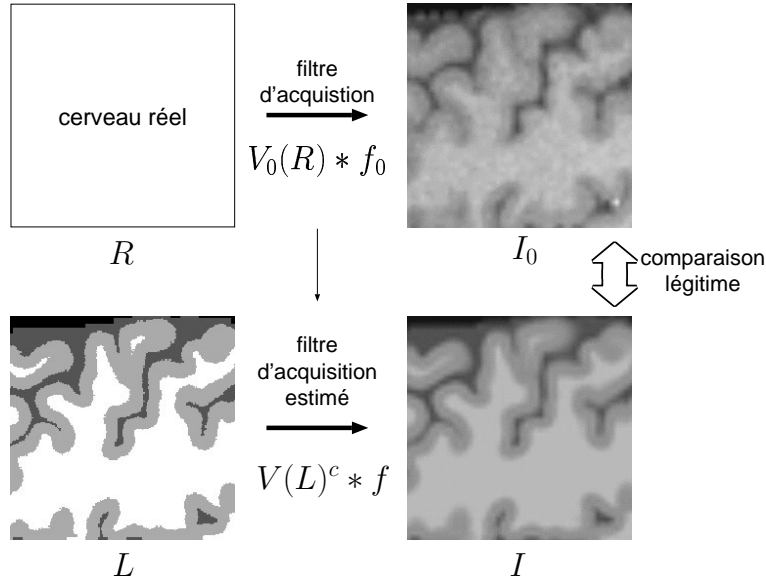


FIG. 4.7 – Pour obtenir une segmentation sous-voxel et tenir compte des effets de « volumes partiels », il est nécessaire de modéliser rigoureusement le processus d’acquisition. La plupart des approches comparent directement  $I_0$  et  $V(L)$ , ce qui n’est pas légitime. D’autres approches modélisent les « volumes partiels » en termes de mélange de classes, ce qui revient à imposer un filtre d’acquisition dont la réponse impulsionnelle serait une fonction porte. Le modèle proposé ici est plus général.

spécifiques à une technique d’acquisition, comme l’artefact RF en IRM, peuvent être ajoutés à ce modèle (voir § 4.4.5, p. 69).

**Application à la segmentation** Maintenant que nous avons modélisé le processus d’acquisition, nous pouvons l’intégrer dans notre approche de segmentation. Pour cela, nous allons reconstruire une image  $I$  à partir de la segmentation courante  $L$  qui pourra être comparée en toute rigueur avec l’image acquise  $I_0$ . Nous estimons d’abord une fonction  $V$  qui associe une intensité à un label (la méthode pratique d’estimation est expliquée au paragraphe 4.6.1, p. 85). Ensuite nous estimons le filtre d’acquisition  $f$ . Nous avons choisi pour cela un filtre de Butterworth paramétré par son ordre et sa fréquence de coupure. Une estimation acceptable de la fréquence de coupure est obtenue par le rapport entre la résolution de  $L$  et celle de  $I_0$ . L’ordre du filtre pourrait être estimé à partir du spectre de  $I_0$ , mais nous avons constaté que ce paramètre n’était pas critique et que l’estimation n’était pas fiable. Il est choisi constant  $n_f = 2$  en se fondant sur des observations. On définit donc la relation entre l’image de labels  $L$  et l’image reconstruite  $I$  :

$$I \triangleq V(L)^c * f \quad (4.5)$$

Ce modèle d’acquisition est bien adapté aux images de résolution anisotropes. Les images anisotropes sont fréquentes en imagerie médicale, où les cliniciens ont l’habitude de travailler sur des séries de coupes épaisses (§ 2.1.4, p. 19). Il suffit alors d’adapter le filtre  $f$ . La fréquence de coupure dans chaque direction est choisie de manière à correspondre au rapport de résolution dans la direction correspondante. Il serait aussi aisé d’adapter  $f$  à différents profils de coupe (carré, gaussienne...). Il est important de noter que, quelle que soit la résolution de l’image à segmenter  $I_0$ , la segmentation  $L$  s’effectue toujours dans la même résolution, isotrope, indépendante de la résolution de l’image  $I_0$ .

Le modèle décrit ci-dessus permet donc de tenir compte des effets dits de volumes partiels. Il s'appuie sur un modèle qui s'adapte aux différents types d'acquisition. En particulier, il est bien adapté aux différentes formes d'IRM (types d'acquisition et modalités). Il est également envisageable, par exemple, d'utiliser simultanément plusieurs contraintes (décrites ci-dessous) associées à des images de modalités différentes<sup>10</sup>. Chaque contrainte associerait alors une intensité différente (fonction  $V$ ) aux labels, et pourrait avoir des paramètres d'acquisition  $f$  différents.

### 4.4.3 Contrainte image à échelle variable

L'image reconstruite  $I$ , décrite précédemment, nous permet de définir la contrainte image. L'énergie associée à la contrainte image est une énergie quadratique simple, incluant d'un filtre de lissage  $h$  :

$$E \triangleq \|(I - I_0) * h\|^2 \quad (4.6)$$

Dans le cas particulier où  $h = \delta_0$  (Dirac), on obtient une contrainte qui permet un positionnement précis, sous-voxel, des surfaces segmentées. Le profil énergétique d'une telle contrainte est irrégulier, présentant de nombreux minima locaux. Si on n'utilise qu'une seule contrainte image, avec  $h = \delta$ , la segmentation pourrait se focaliser sur des détails de l'image et ne pas prendre en compte des erreurs à plus large échelle. Nous allons donc associer plusieurs contraintes image, chacune utilisant un lissage  $h$  différent, et permettant une convergence globale, sans empêcher la prise en compte des détails. Cette approche est analogue à une approche multi-résolution (voir figures 4.8 et 4.9).

### 4.4.4 Développement analytique de la variation d'énergie $\Delta E_l$

Calculons la variation d'énergie  $\Delta E_l(p)$  lorsque le label  $L(p)$  change de  $l_0$  à  $l$ . Soit  $I'$  l'image reconstruite, après ce changement. D'après la définition de l'image reconstruite  $I$  (équation 4.5) et la définition de  $V(L)^c$  (équation 4.4) on a :

$$\begin{aligned} I' &= \Pi * \left( \sum_{x \in \Omega, x \neq p} V(L(x)) \delta_x + V(l) \delta_p \right) * f \\ &= I + (V(l) - V(l_0)) \cdot \Pi * f * \delta_p \end{aligned}$$

D'après la définition de l'énergie  $E$  on a :

$$\begin{aligned} \Delta E_l(p) &= \|(I_0 - I') * h\|^2 - \|(I_0 - I) * h\|^2 \\ &= \int ((I_0 - I') * h)^2 - \|(I_0 - I) * h\|^2 \end{aligned}$$

<sup>10</sup>Nous supposons, dans cette segmentation multimodale, que les images ont été recalées entre elles.

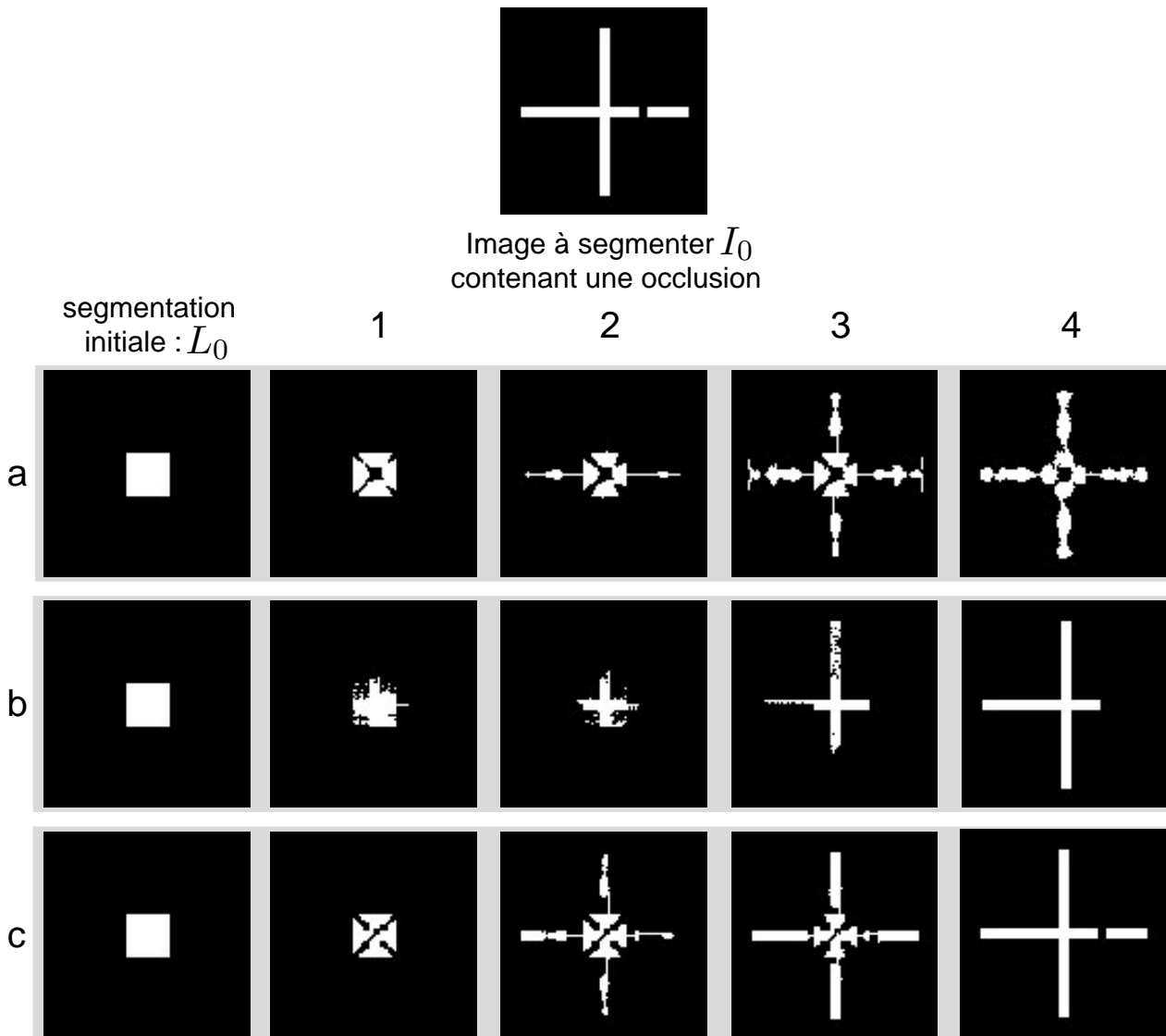


FIG. 4.8 – En associant plusieurs contraintes images, à des échelles différentes, on peut obtenir à la fois une convergence globale et une convergence locale par rapport à l'image à segmenter  $I_0$ . Cette approche est analogue à une approche multi-résolution. L'évolution ci-dessus n'utilise que des contraintes image. L'image à segmenter  $I_0$  est une croix qui a été dégradée par une occlusion. La ligne (a) montre une évolution employant une seule contrainte image avec un facteur de lissage  $h$  important. La segmentation converge globalement vers la croix en entier, mais le résultat n'est pas très précis. La ligne (b) montre une évolution employant une seule contrainte image sans facteur de lissage ( $h = \delta$ ). La segmentation converge précisément vers la croix, mais est bloquée dans son évolution par l'occlusion. La ligne (c) montre une évolution employant simultanément les deux contraintes image précédentes. La segmentation converge à la fois globalement et avec précision. Remarquez que le facteur de lissage  $h$  influe aussi sur la dynamique d'évolution : à la ligne (b), sans lissage, l'évolution se concentre sur des petits détails, parvenant alors plus lentement vers la forme générale de la croix.

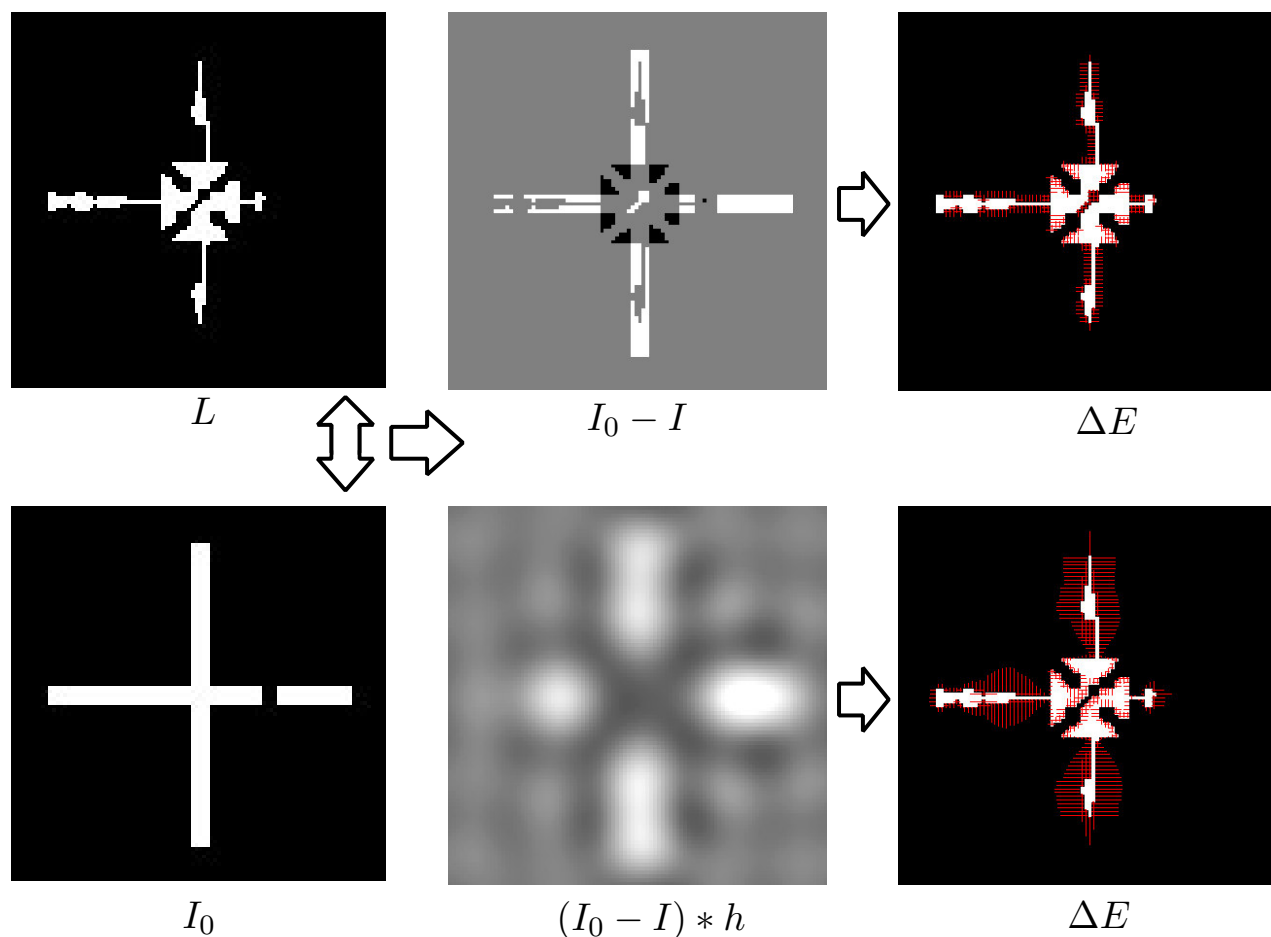


FIG. 4.9 – Variation d'énergie  $\Delta E$  associée à deux contraintes image à des échelles différentes. Cet exemple fait partie de l'évolution (c) décrite à la figure précédente 4.8-(c). L'image  $I_0$  est segmentée en utilisant uniquement deux contraintes images : l'une sans lissage et l'autre avec un fort lissage  $h$ . Le filtre d'acquisition est un filtre unité ( $\delta$ ). Les images de droite montrent, à l'aide de flèches rouges, la valeur de chaque contrainte  $\Delta E$ . L'amplitude des variations d'énergie  $\Delta E$  associées à la contrainte avec lissage est plus grande. Celle-ci tient compte d'erreurs sur une région étendue, contrairement à la contrainte sans lissage, qui ne considère que le voisinage immédiat.



En introduisant l'expression de  $I'$  :

$$\begin{aligned}
\Delta E_l(p) &= \int \left( (I_0 - I) * h - (V(l) - V(l_0)) \cdot \Pi * f * h * \delta_p \right)^2 - \|(I_0 - I) * h\|^2 \\
&= \int \left( (I_0 - I) * h \right)^2 + (V(l) - V(l_0))^2 \int (\Pi * f * h * \delta_p)^2 \\
&\quad - 2(V(l) - V(l_0)) \int (I_0 - I) * h \cdot (\Pi * f * h * \delta_p) - \|(I_0 - I) * h\|^2 \\
&= (V(l) - V(l_0))^2 \|\Pi * f * h\|^2 \\
&\quad - 2(V(l) - V(l_0)) \int_x [(I_0 - I) * h](x) \cdot [\Pi * f * h](x - p) dx
\end{aligned}$$

La fonction  $[\Pi * f * h]$  étant symétrique, on reconnaît dans le terme de droite un produit de convolution. On obtient finalement l'expression suivante, sans approximations, de la variation d'énergie due à la contrainte image :

$$\Delta E_l(p) = (V(l) - V(l_0))^2 \|\Pi * f * h\|^2 - 2(V(l) - V(l_0)) [(I_0 - I) * \Pi * f * h * h](p) \quad (4.7)$$

### Calcul pratique de la variation d'énergie $\Delta E_l$

L'apparition du terme de convolution sur toute l'image dans le calcul aboutissant à l'équation 4.7 est très intéressante. Elle permet d'effectuer le calcul de la variation d'énergie  $\Delta E_l$  pour tous les points  $p \in \Omega$  de l'image simultanément, et sans faire d'approximations.

Le calcul pratique du terme  $[(I_0 - I) * \Pi * f * h * h](p)$  de l'équation 4.7 s'effectue en trois étapes :

1. À la première étape, on calcule  $I \triangleq V(L)^c * f$  (équation 4.5) à partir de  $L$ . Ce calcul est exigeant en temps de calcul et en mémoire car il traite des images représentées dans la résolution élevée de  $L$  (voir § 5.1.5, p. 97). La segmentation  $L$  (haute résolution) est transformée en une image  $V(L) : \mathbb{N}^3 \rightarrow \mathbb{R}$ . Le support fréquentiel de  $I$  étant limité par le filtrage  $f$ , l'image continue  $I$  vérifie les conditions de Shannon pour la résolution d'acquisition. L'image  $I$  peut donc être représentée dans le domaine discret à la même résolution que la résolution d'acquisition. Le calcul de  $I$  (passage discret au continu et filtrage par  $f$ , équation 4.5), et le sous-échantillonnage, sont effectués simultanément dans le domaine de Fourier<sup>11</sup>. Nous avons constaté que l'existence d'algorithmes très optimisés de transformée de Fourier rapide rendent le filtrage dans le domaine de Fourier plus performant que la convolution spatiale dans tous les cas non triviaux. Le filtrage dans le domaine de Fourier requiert cependant de gérer les problèmes qui peuvent survenir aux bords de l'image. Remarquons que, dans le cas où plusieurs contraintes image sont utilisées simultanément à des échelles différentes, l'image  $I$  n'a pas besoin d'être recalculée.
2. La deuxième étape, le filtrage de  $I - I_0$  par  $[\Pi * f * h * h]$  pour obtenir  $[(I_0 - I) * \Pi * f * h * h]$  s'effectue en basse résolution, et ne pose pas de problèmes particuliers.
3. La dernière étape consiste à prendre la valeur  $[(I_0 - I) * \Pi * f * h * h]$  (dont la représentation discrète est en basse résolution) à chaque point  $p$  de la frontière (donc en haute résolution). On utilise pour cela une interpolation spline de degré 3 (voir § 7.3.5, p. 122).

<sup>11</sup>Une solution élégante alternative était envisageable :  $V(L)$  étant constante par morceaux on pourrait effectuer le calcul sur  $\frac{\partial V(L)^c}{\partial x}$  qui est une image creuse. Mais en pratique, l'approche choisie est plus simple, et plus souple.

**Normalisation de l'énergie** La définition de l'énergie donnée à l'équation 4.6 n'est pas normalisée. C'est-à-dire que si on multiplie l'image source  $I_0$  par un facteur  $a$ , l'énergie  $E(L)$  sera multipliée par un facteur  $a^2$ . On souhaite que le poids de la contrainte soit indépendant de l'image source. On multipliera donc l'énergie par un facteur de normalisation  $\frac{1}{\sigma^2 \|\Pi * f * h\|^2}$ ,  $\sigma^2$  étant la variance de l'image.

### Paramétrage par l'atlas

Plusieurs possibilités existent pour rendre certains paramètres de cette contrainte dépendants de la position (atlas : § 5.2.1, p. 100). L'intensité  $V(l)$  associée à un label  $l$  peut être rendue dépendante de la position dans un atlas, ce qui peut être utilisé pour modéliser des tissus dont l'intensité IRM est plus complexe. L'influence  $\alpha$  de la contrainte image peut aussi dépendre de la position. Ceci permettrait, par un apprentissage statistique, de diminuer l'importance de cette contrainte dans les régions bruitées, où elle serait moins fiable. Dans le système actuel, ces possibilités sont implémentées, mais ne sont pas nécessaires dans son application sur les sillons corticaux.

#### 4.4.5 Correction de l'artefact RF apparaissant en IRM

Lors de la description du processus d'acquisition (§ 4.4.2), nous avons évoqué la possibilité d'étendre le modèle d'acquisition pour tenir compte des spécificités d'une technique d'acquisition particulière. En IRM, le gain de l'antenne radiofréquence peut varier spatialement, donnant lieu à un artefact multiplicatif à basse fréquence spatiale (voir 2.1.5). Nous obtenons une première estimation de ce champ en calculant le rapport  $\frac{I}{I_0}$  sur une partie de l'image relativement stable (MB érodée). Ensuite ce champ est lissé et étendu sur le reste de l'image. Le champ multiplicatif ainsi calculé est appliqué alors sur l'image reconstruite  $I$ . Cette méthode permet d'obtenir efficacement les champs RF simulés du modèle numérique BrainWeb [10, 15] (§ 5.1.2, p. 91).

## 4.5 Contraintes *a priori*

Les contraintes associées au modèle *a priori* sont des contraintes qui ne dépendent pas de l'image source. Elles représentent les connaissances dont on dispose sur la géométrie (ou la forme) des objets à segmenter.

Comme cela a été exposé dans l'introduction de ce chapitre, nous souhaitons contraindre fortement la segmentation par ces connaissances. Dans notre approche, ces connaissances s'expriment par des termes supplémentaires dans l'énergie globale (équation 4.1). On doit alors construire cette formulation énergétique des connaissances à partir de leur formulation intuitive. La formulation énergétique doit non seulement être capable de distinguer ce qui est « bon » (faible énergie) de ce qui est « mauvais » (forte énergie), mais surtout, elle doit fournir le moyen d'évoluer entre l'un et l'autre. En d'autres termes, l'énergie  $E_i$  ne doit pas présenter de trop forts minima locaux, et l'expression de la variation d'énergie  $\Delta E_i$  doit pouvoir être déduite de  $E_i$ , même si cela implique de faire des approximations.

### Localité / non localité

Pour certaines contraintes, nous verrons apparaître deux termes dans la variation d'énergie. Un terme local, dans lequel le changement en  $p$  provoque une variation de la contribution en  $p$

à l'énergie totale. Un terme non local, dans lequel le changement en  $p$  provoque une variation de la contribution en d'autres points que  $p$  à l'énergie totale.

### Définition de la distance à une surface

Dans la description des contraintes qui suivent, nous allons utiliser des cartes de distance semblables aux transformées de distance, aussi appelées distance de chanfrein. Rappelons tout d'abord la définition de la distance de chanfrein  $D_{O_k}^c(p)$  en un point  $p \in \mathbb{Z}^3$ , à l'ensemble de points  $O_k$  :

$$D_{O_k}^c(p) \triangleq \begin{cases} 0 & \text{si } p \in O_k \\ \min\{D_{O_k}^c(p') + M^{26}(p' - p) \mid p' \in N^{26}(p)\} & \text{sinon} \end{cases} \quad (4.8)$$

Ici,  $M^{26}$  sont les coefficients de chanfrein associés au voisinage  $N^{26}$  (voir notations, tableau 4.1).

Les distances que nous allons définir sont des distances à des surfaces (discrètes) au lieu d'être des distances à un ensemble de points. Les distances à une surface sont plus adaptées au calcul de l'épaisseur d'un objet en un point de sa surface (voir § 4.5.2, p. 76). Elles permettent aussi de calculer simultanément la distance à l'intérieur et à l'extérieur d'un objet  $O_k$  en fournissant des valeurs plus cohérentes que celles qui seraient fournies par ce double calcul utilisant la transformée de distance classique.

Soit  $S_k$  la surface entourant le label  $k$  (figure 4.3), nous définissons alors la distance  $D_{S_k}$  à cette surface de la même manière que la distance de chanfrein :

$$D_{S_k}(p) \triangleq \begin{cases} d(p, S_k) & \text{si } d(p, S_k) \leq \sqrt{\frac{3}{2}} \\ \min\{D_{S_k}(p') + M^{26}(p' - p) \mid p' \in N^{26}(p)\} & \text{sinon} \end{cases} \quad (4.9)$$

où  $d(p, S_k)$  est la distance euclidienne du point  $p$  à la surface  $S_k$ . Le point  $p$  et la surface  $S_k$  sont, ici, plongés dans  $\mathbb{R}^3$  (voir figure 4.10). Par exemple, pour  $p$  directement adjacent à la surface  $S_k$ ,  $d(p, S_k) = 0,5$ . A l'aide de la distance à la surface  $S_k$  nous pouvons aussi définir une distance  $D_{O_k}$  à l'objet  $O_k$  par :

$$D_{O_k}(p) \triangleq \begin{cases} 0 & \text{si } p \in O_k \\ D_{S_k}(p) & \text{sinon} \end{cases} \quad (4.10)$$

## 4.5.1 Contraintes : distribution relative

### Introduction

La contrainte de distribution relative modélise la probabilité d'observer un label à une distance donnée d'un autre label. Elle traduit la notion intuitive, selon laquelle autour d'un label donné (c'est-à-dire, à une certaine distance de celui-ci) certains labels sont plus probables que d'autres. Elle joue un rôle important dans la segmentation des sillons corticaux pour modéliser la distribution de matière grise autour du liquide. Dans cet exemple, elle a une fonction comparable aux surfaces couplées utilisées dans Zeng *et al.* [130] (voir § 3.2.3, p. 51). Par ailleurs, elle peut être vue comme une extension de la notion de voisinage utilisée dans les champs de Markov à un voisinage non forcément local. Cette contrainte n'est pas limitée à la modélisation de distributions comme celles des sillons corticaux et peut être utilisée dans d'autres cas.

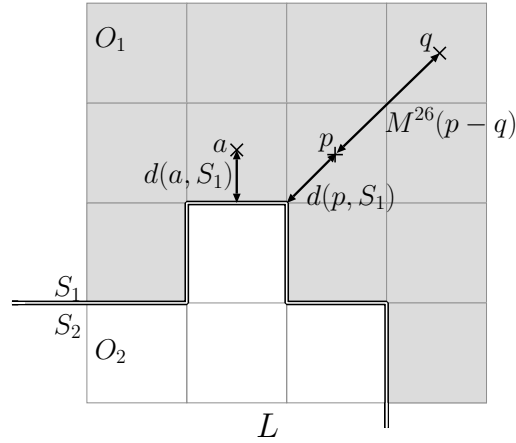


FIG. 4.10 – Au lieu d'utiliser la transformée de distance de chanfrein à un ensemble de points, nous définissons une distance à une surface. La distance en des points  $q$  éloignés de la surface est calculée de la même manière que pour la distance de chanfrein. En revanche, la distance des points proches de la surface ( $p$  et  $a$ ) est définie comme la distance euclidienne du centre du voxel à la surface.

Chaque instance de cette contrainte modélise la distribution autour d'un label  $k$  donné. Plusieurs instances de cette contrainte pourront être utilisées simultanément pour modéliser la distribution autour de plusieurs labels différents (voir § 4.3, p. 61).

### Définition

Pour alléger les notations, nous noterons  $D_k \triangleq D_{O_k}$ . Nous définissons l'énergie de la contrainte de distribution relative par :

$$E \triangleq \sum_{p \in \Omega} -\log(P(L(p)|D_k(p))) \quad (4.11)$$

où  $P(L(p)|D_k(p))$  est la probabilité d'observer le label  $L(p)$  à une distance extérieure  $D_k(p)$  de l'objet  $O_k$ . Remarquons que si  $P$  est gaussien ou asymptotiquement gaussien (par exemple la fonction erfc, voir § 4.5.1, équation 4.16), alors  $E$  prend une forme quadratique. Cette propriété est souhaitable pour la définition d'une énergie.

### Variation d'énergie

Soit  $O'_k$  l'objet formé par le label  $k$ , après le changement de  $L(p)$  de  $l_0$  en  $l$ . Soit  $D'_k$  la distance à  $O'_k$ . Calculons la variation d'énergie associée à la contrainte de distribution relative :

$$\Delta E_l(p) = -\log\left(\frac{P(l|D'_k(p))}{P(l_0|D_k(p))}\right) + \sum_{x \neq p} -\log\left(\frac{P(L(x)|D'_k(x))}{P(L(x)|D_k(x))}\right) \quad (4.12)$$

Le premier terme est local, il représente la variation de la contribution du point  $p$  à l'énergie totale. En revanche, le deuxième terme est non local. Si on suppose que la carte de distance  $D_k$  ne varie pas, ce deuxième terme, non local, est nul. Ce terme non local représente la variation de l'énergie provoquée par la modification de la carte de distance. Ce terme sera particulièrement important pour des points présentant une forte convexité de l'objet  $O_k$  (voir figure 4.11)

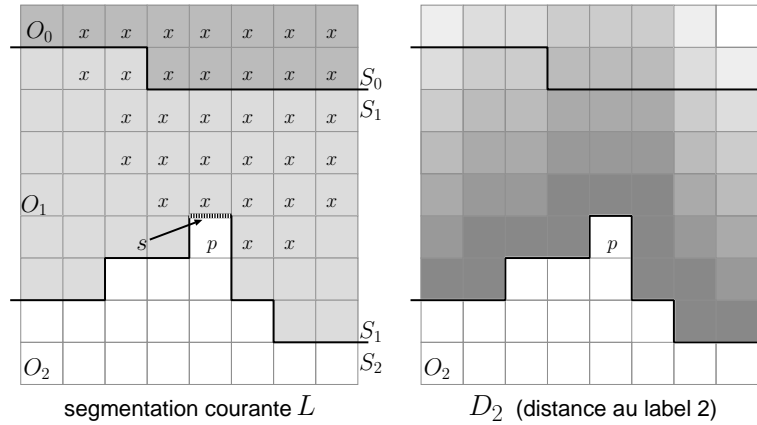


FIG. 4.11 – La modification du label au point  $p$  se trouvant sur une partie convexe de  $O_2$ , provoque un changement dans la carte de distance  $D_2$  sur un nombre important de points  $x$ . En effet, si  $L$  change de  $2 \rightarrow 1$  en  $p$ , alors la distance au label 2 ( $D_2$ ) va diminuer légèrement sur tous les points  $x$ . Ces changements constituent le terme non local de la variation d'énergie.

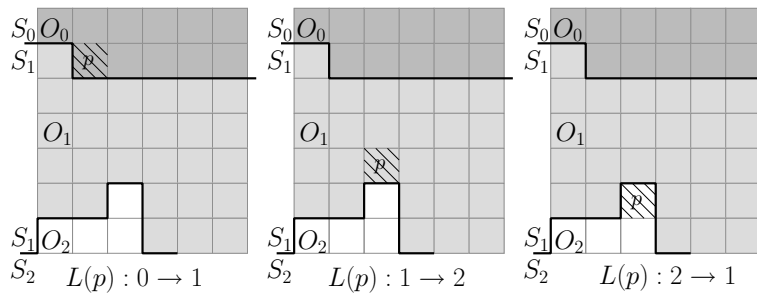


FIG. 4.12 – Calcul de la variation d'énergie d'une contrainte de distribution relative autour du label 2. On peut distinguer 3 cas par rapport à la position  $p$  du changement. Cas-1 (à gauche) : le changement n'affecte pas le label 2, la distance extérieure à l'objet  $O_2$  est inchangée, et le terme non local est nul. Cas-2 (au centre) : le changement fait grandir  $O_2$ , déplace  $S_2$  et modifie donc la distance extérieure  $D_2$  à  $O_2$ . Cas-3 (à droite) : le changement fait rétrécir  $O_2$ , déplace  $S_2$  et modifie donc la distance extérieure  $D_2$  à  $O_2$ .

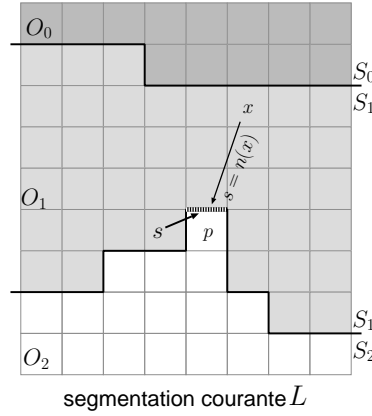


FIG. 4.13 – Calcul du terme non local. L'élément de surface  $s = n(x)$  est l'élément de surface le plus proche de  $x$ . On suppose alors que seul un changement de label en  $p$ , voisin de  $s$ , est susceptible de modifier la distance  $D_2(x)$  au label 2 en  $x$ .

Considérons les différents cas possibles pour le changement en  $p$  (voir figure 4.12). Dans le **premier cas**, le plus simple,  $p$  n'est pas adjacent à  $S_k$ . Le changement n'affecte donc pas le label  $k$  ( $l_0 \neq k$  et  $l \neq k$ ). La surface  $S_k$  ne change pas et les cartes de distance non plus :  $D'_k = D_k$ . Le terme non local est alors nul. Dans les deux autres cas,  $p$  est adjacent à la surface  $S_k$ . Le changement affecte donc le label  $k$  et le terme non local n'est plus nul. Dans le **deuxième cas**  $p$  est adjacent à la surface  $S_k$  et se trouve à l'extérieur de  $O_k$ . Le basculement ( $l = k$ ) fait grandir l'objet  $O_k$ , et  $p$  se retrouve dans  $O'_k$ . On a donc  $D'_k(p) = 0$ ,  $P(L(p) = k | D'_k = 0) = 1$ . Avant le changement, le point  $p$  se situait juste à l'extérieur de  $O_k$ , donc  $D_k(p) = 0,5$ . Dans le **troisième cas**,  $p$  est adjacent à la surface  $S_k$  et se trouve à l'intérieur de l'objet  $O_k$ . Le basculement ( $l \neq k$ ) fait rétrécir  $O_k$  et  $p$  se retrouve, après le changement, juste extérieur à  $O'_k$ . On a donc  $D'_k(p) = 0,5$ . Remarquons que dans ce cas  $l_0 = k$ , et donc que  $P(L(p) = k) = 1$ .

Pour résumer, on a :

$$\Delta E_l(p) = \begin{cases} -\log\left(\frac{P(l | D'_k(p))}{P(l_0 | D_k(p))}\right) & \text{si } l_0 \neq k \text{ et } l \neq k \\ \log(P(l_0 | D_k = 0,5)) + \sum_{x \neq p} -\log\left(\frac{P(L(x) | D'_k(x))}{P(L(x) | D_k(x))}\right) & \text{si } l_0 \neq k \text{ et } l = k \\ -\log(P(l | D_k = 0,5)) + \sum_{x \neq p} -\log\left(\frac{P(L(x) | D'_k(x))}{P(L(x) | D_k(x))}\right) & \text{si } l_0 = k \text{ et } l \neq k \end{cases} \quad (4.13)$$

L'évaluation du terme non local  $\sum_{x \neq p} -\log\left(\frac{P(L(x) | D'_k(x))}{P(L(x) | D_k(x))}\right)$  pose des difficultés pratiques. Il serait rédhibitoire de recalculer  $D'_k$  pour tous les points  $p$ . Cela impliquerait de recalculer la carte de distance  $D_k$  pour chaque variation envisagée en chaque point de la frontière. Nous allons proposer une approximation de  $D'_k$  permettant le calcul rapide en tous les points de la frontière.

Lors du calcul de  $D_k$ , nous pouvons obtenir simultanément une information de voisinage qui nous permet d'associer à chaque point  $x \in \Omega$ , l'élément de surface  $n(x) \in S_k$  le plus proche de  $x$ . On suppose alors qu'un changement de  $L$  en  $p$  ne provoque de modifications de  $D_k$  qu'aux points  $n^{-1}(s)$ , où  $s$  est un élément de surface voisinant  $p$  (voir figure 4.13) :

$$D'_k(x) \simeq D_k(x) + \begin{cases} \Delta^a D_k(x) & \text{pour } x \in n^{-1}(s) \\ 0 & \text{ailleurs} \end{cases} \quad (4.14)$$

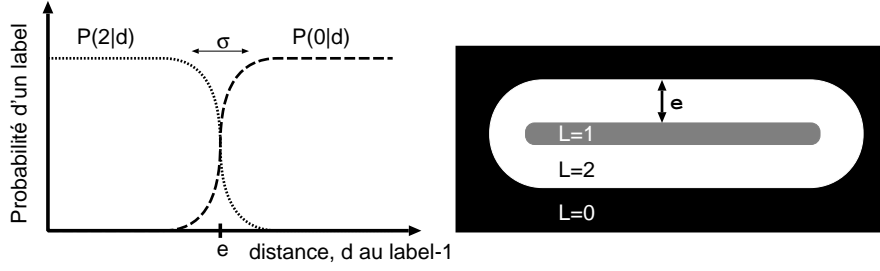


FIG. 4.14 – Distributions de probabilités utilisées pour les exemples de la figure 4.15. Remarque : dans cet exemple 2D les valeurs arbitraires des labels ne correspondent pas à celles utilisées dans l'application réelle correspondante (LCR :0 entouré de MG :1 puis de MB :2).

où  $\Delta^a D_k(x)$  est une approximation de la variation de distance en un point  $x$  lorsque le label au point  $p$  bascule. Cette approximation est décrite en annexe § C.1, p. 171.

Le terme non local devient alors :

$$\sum_{x \neq p} -\log \left( \frac{P(L(x)|D'_k(x))}{P(L(x)|D_k(x))} \right) \simeq \sum_{x \in n^{-1}(p)} -\log \left( \frac{P(L(x)|D_k(x) + \Delta^a D_k(x))}{P(L(x)|D_k(x))} \right) \quad (4.15)$$

où on note  $n^{-1}(p) = n^{-1}(\{s \in S_k | s \text{ adjacent à } p\})$ .

L'expression précédente permet de calculer rapidement la variation d'énergie  $\Delta E_l$  simultanément pour tous les points  $p$  de la frontière. Le calcul est effectué en parcourant tous les points  $x \in \Omega$  et en ajoutant la contribution à la variation d'énergie  $\Delta E_l(u)$  pour les points  $u$  voisins de  $s = n(x)$ .

## Exemples d'évolution

Trois exemples, illustrant la contrainte de distribution relative, sont décrits à la figure 4.15.

## Paramétrage par un atlas

Le paramétrage de la contrainte de distribution relative s'effectue au travers de la probabilité  $P(l|d)$ . Cette contrainte peut servir dans diverses situations. Elle est particulièrement adaptée pour décrire deux labels adjacents, dont l'un aurait une épaisseur à variation lente. Dans ce cas, la probabilité  $P(l|d)$  peut être spécifiée comme une fonction de forme « sigmoïde » (figure 4.14). On utilise alors la forme paramétrique suivante :

$$P(l|d) = \frac{1}{2} \operatorname{erfc} \left( \pm \frac{d-e}{\sqrt{2}\sigma} \right) \quad (4.16)$$

où les paramètres sont  $e$  et  $\sigma$ . La fonction  $\operatorname{erfc}$  est définie par  $\operatorname{erfc}(x) \triangleq \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt$ . Remarquons que  $-\log(\operatorname{erfc}(x)) = x^2(1 + o(\frac{1}{x}))$ . L'expression de l'énergie est donc asymptotiquement quadratique en  $x$ , ce qui justifie le choix de  $\operatorname{erfc}$  comme fonction sigmoïde. Les deux paramètres  $e$  et  $\sigma$  sont simples et intuitifs. Le premier correspond à l'épaisseur (figures 4.15 et 4.14) et le second à la largeur de la transition.

La probabilité  $P(l|d)$  peut faire l'objet d'un apprentissage décrit au paragraphe 5.2.1, p. 100.

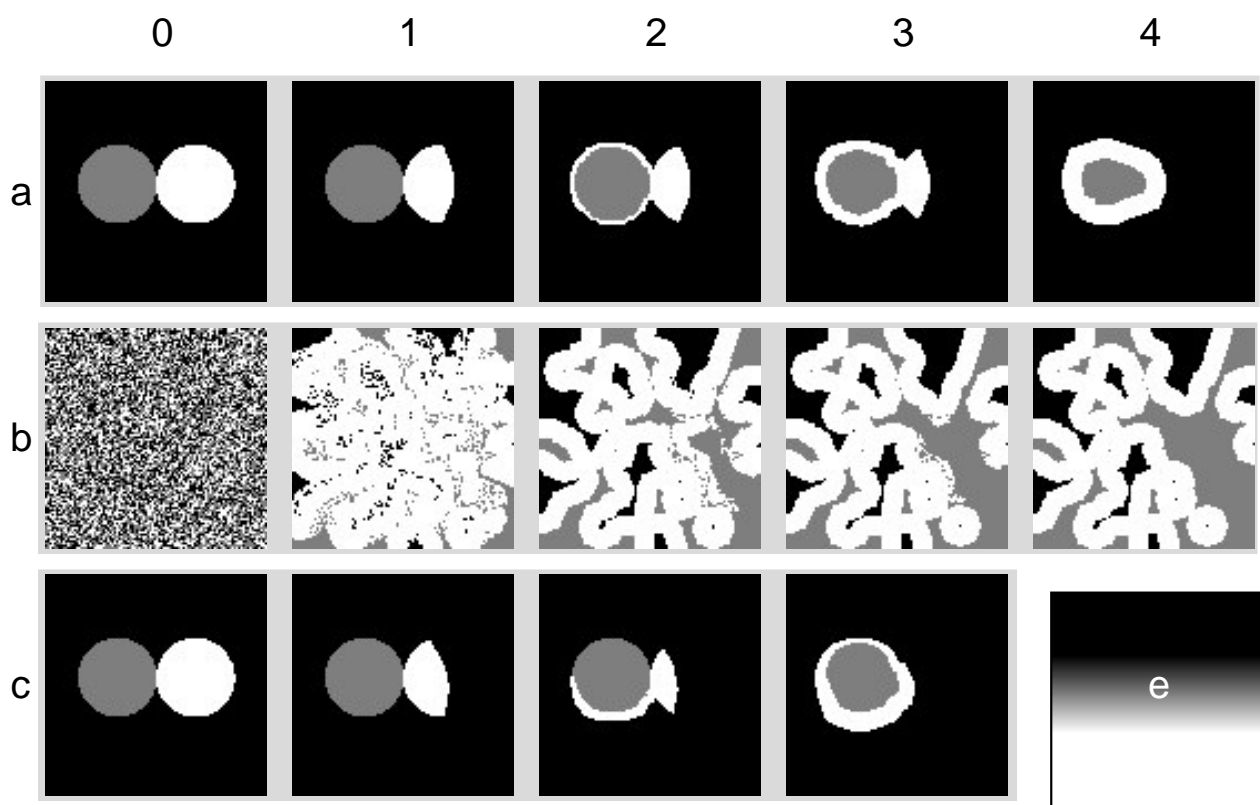


FIG. 4.15 – Trois exemples (a, b, c) d'évolution (0-1-2-3(-4)) sous le seul effet de la contrainte de distribution relative. La segmentation initiale se trouve à la colonne 0. Les distributions de probabilités utilisées sont présentés à la figure 4.14. Dans les deux premières évolutions (a, b), le paramètre  $e$  est constant. Dans la dernière évolution (c), le paramètre  $e$  est plus grand vers le bas de l'image. La valeur du paramètre  $e$  est donnée par l'image c-4 (blanc = valeur élevée, noir = valeur faible).

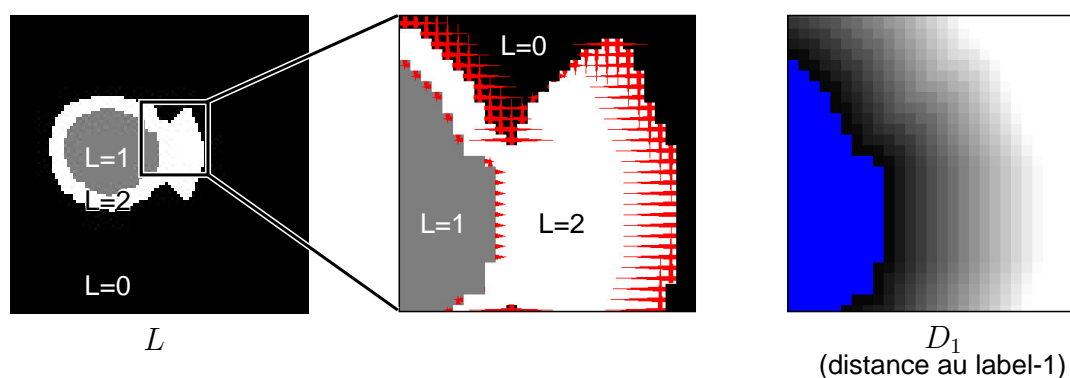


FIG. 4.16 – Agrandissement de l'exemple donné à la figure 4.15-a-3. Les flèches en rouge indiquent les diminutions d'énergie  $\Delta E(p)$  associées à chaque point de la frontière. Les variations d'énergie sur la frontière entre le label 1 et le label 2 sont dues au terme non local. Les variations sur la frontière 0-2 sont dues au terme local.



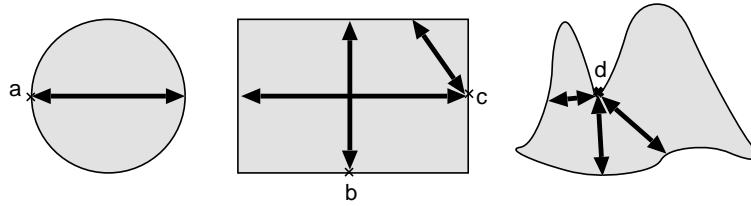


FIG. 4.17 – La définition de l'épaisseur d'un objet en un point de sa surface n'est pas triviale. Dans le cas d'une sphère l'épaisseur au point (a) peut être définie simplement comme le diamètre de la sphère. Dans le cas d'un rectangle, l'épaisseur au point (b) paraît intuitive, mais au point (c) il y a une ambiguïté. Sur un point d'un objet plus complexe (d) l'épaisseur est difficile à définir.

## 4.5.2 Contraintes : épaisseur

### Introduction

La contrainte de distribution relative, présentée précédemment, permet de décrire le voisinage (au sens large) d'un objet. Dans la disposition particulière des labels qui se produit dans les sillons corticaux, la contrainte de distribution relative permet, de fait, d'imposer une certaine épaisseur moyenne à la matière grise. Cependant, la contrainte de distribution relative n'est pas capable d'imposer une épaisseur à un label dans d'autres configurations plus générales. En particulier, il est souhaitable de pouvoir décrire l'épaisseur de liquide au centre des sillons. Nous savons que ces régions sont minces, et souhaitons apporter cette connaissance au système de segmentation. Néanmoins, le champ d'application de cette contrainte d'épaisseur est beaucoup plus large, et elle pourrait être utilisée conjointement à un atlas pour guider la segmentation des structures centrales du cerveau (noyaux, ventricules, etc.).

### Définition de l'épaisseur d'un objet

Pour élaborer une contrainte d'épaisseur, il faut tout d'abord définir ce qu'est l'épaisseur  $\xi(s)$  en un élément  $s$  de la surface d'un objet  $O_k$ . Si la notion d'épaisseur d'un objet paraît intuitive à premier abord, sa définition précise, et *a fortiori* sa définition en un point précis d'une surface, n'est pas évidente (voir figure 4.17).

Nous cherchons une définition de l'épaisseur qui ne soit pas sensible à des irrégularités de la surface. La notion d'irrégularité dépend du contexte géométrique. Une même forme géométrique, placée dans deux contextes différents pourra, dans un cas être considérée comme un irrégularité, et dans l'autre, comme une forme à part entière (voir figure 4.18).

**Trois définitions possibles de l'épaisseur** Nous allons présenter trois définitions possibles de l'épaisseur (voir figure 4.19), en présentant leurs avantages et inconvénients, et nous proposerons une définition plus générale englobant ces trois cas. On note  $\Gamma(O_k)$  l'ensemble des sphères maximales contenues dans  $O_k$ . Chaque sphère  $\gamma \in \Gamma(O_k)$  est définie par son centre  $\gamma_C$  et son rayon  $\gamma_R$ .

Une **première** définition simple de l'épaisseur en un point  $s$ , est le diamètre de la plus grande sphère contenue dans l'objet  $O_k$  et qui passe par  $s$ . Cette définition de l'épaisseur est intéressante, mais est sensible aux perturbations sur la surface de l'objet. Sur une irrégularité, comme celle de la figure 4.19 (irrégularité-1), l'épaisseur sera donnée par la sphère incluse dans l'irrégularité. Remarquons que la sphère  $\gamma$  associée à  $s$ , par cette définition, est une sphère

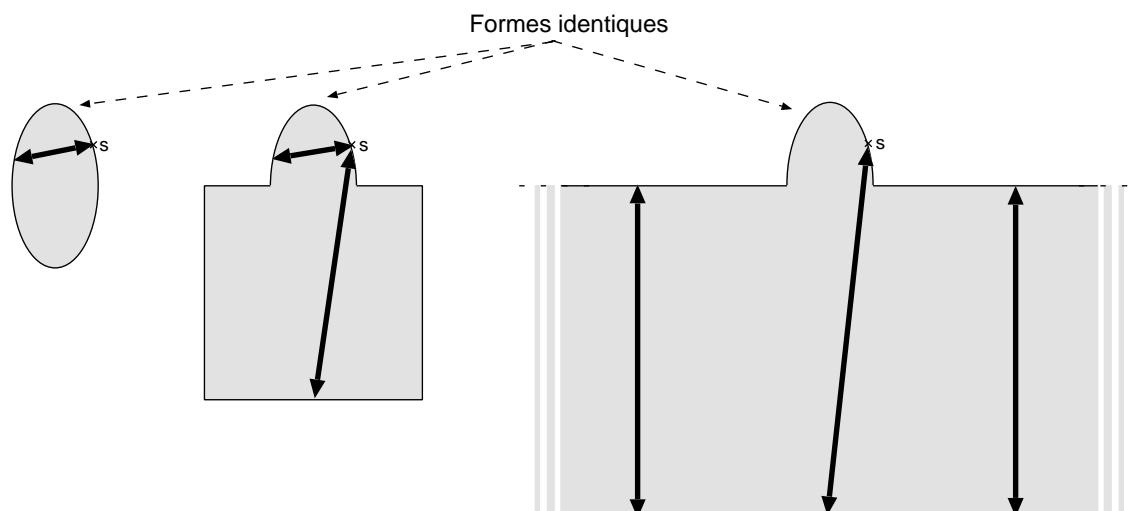


FIG. 4.18 – L'épaisseur en un point d'une forme dépend du contexte : sur la figure de gauche la forme arrondie est déterminante pour définir l'épaisseur, sur la figure de droite la forme arrondie peut être considérée comme étant une irrégularité sur une forme plus grande. La figure du milieu présente une situation ambiguë.

maximale et qu'elle appartient donc à  $\Gamma(O_k)$ . Rechercher la sphère  $\gamma$  à partir de  $s$  est exigeant calculatoirement, et demande de parcourir toutes les sphères de  $\Gamma(O_k)$ .

Une **deuxième** définition possible, plus légère calculatoirement, est de calculer la carte de distances au squelette  $\Gamma(O_k)$ . On se heurte alors à un problème majeur : l'instabilité intrinsèque au squelette. En effet, de très petites perturbations sur la surface de  $O_k$  peuvent faire apparaître des branches dans le squelette, rendant ainsi la définition de l'épaisseur très instable. Diverses techniques existent pour « nettoyer » le squelette. Elles sont motivées par la nécessité d'obtenir un squelette présentant certaines propriétés topologiques telle que la connexité. Dans notre application, ces propriétés ne sont pas importantes, au contraire, elles risquent d'induire des erreurs et rendraient la définition de l'épaisseur difficile à utiliser.

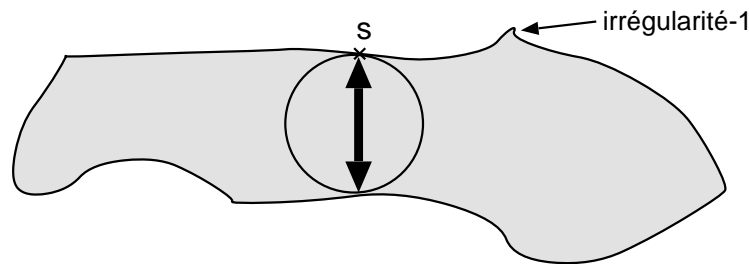
Une **troisième** possibilité est de définir l'épaisseur en  $s$  comme le diamètre de la plus grande sphère incluse dans l'objet. Cette définition est très robuste (elle n'est pas sensible aux irrégularités), mais manque de précision. Elle ne fournit qu'une épaisseur unique pour tout l'objet, ce qui n'est pas acceptable pour des objets complexes.

**Proposition de définition** En tenant compte des considérations précédentes, nous proposons une définition de l'épaisseur reposant sur l'analyse des sphères maximales  $\Gamma(O_k)$  de l'objet  $O_k$ <sup>12</sup>. Au lieu d'associer à chaque élément de surface  $s$ , la sphère maximale passant par  $s$  (première définition), nous allons choisir une sphère maximale répondant à des critères moins contraignants (Figure 4.20). Pour un élément  $s$  donné de la surface, nous associons un coût à chaque sphère maximale :

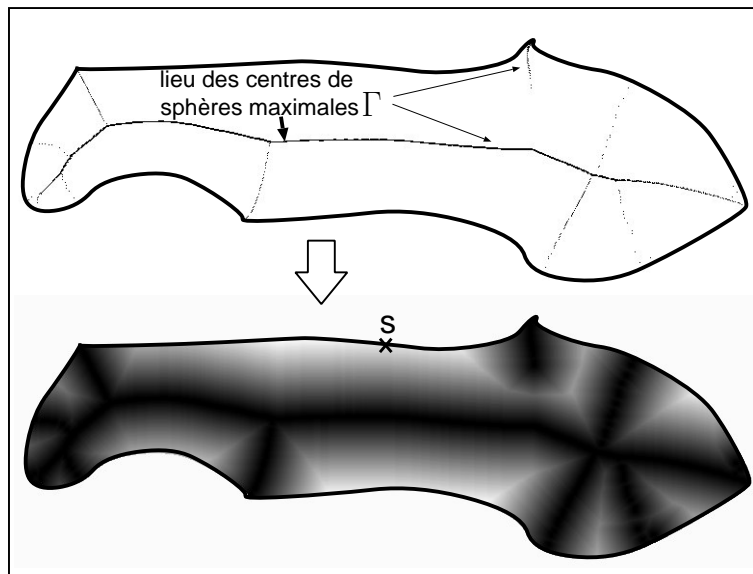
$$\zeta(\gamma_R, d(s, \gamma_C)) = d(s, \gamma_C) - \alpha\gamma_R \quad (4.17)$$

où  $d(s, \gamma_C)$  est la distance de  $s$  à  $\gamma_C$ . L'épaisseur en  $s$  est alors calculée à partir de la sphère minimisant le coût  $\zeta$ . On impose aussi un coût infini si  $\gamma_C$  n'est pas visible de  $s$  (si le segment

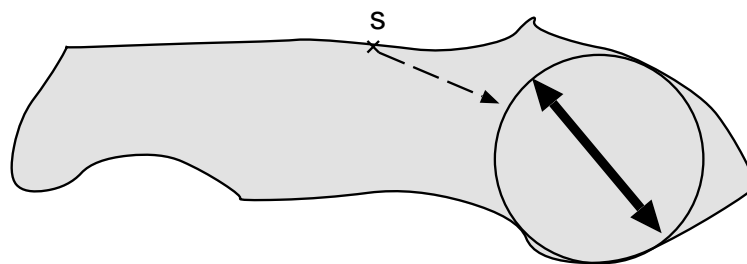
<sup>12</sup>Notre définition n'est pas calculatoirement moins exigeante que la première définition envisagée (sphère maximale passant par  $s$ ), mais elle n'est pas sensible aux perturbations sur la surface, tout en restant précise.



définition 1 : diamètre de la plus grande sphère passant par  $s$



définition 2 : diamètre de la sphère maximale la plus proche de  $s$   
(=  $2 \times$  distance au squelette)



définition 3 : diamètre de la plus grande sphère comprise dans l'objet

FIG. 4.19 – Trois définitions possibles pour l'épaisseur en un point  $s$  de la surface d'un objet. La première définition est sensible aux irrégularités sur la surface (voir texte). La deuxième définition est rapide et simple à calculer, mais très sensible aux irrégularités de la surface. Par exemple, dans la partie droite de l'objet, si le squelette était nettoyé des trois branches, pour ne laisser que l'axe central, alors la distance au squelette donnerait correctement l'épaisseur. La troisième définition est très robuste, mais ne donne qu'une épaisseur unique pour tout l'objet. Ces trois définitions apparaissent comme des cas particuliers de la définition de l'épaisseur que nous proposons.

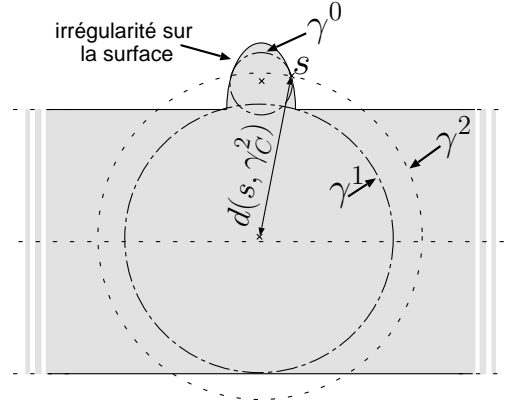


FIG. 4.20 – La définition de l'épaisseur en  $s$ , à l'aide de la sphère maximale  $\gamma^0$  associée à  $s$  (première définition), est sensible aux irrégularités de la surface. En choisissant  $\gamma^0$ , l'épaisseur serait celle de l'irrégularité, ce qui n'est pas souhaitable. Nous utilisons un critère moins contraignant pour choisir une sphère  $\gamma^1$  parmi les sphères maximales. On permet alors que la sphère  $\gamma^2$  passant par  $s$  et concentrique à  $\gamma^1$  déborde un peu de l'objet.

$[s\gamma_C]$  n'est pas contenu dans l'objet  $O_k$ ). Ceci est nécessaire, sur des objets non convexes. La fonction de coût définie précédemment, nous permet de sélectionner des sphères maximales qui soient :

- le plus près possible de  $s$ , pour éviter d'associer le point  $s$  de la surface à une sphère maximale lointaine ne correspondant pas à la forme autour de  $s$
- les plus grandes possible ; ce qui permet d'éviter les sphères correspondant à des perturbations locales de la surface.

Le coefficient  $\alpha$  règle le compromis entre les deux facteurs.

Une fois la sphère optimale minimisant  $\zeta$  trouvée, nous définissons l'épaisseur  $\xi(s)$  comme deux fois la distance de  $s$  au centre  $\gamma_C$  de la sphère optimale (figure 4.20) :

$$\xi(s) \triangleq 2d(s, \gamma_C(s)) \quad \text{avec} \quad \gamma(s) \triangleq \underset{\gamma \in \Gamma(O_k)}{\operatorname{argmin}} \left\{ \zeta(\gamma_R, d(s, \gamma_C)) \right\} \quad (4.18)$$

La figure 4.21 illustre cette définition de l'épaisseur pour une forme aléatoire en 2D.

**Trois cas particuliers** Nous allons démontrer que les deux premières définitions décrites précédemment sont des cas particuliers de la définition de l'épaisseur que nous proposons. Et que la troisième est étroitement liée.

Pour  $\alpha = 0$ , l'épaisseur en  $s$  devient la distance  $d(s, \Gamma(O_k))$  au squelette. C'est la deuxième définition décrite ci-dessus. En effet, Si  $\alpha = 0$  alors  $\zeta(\gamma_R, d(s, \gamma_C)) = d(s, \gamma_C)$ . La sphère maximale optimale associée à  $s$  est donc la sphère du squelette  $\Gamma(O_k)$  qui minimise  $d(s, \gamma_C)$ . On a donc  $d(s, \gamma_{C_{\text{optimal}}}) = d(s, \Gamma(O_k))$ .

Pour  $\alpha = 1$ , l'épaisseur sera le diamètre de la sphère maximale passant par  $s$ , C'est la première définition décrite ci-dessus. En effet, si  $\alpha = 1$  alors  $\zeta(\gamma_R, d(s, \gamma_C)) = d(s, \gamma_C) - \gamma_R$ . La sphère  $\gamma$  étant par définition incluse dans l'objet, on a donc  $\gamma_R \leq d(s, \gamma_C)$ . D'où  $\zeta \geq 0$ . Or  $\zeta = 0$  si on choisit la sphère maximale passant par  $s$ , qui réalise donc le minimum de  $\zeta$ .

Pour  $\alpha \rightarrow \infty$ , l'objet<sup>13</sup> entier sera réduit à la plus grande sphère qu'il contient. C'est la sphère dont le diamètre correspond à la troisième définition décrite ci-dessus. Cependant, dans

<sup>13</sup>On suppose ici que l'objet est convexe.

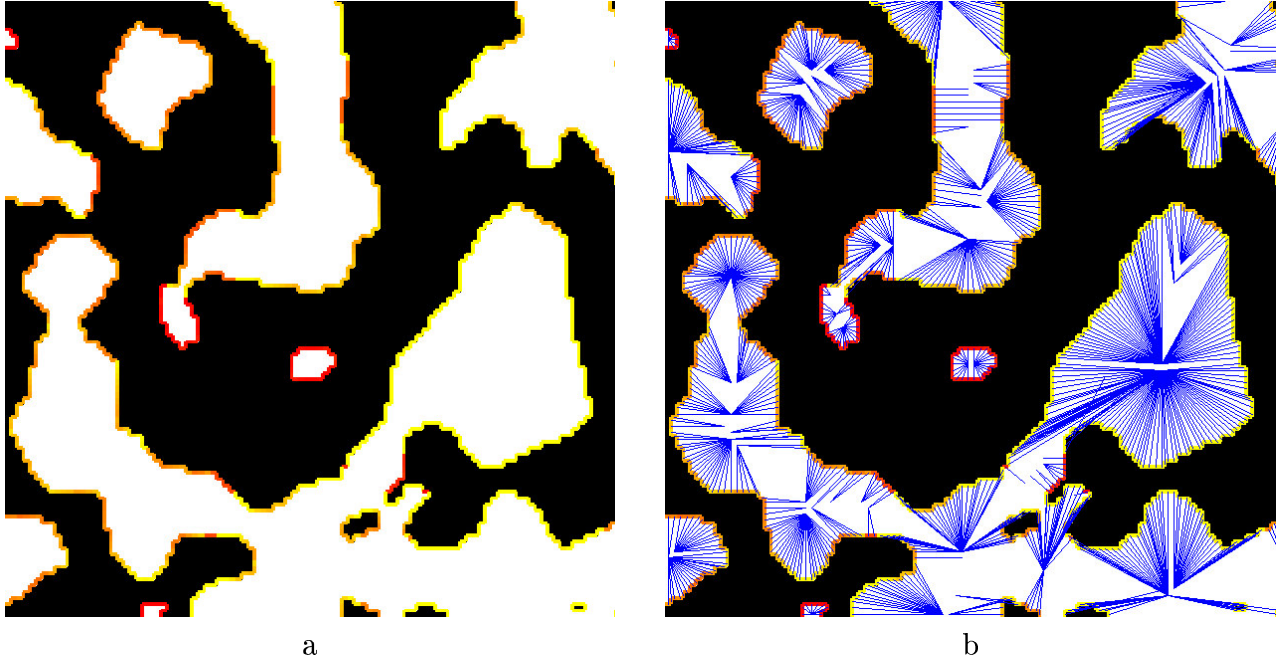


FIG. 4.21 – (a) Épaisseur d'un objet aléatoire en deux dimensions, avec  $\alpha = 2$ . En chaque élément de sa surface, l'épaisseur est représentée par sa couleur. (rouge : faible épaisseur, jaune : forte épaisseur). (b) Les traits bleus relient chaque élément de surface  $s$  à la sphère maximale optimale associée  $\gamma(s)$ .

notre définition, l'épaisseur est calculée à partir de la distance au centre de la sphère et non par le diamètre de la sphère. Les deux définitions utilisent la même sphère, mais donnent des résultats différents.

### Définition de l'énergie de la contrainte d'épaisseur

Maintenant que nous avons une définition de l'épaisseur, nous pouvons définir l'énergie associée à cette contrainte d'épaisseur de l'objet  $O_k$  :

$$E \triangleq \sum_{s \in S_k} -\log(P(\xi(s))) \quad (4.19)$$

où  $P(\xi(s))$  est la probabilité d'observer l'épaisseur  $\xi(s)$ , sur l'élément de surface  $s$ .  $P$  est une distribution de probabilité, éventuellement paramétrique, qui peut faire l'objet d'un apprentissage.

### Variation d'énergie

Pour calculer la variation de l'énergie qui vient d'être définie, nous devons déterminer la variation d'épaisseur qui a lieu lorsqu'un voxel bascule.

**Variation d'épaisseur** Le calcul de la variation exacte de l'épaisseur de l'objet  $O_k$  lorsqu'un voxel bascule, implique de recalculer la carte des distances intérieures de  $O_k$ . Ceci exigerait un temps de calcul rédhibitoire. Nous proposons ici un raisonnement permettant un calcul analytique de la variation d'épaisseur en un élément de surface  $s_1$  (quand il y a une variation

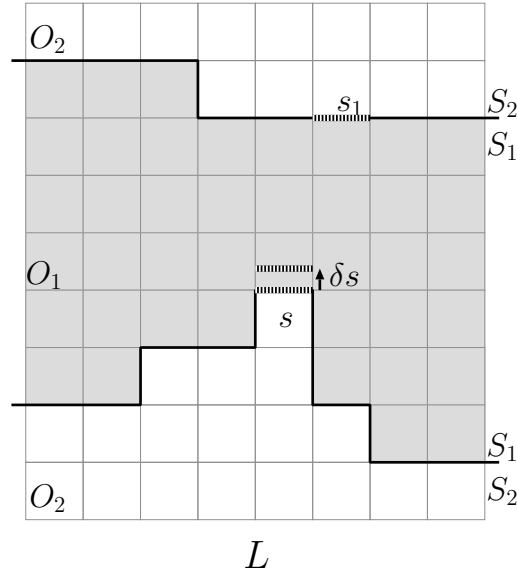


FIG. 4.22 – Calcul de la variation d'épaisseur en un élément de surface  $s_1$  de l'objet  $O_1$  : un déplacement infinitésimal  $\delta s$  de l'élément de surface  $s$  peut provoquer un changement d'épaisseur en un élément de surface distant  $s_1$ .

$\delta s$  en un élément de surface  $s$ , éventuellement distinct de  $s_1$ , voir figure 4.22) à l'aide de l'approximation suivante : nous allons supposer que nous sommes proches du cas continu, que les éléments de surface sont réduits à des points, et que les distances sont euclidiennes. La différenciation sera dénotée par le symbole  $\delta$  pour ne pas confondre avec la distance  $d$ . Calculons ainsi la variation d'épaisseur  $\delta\xi(s_1)$  en un élément de surface  $s_1$ , lorsque l'élément de surface  $s$  (réduit à un point) effectue un déplacement  $\delta s$ . Remarquons que  $s$  peut être soit distant de  $s_1$  (et alors  $\delta s_1 = 0$ ), soit égal à  $s_1$  (et alors  $\delta s_1 = \delta s$ ). Avec les approximations proposées, nous pouvons différentier<sup>14</sup> la définition de l'épaisseur (équation 4.18).

$$\delta\xi(s_1) = 2\vec{u}_{s_1} \cdot \delta\gamma_C(s_1) - 2\vec{u}_{s_1} \cdot \delta s_1 \quad (4.20)$$

où  $\vec{u}_{s_1}$  est le vecteur unitaire allant de  $s_1$  vers le centre de la sphère associée  $\gamma_C(s_1)$  :  $\vec{u}_{s_1} \triangleq \frac{\overrightarrow{s_1\gamma_C(s_1)}}{d(s_1, \gamma_C(s_1))}$ . L'étude, qui suit, des deux termes de cette équation fait apparaître quatre configurations (voir figure 4.23).

Le premier terme correspond au mouvement du centre de la sphère  $\gamma(s_1)$  associé à  $s_1$ . Si la sphère  $\gamma(s_1)$  ne passe pas par  $s$  (configurations b et c, figure 4.23) alors  $\delta\gamma_C(s_1) = 0$ , et ce deuxième terme est nul. Si la sphère  $\gamma(s_1)$  passe par  $s$  (configurations a et d, figure 4.23), alors nous approchons le déplacement de son centre par  $\delta\gamma_C(s_1) = \frac{1}{2}\delta s$ . Ce premier terme s'écrit alors :

$$2\vec{u}_{s_1} \cdot \delta\gamma_C(s_1) \simeq \vec{u}_{s_1} \cdot \delta s \begin{cases} 0 & \text{si } s \notin \gamma(s) \\ 1 & \text{si } s \in \gamma(s) \end{cases} \quad (4.21)$$

Le deuxième terme de l'équation 4.20 comporte deux cas. Soit  $s = s_1$  (le point sur lequel nous calculons l'épaisseur est le même que celui qui se déplace, configurations a et b, figure 4.23), auquel cas ce terme peut s'écrire  $\vec{u}_{s_1} \cdot \delta s$ . Soit  $s \neq s_1$  (configurations c et d, figure 4.23),

<sup>14</sup>L'expression de la différentielle d'une distance est donnée en annexe § C.2, p. 171.

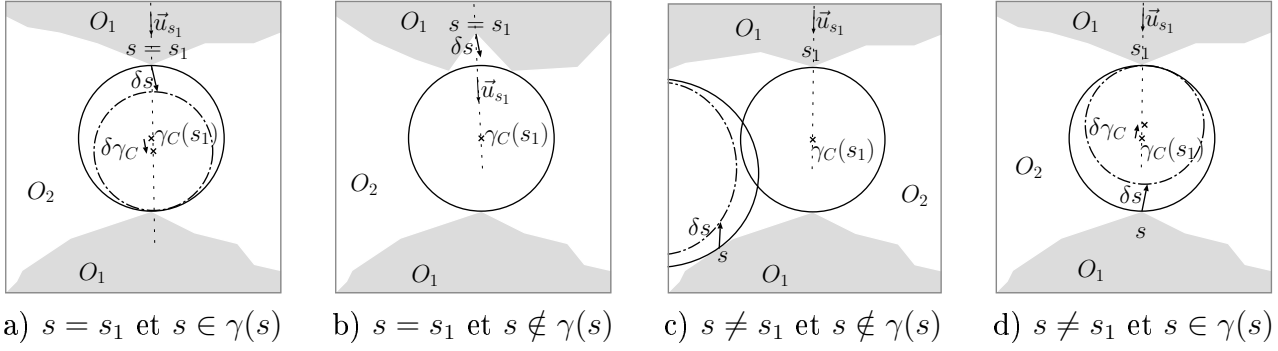


FIG. 4.23 – Variation d'épaisseur au point  $s_1$  due au mouvement  $\delta s$  de  $s$  dans l'approximation continue : quatre configurations sont possibles (voir texte).

et dans ce cas,  $\delta s_1 = 0$  et le terme est nul. Ce deuxième terme s'écrit alors :

$$-2\vec{u}_{s_1} \cdot \delta s_1 = \vec{u}_{s_1} \cdot \delta s \begin{cases} 0 & \text{si } s \neq s_1 \\ -2 & \text{si } s = s_1 \end{cases} \quad (4.22)$$

Pour résumer, un changement  $\delta s$  en  $s$ , produit la variation d'épaisseur  $\delta \xi(s_1)$  au point  $s_1$  :

$$\delta \xi(s_1) = \vec{u}_{s_1} \cdot \delta s \begin{cases} -1 & \text{si } s = s_1 \text{ et } s \in \gamma(s) \\ -2 & \text{si } s = s_1 \text{ et } s \notin \gamma(s) \\ 0 & \text{si } s \neq s_1 \text{ et } s \notin \gamma(s) \\ 1 & \text{si } s \neq s_1 \text{ et } s \in \gamma(s) \end{cases} \quad (4.23)$$

**Expression de la différentielle de l'énergie associé à un objet  $O_k$**  Calculons maintenant la variation d'énergie totale lors d'un déplacement  $\delta s$ . D'après la définition de l'énergie de cette contrainte (équation 4.19)

$$\begin{aligned} \delta E &= \sum_{s_1 \in S_k} -\log\left(\frac{P(\xi(s_1) + \delta \xi(s_1))}{P(\xi(s_1))}\right) \\ &= -\log\left(\frac{P(\xi(s) + \delta \xi(s))}{P(\xi(s))}\right) + \sum_{\substack{s_1 \in S_k \\ s_1 \neq s}} -\log\left(\frac{P(\xi(s_1) + \delta \xi(s_1))}{P(\xi(s_1))}\right) \end{aligned} \quad (4.24)$$

En tenant compte du cas  $\delta \xi(s_1) = 0$  (équation 4.23) :

$$\begin{aligned} \delta E &= -\log\left(\frac{P(\xi(s) + \delta \xi(s))}{P(\xi(s))}\right) + \sum_{\substack{s_1 \in S_k \\ s_1 \neq s \\ s \in \gamma(s_1)}} -\log\left(\frac{P(\xi(s_1) + \delta \xi(s_1))}{P(\xi(s_1))}\right) \\ &= -\log\left(\frac{P(\xi(s) + a\vec{u}_{s_1} \cdot \delta s)}{P(\xi(s))}\right) + \sum_{\substack{s_1 \in S_k \\ s_1 \neq s \\ s \in \gamma(s_1)}} -\log\left(\frac{P(\xi(s_1) + \vec{u}_{s_1} \cdot \delta s)}{P(\xi(s_1))}\right) \end{aligned} \quad (4.25)$$

$$\text{avec } a \triangleq \begin{cases} -1 & \text{si } s \in \gamma(s) \\ -2 & \text{si } s \notin \gamma(s) \end{cases}$$

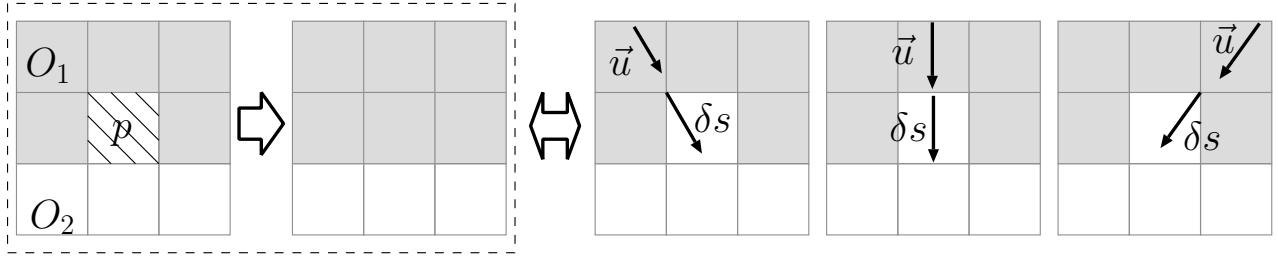


FIG. 4.24 – *Passage de l'approximation continue au cas discret. Le basculement d'un voxel au point  $p$  peut être approché par un déplacement unité dans toutes les directions  $\vec{u}$  où il pourrait y avoir des sphères maximales associées. Ici un voxel est « enlevé » à l'objet  $O_2$ .*

Le terme de gauche est un terme local, il représente la variation d'énergie en  $s$  dûe au mouvement de  $s$ . Le terme de droite est un terme non local. Le mouvement de  $s$  affecte une sphère qui définit l'épaisseur sur un autre point  $s_1$ .

**Passage de l'approximation continue au cas discret** Nous avons jusque là envisagé un déplacement infinitésimal  $\delta s$  d'un élément de surface  $s$ . Nous devons maintenant déterminer la variation d'énergie lorsque le label en un point  $p$  de  $L$  bascule et prend la valeur  $l$ . Nous supposons alors que le basculement d'un voxel au point  $p$  peut être approché par un déplacement unité dans toutes les directions  $\vec{u}$  où il pourrait y avoir des sphères (voir figure 4.24). Nous obtenons ainsi la variation d'énergie associée à un des éléments de surface voisins de  $p$ . Or, le basculement de  $p$  concerne plusieurs éléments de surface voisins de  $p$ . Nous considérons alors tous les éléments de surface de  $S_k$  voisins de  $p$  et retenons la variation maximale d'énergie. La variation d'énergie due au basculement au point  $p$  s'écrit donc :

$$\Delta E_l(p) = \Delta E_l(p, s_{max}) \quad \text{avec} \quad s_{max} \triangleq \underset{s \text{ voisin de } p}{\operatorname{argmax}} |\Delta E_l(p, s)| \quad (4.26)$$

où la variation  $\Delta E_l(p, s)$  associé à un élément de surface  $s$  particulier voisin de  $p$  est :

$$\Delta E_l(p, s) \triangleq -\log\left(\frac{P(\xi(s) + a\vec{u}_{s_1}.\delta s)}{P(\xi(s))}\right) + \sum_{\substack{s_1 \in S_k \\ s_1 \neq s \\ s \in \gamma(s_1)}} -\log\left(\frac{P(\xi(s_1) + \vec{u}_{s_1}.\delta s)}{P(\xi(s_1))}\right) \quad (4.27)$$

$$\text{avec } a \triangleq \begin{cases} -1 & \text{si } s \in \gamma(s) \\ -2 & \text{si } s \notin \gamma(s) \end{cases}$$

La variation d'énergie décrite est donc composée de deux termes (équation 4.27) : le premier est un terme local, dans lequel la variation d'énergie en un point est due à son propre mouvement. Et le deuxième est un terme non local dans lequel la variation d'énergie est due au mouvement d'un point de surface distant. (voir § 4.5, p. 69, et figure 4.25)

**Cas des petites structures** Les approximations dans le passage entre le continu et le discret ne sont plus valables pour des petites structures dont l'épaisseur est de l'ordre du voxel. Les petites structures représentent un double intérêt. Elles sont fréquentes dans les images, et elles apparaissent spontanément lors de l'évolution des objets. Il est donc important de les traiter ce cas avec précision. Pour ces petites structures, il est possible de précalculer l'épaisseur de toutes les configurations de labels possibles dans un voisinage restreint (voir figure 4.26).



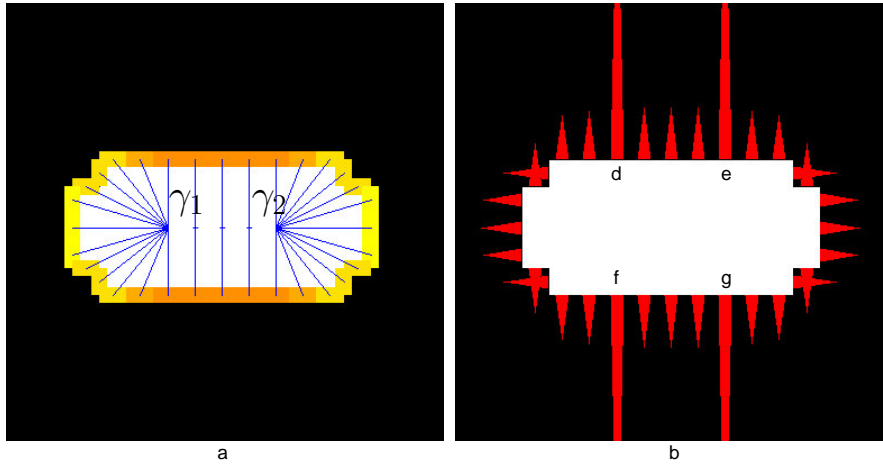


FIG. 4.25 – (a) L'épaisseur d'un objet simple en chaque élément de sa surface est représentée par sa couleur. (rouge : faible épaisseur, jaune : forte épaisseur). Les traits bleus relient chaque élément de surface  $s$  à la sphère maximale optimale associée  $\gamma(s)$ . (b) Variations d'énergie en chaque élément de la surface du même objet. L'épaisseur souhaitée est plus grande que celle présente en chaque point de cet objet, et donc l'objet a tendance à grandir. Remarquez les variations d'énergie plus fortes aux quatre points  $d, e, f, g$ . Ces variations plus fortes sont dues au terme non local. En effet, un changement en ces points provoque une variation des sphères  $\gamma_1$  ou  $\gamma_2$ , entraînant une variation d'épaisseur sur un grand nombre de points.

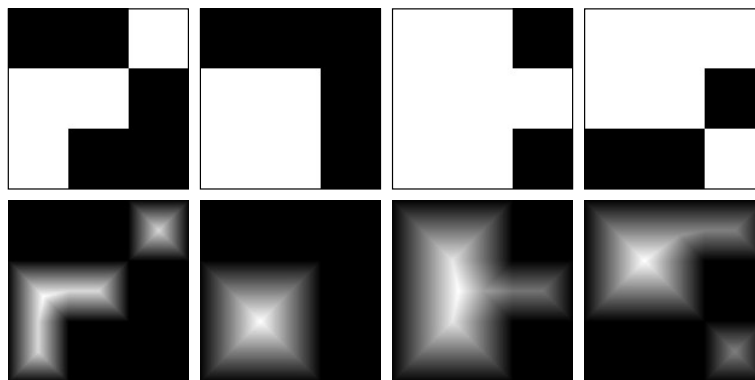


FIG. 4.26 – Pour des petites structures (faible épaisseur), il est possible de précalculer l'épaisseur de toutes les configurations possibles dans un voisinage restreint. La première rangée montre quatre configurations parmi les  $2^{3 \times 3}$  possibles dans le cas d'un voisinage  $3 \times 3$  en 2D. La deuxième rangée montre les cartes de distances associées. L'épaisseur d'une petite structure sera définie comme deux fois le maximum de la carte de distance associée.

### Exemples d'évolution

Trois exemples simples illustrant le fonctionnement de cette contrainte sont montrés et commentés à la figure 4.27.

### Paramétrage par un atlas

Le paramétrage de la contrainte d'épaisseur est effectué au travers de la probabilité  $P(\xi(s))$ . Une distribution gaussienne avec des moyennes et écarts types appris et associés à un atlas peut être envisagée pour modéliser, par exemple, les structures internes du cerveau.

## 4.6 Pré-traitements et initialisation

Nous abordons ici la détermination de la segmentation initiale pour le processus itératif de segmentation, et la détermination des intensités associées aux labels.

### 4.6.1 Estimation de l'intensité des labels

Le modèle d'acquisition (§ 4.4.2, p. 63) décrit la relation entre l'image de labels  $L$  et l'image à segmenter  $I_0$ . Son utilisation nécessite d'associer une intensité  $V(l)$  à un label  $l$ . Nous avons adopté une approche spécifique à notre problème, robuste et utilisant l'ensemble de l'information disponible dans l'image.

Tout d'abord, les intensités  $V_0(l)$ ,  $l \in \{1 \dots n\}$  des différents labels sont déterminées une fois pour toutes sur une image de référence<sup>15</sup>. On établit ensuite la correspondance entre ces intensités et celles des images sources à segmenter, par la procédure suivante : pour chaque image source  $I_0$  qu'on souhaite segmenter, l'image de référence est recalée sur l'image source en utilisant un recalage déformable. On détermine ensuite l'histogramme joint entre l'image de référence recalée et l'image à segmenter  $I_0$ . Une fonction de transfert d'intensité  $f_T$  est alors calculée à partir de l'histogramme joint selon la procédure décrite au paragraphe 7.4.1. On obtient finalement, à l'aide de  $f_T$ , la relation entre l'intensité des labels dans l'image source  $I_0$  et l'intensité des labels dans l'image de référence de l'atlas  $I_0 : V(l) = f_T(V_0(l))$  (voir figure 4.28).

### 4.6.2 Initialisation

Dans notre approche itérative, nous devons fournir une segmentation initiale  $L_0$ . Nous avons constaté, comme le décrit la figure 4.29, que le résultat final de la segmentation ne dépend pratiquement pas de l'image initiale. Notre système est capable de converger même si la segmentation initiale est très loin de la solution finale. Comme l'illustre la figure 4.29, des changements de topologies très complexes sont possibles. Une mauvaise initialisation demandera alors des itérations supplémentaires pour converger (en pratique, une quinzaine sur un total de 150 itérations, voir chapitre suivant). Lors de la segmentation du cerveau en trois classes, un simple seuillage utilisant les intensités calculées au paragraphe précédent est suffisant. Une initialisation plus fine pourrait être nécessaire dans des applications où certaines

<sup>15</sup>Diverses techniques sont envisageables pour déterminer l'intensité des différents labels sur une image de référence. Cette procédure n'étant effectuée qu'une seule fois, on utilise une approche interactive à l'aide d'outils présents dans un visualiseur (histogrammes de régions d'intérêt, voir § B.3, p. 167).

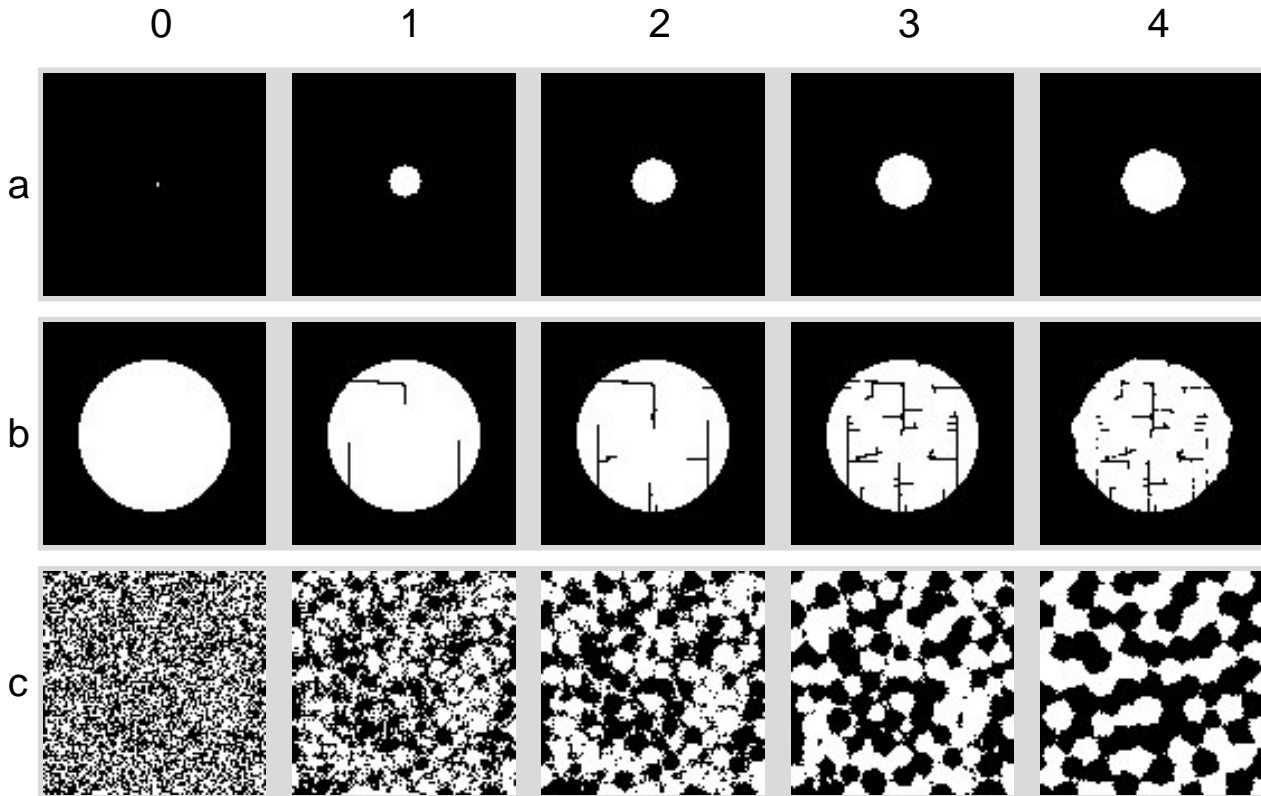


FIG. 4.27 – Trois exemples (a, b, c) d'évolution (0-1-2-3-4) sous le seul effet de la contrainte d'épaisseur. La segmentation initiale est représentée à la colonne 0. La **première** évolution (a) part d'une image réduite à un point. La distribution de probabilité d'épaisseur est une gaussienne de valeur moyenne 30 pixels. Le point grandit uniformément pour définir une sphère, démontrant ainsi les propriétés d'évolution de la contrainte. Si cet exemple peut sembler trivial, il est important de remarquer que l'obtention d'un épaissement uniforme est loin d'être évident. Il exige qu'à chaque itération, le système permette de choisir précisément le pixel provoquant un épaissement dans la bonne direction ; en particulier lorsque l'objet n'est composé que de quelques pixels, cela démontre la robustesse du traitement des petites structures. Les approximations faites dans le calcul de la variation de l'épaisseur sont ainsi validées. La **deuxième** évolution (b) part d'une grande sphère, utilisant une contrainte d'épaisseur gaussienne de 15 pixels. Comme le montre cette évolution, contrairement à ce qui pourrait être attendu intuitivement, l'épaisseur d'un objet est diminué le plus efficacement en le scindant. La **dernière** évolution (c) part d'une image aléatoire. L'évolution se fait sous l'effet de deux contraintes d'épaisseur 15. L'une s'applique au label blanc, l'autre au label noir. L'évolution aboutit à des régions homogènes ayant les épaisseurs souhaitées.

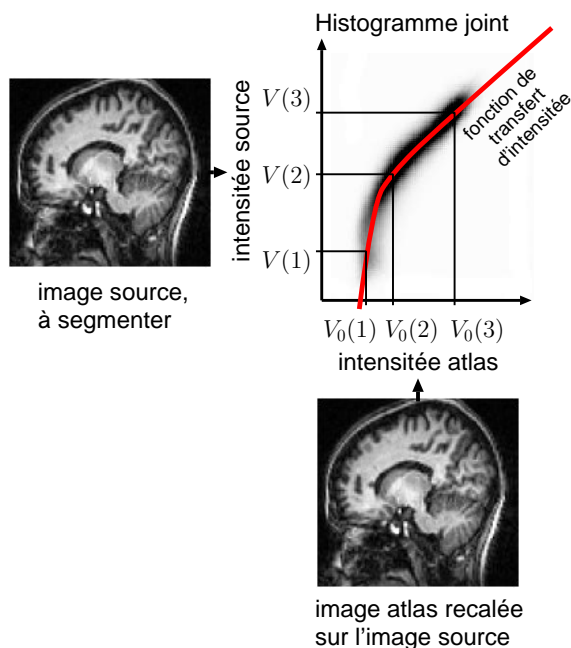


FIG. 4.28 – L'intensité,  $V(1)$ ,  $V(2)$ ,  $V(3)$  de chaque label dans l'image à segmenter est déduite de l'intensité  $V_0(1)$ ,  $V_0(2)$ ,  $V_0(3)$  du label dans l'image de référence de l'atlas. La correspondance est effectuée en calculant une fonction de transfert d'intensité à l'aide de l'histogramme joint.

classes n'ont pas de niveau d'intensité bien défini. Dans ces cas, une initialisation à partir de l'image d'atlas de référence recalée pourrait être envisagée.

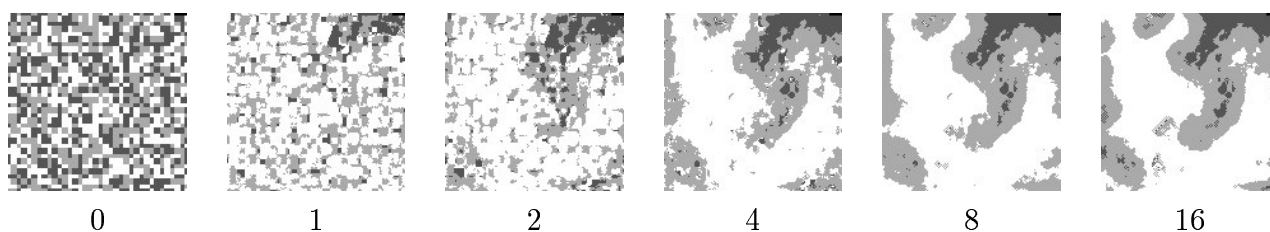


FIG. 4.29 – Premières itérations d'une segmentation d'une portion 3D de cerveau à partir d'une initialisation aléatoire. La convergence -non montrée ici- est atteinte après 150 itérations environ. Après une dizaine d'itérations, la segmentation ne porte plus la trace de son initialisation. Le résultat final ne diffère que de 2% avec une initialisation distincte. Cet exemple de segmentation d'une image réelle est obtenu avec l'ensemble des contraintes du modèle des sillons corticaux décrit au chapitre suivant.



# Chapitre 5

## Résultats, perspectives, et conclusion

Le fonctionnement du système de segmentation a été décrit au chapitre précédent, nous allons maintenant l'appliquer à la segmentation des sillons corticaux. Nous évoquerons les limites de ce système et proposerons des extensions, avant de conclure cette partie consacrée à la segmentation.

### 5.1 Segmentation des sillons corticaux

Notre système de segmentation est général, permettant de segmenter des types d'objets très variés, comme l'attestent les exemples d'objets bidimensionnels décrits lors de la présentation de chaque contrainte. Son instanciation pour une application particulière, comme la segmentation des sillons corticaux, nécessite de construire une description des objets à segmenter. Après avoir présenté cette description (§ 5.1.1) et les outils nécessaires, nous évaluerons sa performance (§ 5.1.2), et évoquerons les points pratiques de son implémentation.

#### 5.1.1 Construction du modèle des sillons corticaux

La description des sillons corticaux est construite en choisissant les différentes contraintes que l'on souhaite utiliser, puis en déterminant les paramètres de ces contraintes. Ces paramètres peuvent éventuellement dépendre de la position, et faire l'objet d'un apprentissage statistique (§ 5.2.1, p. 100).

Le cerveau a été modélisé ici, à l'aide de deux contraintes images (§ 4.4.3, p. 65). La première contrainte image opère à une échelle très fine, sans lissage. La deuxième utilise un filtre de Butterworth d'ordre 1 et de fréquence de coupure (normalisée) 0,12 pour le lissage. Nous rappelons que cette association de plusieurs contraintes images permet d'obtenir à la fois une convergence à petite échelle et une convergence globale, à grande échelle.

Deux contraintes complémentaires de distribution relative sont employées (voir figure 5.1). Les distributions de probabilités associées sont les « sigmoïdes » décrites à l'équation 4.16 :  $P(l, d) = \frac{1}{2} \operatorname{erfc}\left(\pm \frac{d-d_0}{\sqrt{2}\sigma}\right)$ . Le paramètre  $d_0$  correspond à l'épaisseur de la matière grise, et  $\sigma$  à la variance de cette épaisseur. Les paramètres  $d_0 = 1.7$  mm et  $\sigma = 0,5$  mm sont choisis fixes (à partir des épaisseurs publiées dans la littérature [68]), ne dépendant donc pas d'un atlas<sup>1</sup>. La première contrainte décrit la distribution de matière grise et de matière blanche autour

---

<sup>1</sup>L'épaisseur de MG autour du LCR est en réalité légèrement supérieure autour de la faux centrale [68], il pourrait donc être intéressant de faire varier ce paramètre.

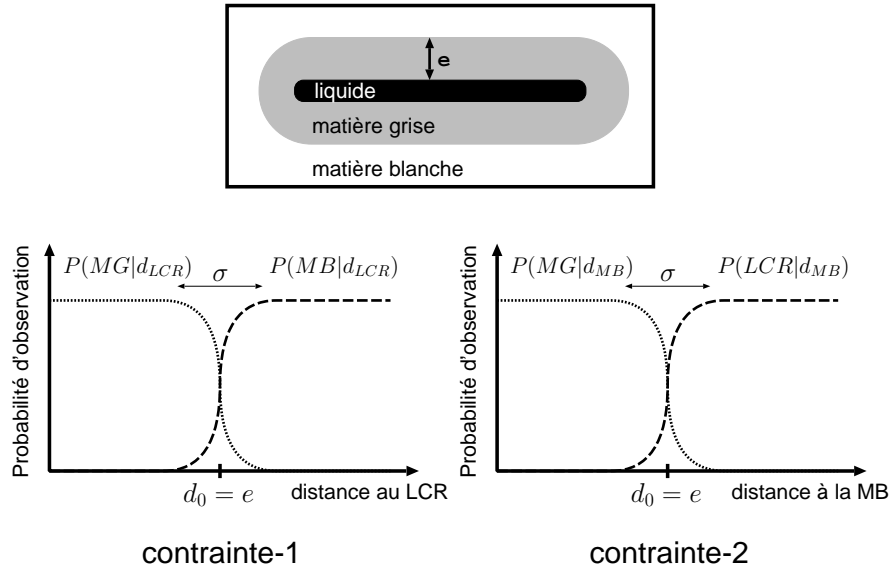


FIG. 5.1 – Deux contraintes de distribution relative complémentaires sont utilisées dans la modélisation des sillons.

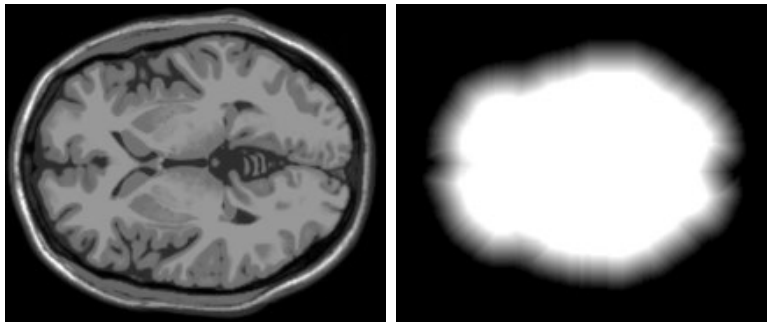


FIG. 5.2 – Le poids de la contrainte d'épaisseur diminue près de la surface extérieure du cerveau. À droite, le coefficient de cette contrainte et à gauche, l'image de référence de l'atlas associé (gauche).

du liquide. Et la deuxième décrit la distribution de matière grise et de liquide autour de la matière blanche<sup>2</sup>.

Nous souhaitons utiliser la contrainte d'épaisseur pour limiter l'épaisseur du liquide dans les sillons. Dans son implémentation actuelle, la contrainte d'épaisseur (pour une densité de probabilité générale) décrite au chapitre précédent est trop lente pour être utilisée sur des segmentations 3D en haute résolution. Les premiers essais montrent que ce problème devrait bientôt être résolu<sup>3</sup>. Une version simplifiée de cette contrainte, où la densité de probabilité est une fonction en marche d'escalier ( $P(\xi) = 1$  pour  $\xi < \xi_0$  et  $P(\xi) = 0$  pour  $\xi \geq \xi_0$ ) est utilisée. L'épaisseur maximale du liquide étant moins prévisible vers l'extérieur du cerveau, on diminue le coefficient de cette contrainte près de sa surface extérieure (figure 5.2).

<sup>2</sup>Une contrainte unique de distribution (par exemple autour du LCR) pourrait être suffisante. Cependant, deux contraintes symétriques permettent une meilleure convergence.

<sup>3</sup>Le problème de la recherche de la sphère optimale minimisant la fonction coût  $\zeta$  (voir § 4.5.2, p. 77) a été résolu par une stratégie de subdivision récursive de l'espace.

**L'interface graphique** Pour l'utilisateur final, la segmentation est entièrement automatique. Cependant, la mise en place du modèle décrit précédemment, nécessite une phase d'expérimentation et donc des outils interactifs de visualisation que nous décrivons ici brièvement.

L'interface graphique (figure 5.3) est une extension dynamique (plugin) du visualiseur de ImLib3D (voir § B.5.2, p. 169). Elle permet de visualiser les éléments suivants nécessaires pour comprendre l'évolution de la segmentation :

- la segmentation courante et son évolution ;
- les variations d'énergie associées à chaque contrainte ;
- des données associées à des contraintes, comme l'épaisseur en un point, ou image d'erreur  $(I - I_0) * h$  d'une contrainte image.

L'interface graphique joue aussi un rôle important lors du développement des contraintes en permettant d'interagir avec le système de segmentation. Lors de l'expérimentation, l'utilisateur peut choisir de ne travailler que sur une portion restreinte de l'image, ce qui lui permet d'obtenir plus rapidement les résultats.

### 5.1.2 Validation expérimentale

La méthode présentée ici a été utilisée sur un grand nombre d'images de différentes résolutions. En particulier nous avons segmenté 75 images GE3D provenant de notre base d'images de sclérose en plaques (§ 8.1, p. 134). Nous avons pu vérifier qualitativement (visuellement) la pertinence des résultats (voir exemples aux figures 5.4, 5.5, 5.6, 5.7). La validation quantitative d'une approche de segmentation sous-voxel n'est pas simple. Il n'existe pas de vérité terrain accessible. Nous avons choisi d'utiliser le fantôme numérique BrainWeb [10, 15] qui est largement employé dans ce but dans la littérature. Malgré ses qualités indéniables, il présente tout de même quelques imperfections. Sa résolution est limitée à  $1 \text{ mm}^3$ , et malgré l'utilisation de degrés flous d'appartenance pour chaque classe (LCR, MG, MB), certaines régions fines de LCR des sillons sont manquantes.

Pour quantifier la qualité de la segmentation, nous utilisons le coefficient kappa  $\kappa$  décrivant le degré d'accord entre une segmentation et une référence [11], avec une correction pour les effets du hasard :

$$\kappa \triangleq \frac{p_0 - p_c}{1 - p_c} \quad (5.1)$$

où  $p_0$  est le degré d'accord (pourcentage de voxels correctement classifiés) et  $p_c$  est la proportion de voxels qui seraient correctement classifiés si la classification était due au hasard. On a donc  $\kappa = 0$  pour une classification aléatoire, et  $\kappa = 1$  pour une classification parfaite.

Remarquons que ce coefficient ne donne qu'une vision très étroite de la qualité de la segmentation. Il ne mesure que l'accord voxel par voxel avec la segmentation de référence. Il ne mesure pas la cohérence des structures anatomiques ni les propriétés des surfaces de séparation entre labels.

Le fantôme BrainWeb permet de choisir la quantité de bruit et l'intensité de l'artefact RF. Nous avons choisi l'artefact RF à 20% (valeur proposée par défaut), et un niveau de bruit compris entre 1% et 9%. Notre modèle ne décrivant que les sillons corticaux et non les structures internes (ventricules, noyaux, etc.), nous avons calculé les coefficients sur toute la portion supérieure du cerveau située au-dessus des ventricules. Cette limitation rend malheureusement difficile la comparaison précise avec d'autres résultats publiés dans la littérature. Nous proposons néanmoins, à la figure 5.8, une comparaison avec les résultats publiés par Ruan *et al.* [95]. Il s'agit d'une approche de classification du cerveau entier à l'aide d'un  $a$



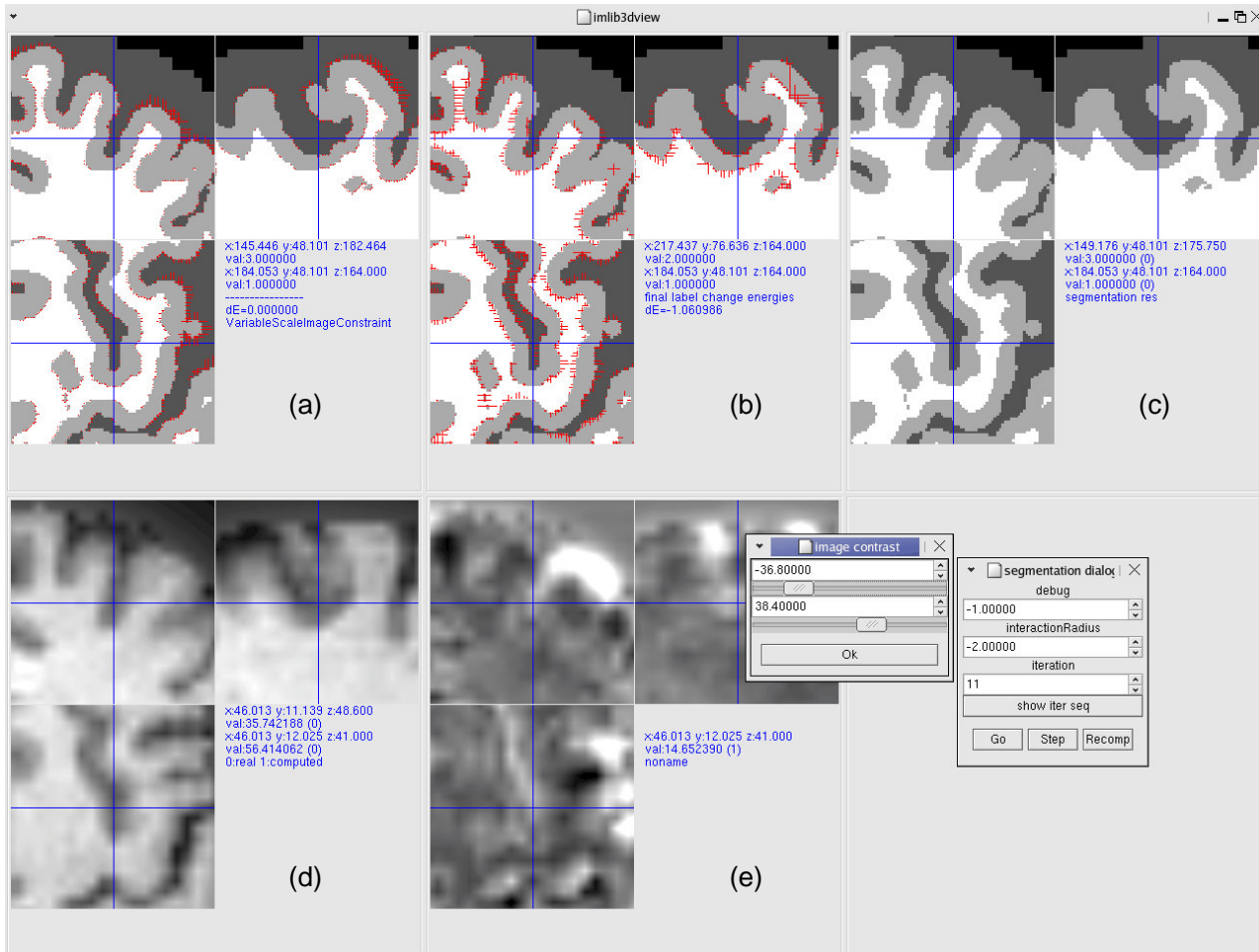


FIG. 5.3 – La mise en place du modèle des sillons corticaux, ainsi que l'expérimentation est facilitée par un système de visualisation interactif. Chaque affichage (a, b, c, d, e) dans le visualiseur montre trois coupes d'une ou plusieurs images volumiques. En (a) sont montrées les variations d'énergie (flèches rouges) associées à chaque contrainte  $\Delta E_i$ . L'utilisateur peut interactivement choisir la contrainte affichée. En (b) est montrée la variation d'énergie totale  $\Delta E$  (somme pondérée des  $\Delta E_i$ ). En (c) défile une animation entre la segmentation courante  $L$  et la nouvelle segmentation calculée à la fin de cette itération. En (d) est affichée une animation entre l'image à segmenter  $I_0$  et l'image simulée  $I$ . En (e) sont montrées les contraintes images aux différentes échelles. L'utilisateur peut interactivement choisir l'échelle affichée.

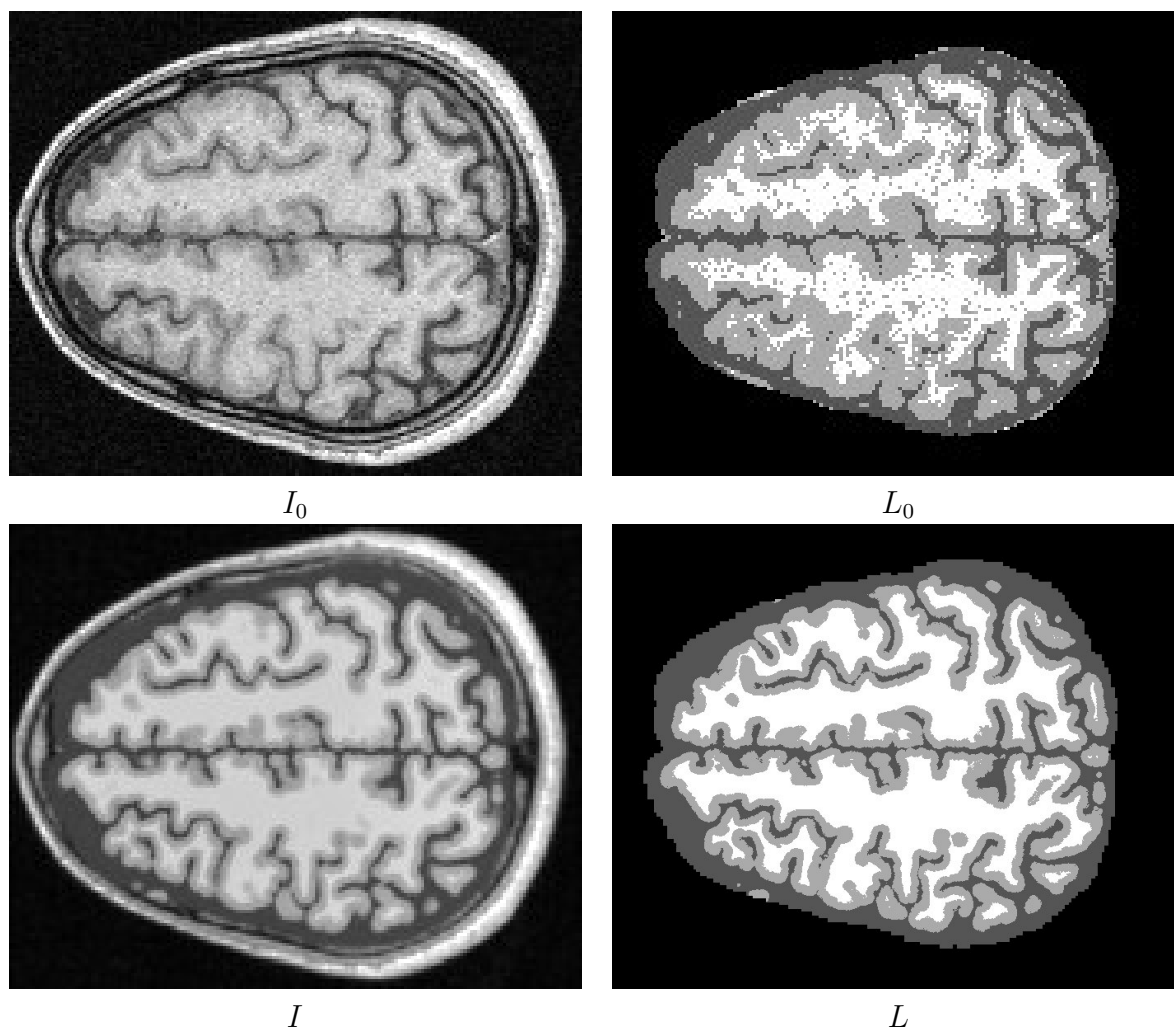


FIG. 5.4 – Exemple de segmentation d'un cerveau. L'image source  $I_0$  a une résolution de  $1 \text{ mm}^3$ . La segmentation, effectuée à une résolution 4 fois supérieure, évolue à partir de  $L_0$  pour atteindre  $L$ . L'image  $I$  est estimée à partir de  $L$  à l'aide du modèle d'acquisition.

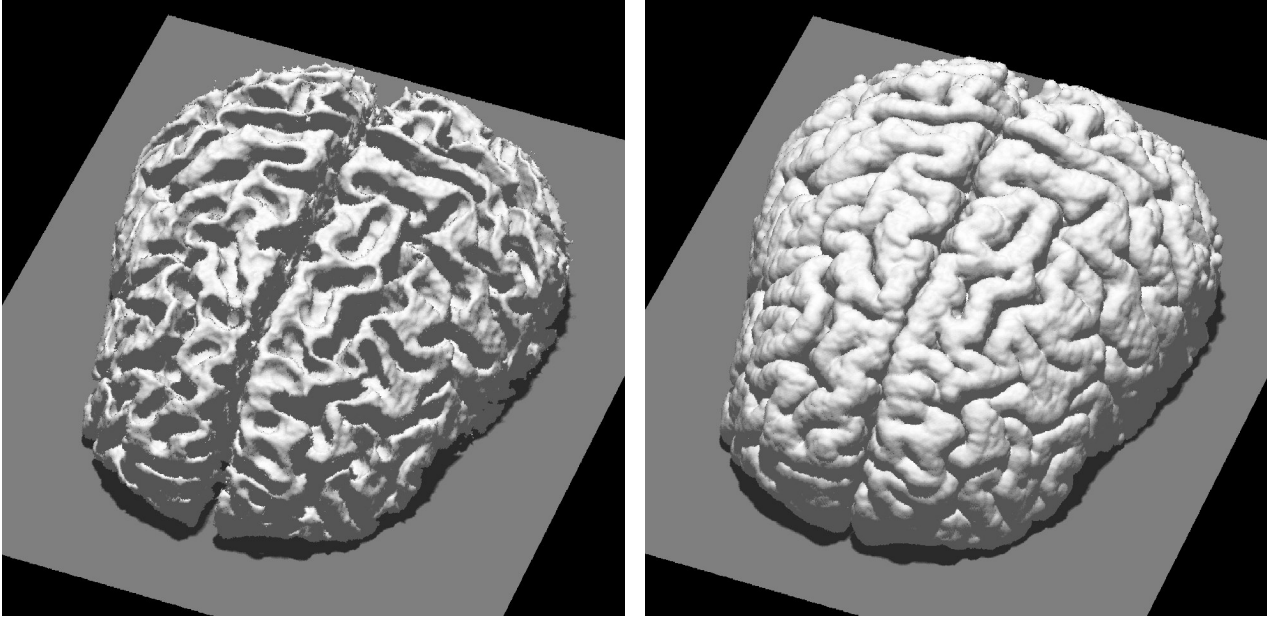


FIG. 5.5 – Rendu 3D de la MB (gauche) et de la MG (droite) segmentée à l'exemple de la figure 5.4. Quelques résidus dus à des erreurs dans la pré-segmentation de l'enveloppe externe du cerveau ont été enlevés manuellement.

*priori* markovien ainsi que d'informations sur la dimension multifractale. Les volumes partiels sont modélisés, dans leur approche, à l'aide de « mixels » (voir aussi § 3.1.3, p. 47).

### 5.1.3 Mise en œuvre pratique de la segmentation

Nous allons maintenant aborder certains aspects pratiques liés à l'implémentation effective de la méthode de segmentation.

#### 5.1.4 Convergence et rayon d'interaction

Comme cela a été présenté au paragraphe 4.2.2, p. 60, le choix du rayon d'interaction permet de réguler le nombre de voxels pouvant basculer à chaque itération et par conséquent, réguler la vitesse d'évolution de la segmentation. Si, dans la majorité des cas, la convergence est atteinte en peu d'itérations (une dizaine d'itérations, voir figure 5.9), dans d'autres cas, en particulier lorsque des contraintes contradictoires sont très fortes, la segmentation peut nécessiter plus d'itérations avant de converger. Pour conserver le caractère automatique de la segmentation, nous devons envisager le pire cas, et prévoir 150 itérations<sup>4</sup>. Le rayon d'interaction est choisi initialement nul, et est incrémenté de 1 voxel toutes les 25 itérations. De cette manière, la segmentation évolue rapidement dans un premier temps, de manière grossière, puis, progressivement, l'évolution se fait de façon plus lente mais plus précise. La nécessité d'augmenter le rayon d'interaction devient apparente lorsqu'apparaissent des oscillations des frontières de la segmentation autour d'une position d'équilibre.

<sup>4</sup>La convergence totale, prévue en théorie, n'est atteinte que sur des images de petite taille (exemples 2D montrées au chapitre précédent). Pour de grandes images avec des contraintes complexes, il faudrait utiliser un rayon d'interaction important pour éviter des petites oscillations résiduelles. Un critère d'arrêt automatique pourrait être implémenté, mais ce n'est pas encore le cas actuellement.

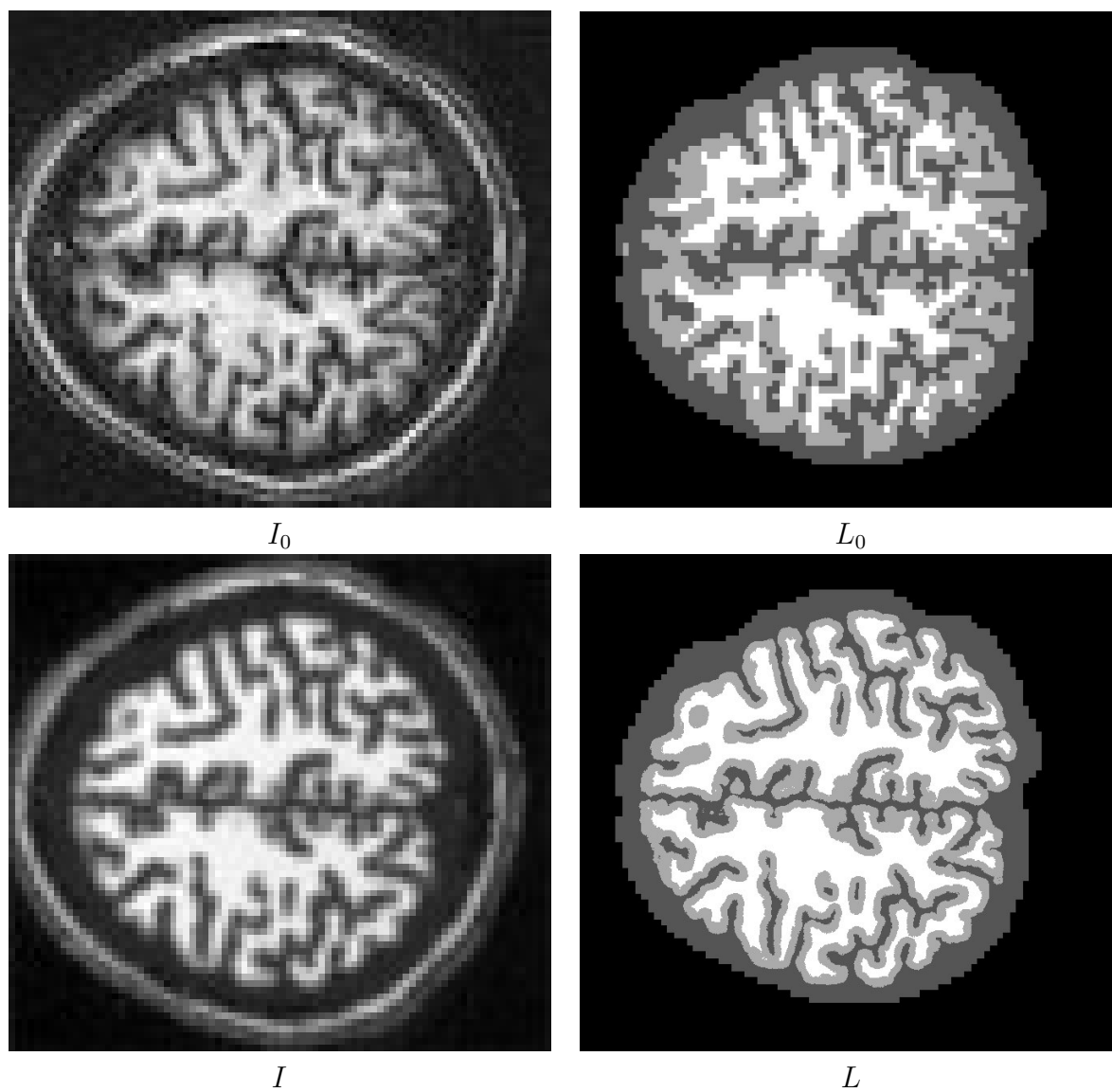


FIG. 5.6 – Exemple de segmentation à  $.25\text{ mm}^3$  d'un cerveau dont l'image source a une résolution de  $2\text{ mm}^3$ .

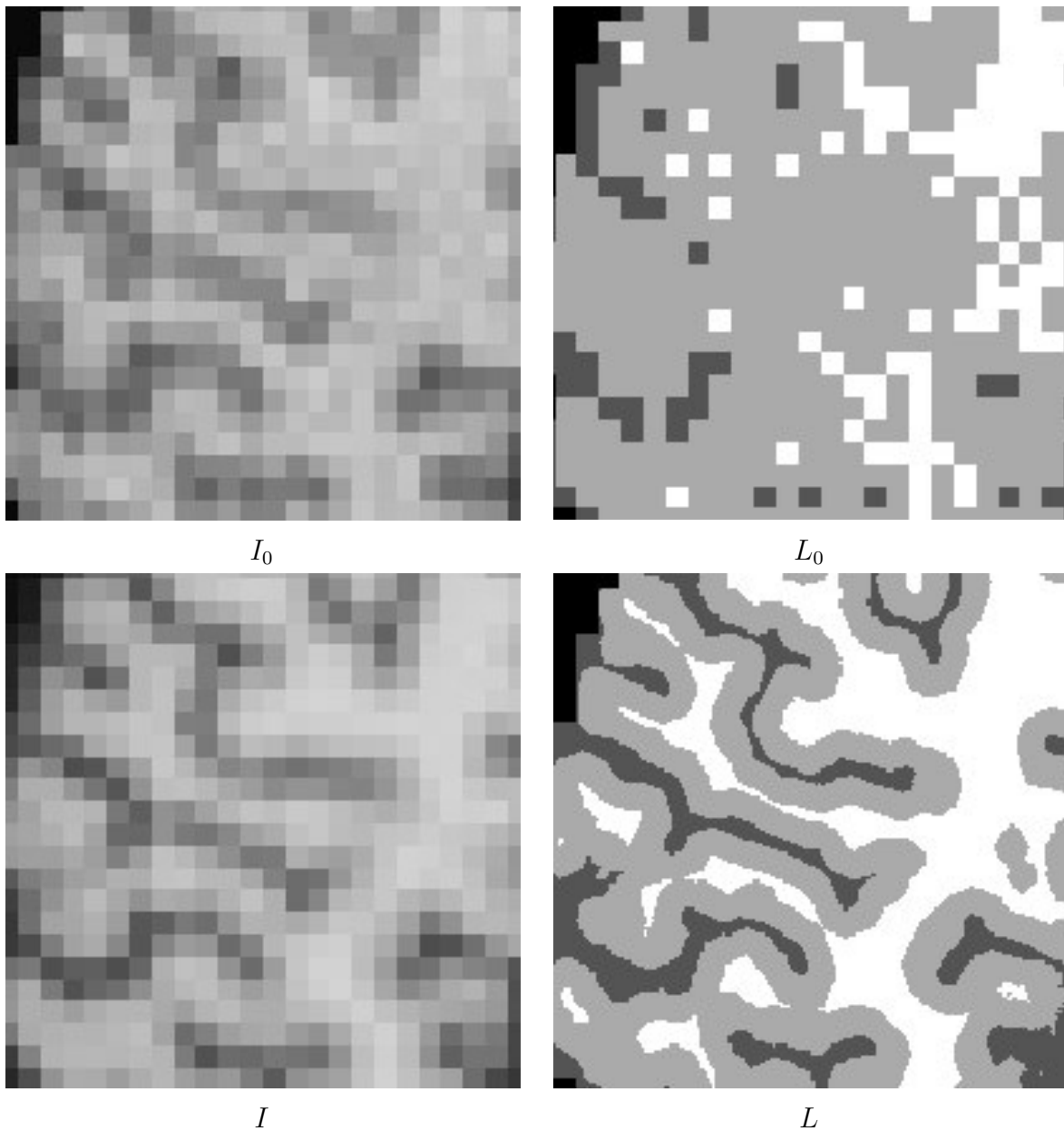


FIG. 5.7 – Exemple de segmentation d'un cerveau dont l'image source a une résolution  $2 \text{ mm}^3$ . L'image source  $I_0$  et l'image estimée  $I$  sont agrandies sans interpolation pour mettre en évidence la taille des voxels. La segmentation converge correctement, malgré une initialisation  $L_0$  éloignée de l'objectif.

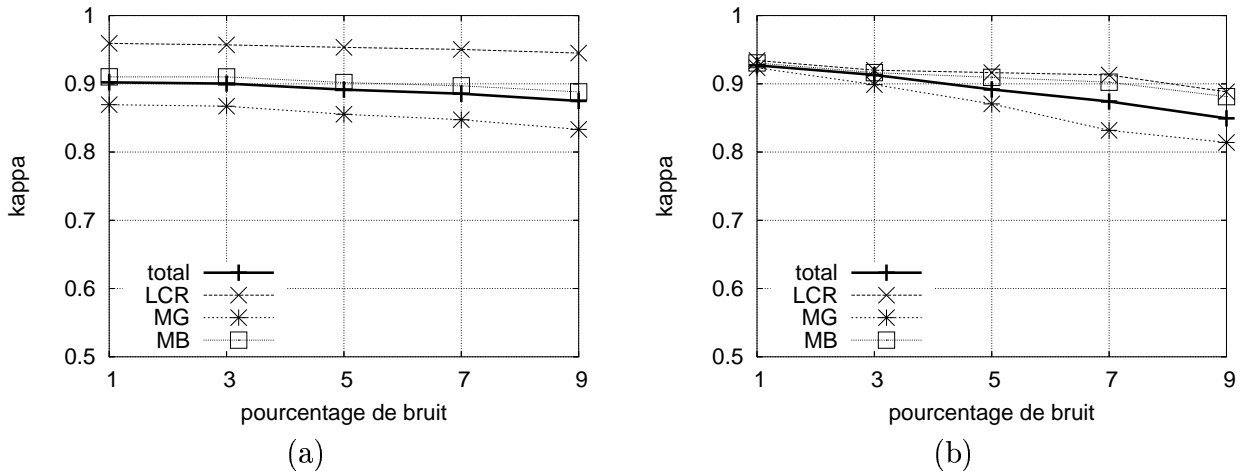


FIG. 5.8 – Comparaison des performances de (a) notre système de segmentation à celles du (b) système proposé par Ruan et al. [95]. L'évaluation est effectuée à l'aide du fantôme numérique BrainWeb. Le coefficient  $\kappa$  donne le degré d'accord entre la segmentation obtenue et la segmentation de référence fournie par BrainWeb. Les performances (total) des deux systèmes sont semblables. L'utilisation, dans notre approche, de connaissances a priori fortes, permet de mieux résister aux forts niveaux de bruit. Cependant, elles introduisent un biais préjudiciable lorsque les images sont de meilleure qualité. Le réglage du poids relatif (coefficients  $\alpha$ ) entre la contrainte image et le modèle a priori permettrait de s'adapter à la qualité de l'image. Lors de l'évaluation de notre approche, ces coefficients ont été maintenus constants.

### 5.1.5 Temps de calcul, encombrement mémoire et structures de données

Le système de segmentation que nous avons présenté résout le problème dit des « volumes partiels », en segmentant à une résolution supérieure à celle de l'image acquise (§ 4.4.2, p. 63). Le temps de calcul et l'occupation mémoire dépendent alors principalement de la résolution à laquelle se déroule la segmentation (image de labels  $L$ ) et non de la résolution de l'image à segmenter ( $I_0$ ).

La portion du cerveau qu'on cherche à segmenter a typiquement une taille réelle d'environ  $146 \times 88 \times 170 \text{ mm} \simeq 2,2 \cdot 10^6 \text{ mm}^3$ . La taille de la segmentation  $L$ , pour une segmentation à une résolution voxel de  $= 0,5 \times 0,5 \times 0,5 \text{ mm}$ , est donc d'environ  $1,7 \cdot 10^7$  voxels. Cette résolution  $R_{0,5}$  est suffisante pour segmenter les sillons corticaux. Pour mieux représenter les détails, en particulier dans les régions centrales, très fines, des sillons, il peut être intéressant de segmenter à une résolution de  $0,25 \text{ mm}$  ( $R_{0,25}$ ), ce qui correspond à  $1,4 \cdot 10^8$  voxels.

Lors du déroulement de la segmentation nous devons gérer les images suivantes en haute résolution :

- la segmentation courante  $L$  ;
- les cartes de distances aux surfaces, pour chaque label, et dans certains cas, les informations de voisinage associées à ces cartes (voir § 4.5.1, p. 71) ;
- l'image estimée à l'aide du modèle d'acquisition  $I \triangleq V(L) * f$  (voir § 4.4.4, p. 68) ;
- La variation d'énergie  $\Delta E_{i,l}(p)$  de chaque contrainte  $i$ , en chaque point  $p$ . Cette image n'est non nulle que sur un nombre réduit de points (voir § 4.2.1, p. 60).

Ces images interviennent à des étapes différentes de la segmentation et ne coexistent pas toutes simultanément, limitant ainsi les exigences en mémoire. Les plus volumineuses sont  $I$  et sa

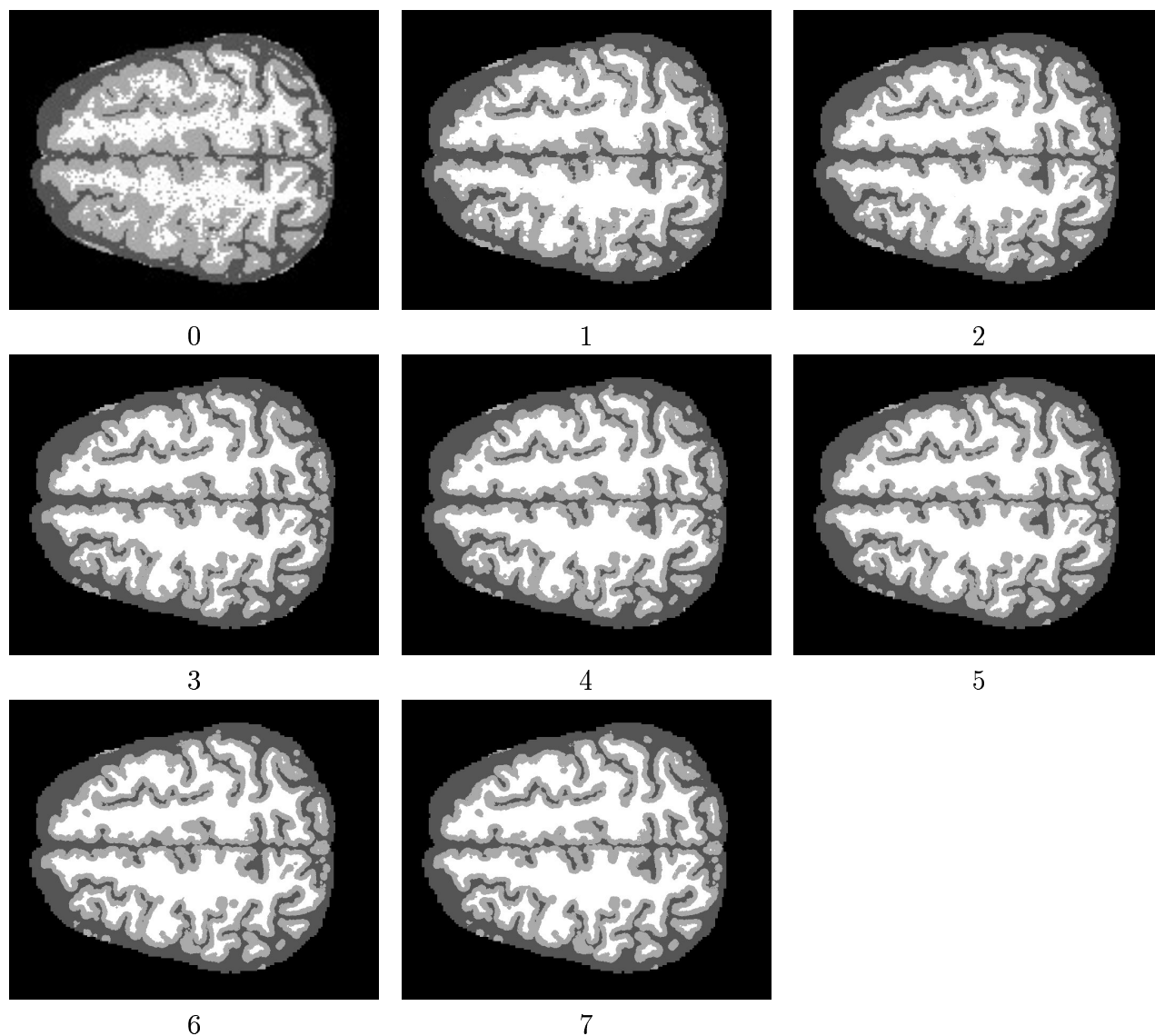


FIG. 5.9 – Les 7 premières itérations d'une segmentation de l'exemple des figures 5.4 et 5.5. Comme le montre cet exemple, la segmentation peut converger après très peu d'itérations. La vitesse de la convergence dépend du rayon d'interaction utilisé et des contraintes appliquées. Lorsque des contraintes contradictoires sont très fortes, la segmentation met plus d'itérations à converger.

transformée de Fourier (voir § 4.4.4, p. 68) qui doivent être représentées en virgule flottante (chacune des deux images fait alors  $R_{0,5}$  : 68Mo,  $R_{0,25}$  : 560Mo). Ces deux images ne sont utilisées que ponctuellement. Les images des contraintes seraient tout aussi volumineuses si nous n'utilisions pas une représentation creuse (décrite ci-dessous). Les images de labels, tel que  $L$ , ne prennent qu'un nombre limité de valeurs  $\{1, \dots, n\}$ , nous pouvons donc utiliser une représentation compacte, décrite également ci-dessous. Pour finir, les cartes de distances sont, elles, représentées en virgule fixe.

### Structures de données

La gestion (en termes de génie logiciel) des différents types d'images évoquées ci-dessus (creuses, compactes, virgule flottante, etc.) pourrait rapidement devenir fastidieuse. Le cadre logiciel (ImLib3D, voir § B.1, p. 156), dans lequel est implémenté ce système de segmentation, repose sur des concepts comme la généricité et les itérateurs, qui assurent la réutilisabilité, transparente, de tous les outils de traitement d'images. Ainsi, des traitements comme l'addition ou le seuillage d'images, sont réutilisables pour tous les types d'images sans nécessiter leur réécriture. Ces concepts seront décrits au paragraphe B.1.

**Images creuses** Chaque contrainte est représentée dans une image  $p \rightarrow \Delta E_{i,l}(p)$ . Dans ces images creuses (voir § 4.2.1, p. 60), seuls les éléments non nuls sont conservés. Une description commune à toutes les images ayant la même structure creuse est utilisée (§ B.2.2, p. 162). Cette structure creuse, ayant une représentation compacte, permet l'accès aléatoire à l'image en un temps logarithmique ( $O(\log_2(n_x))$ ) où  $n_x$  est la largeur de l'image) à l'aide d'arbres binaires, et permet le parcours en temps linéaire des valeurs non nulles. Toutes ces opérations sont transparentes à l'utilisation. Dans un exemple d'application typique, cette structure de données permet de gagner un facteur 13 en termes d'encombrement mémoire, ainsi qu'un gain en performance considérable.

**Images compactes** Grâce à la généricité de son implémentation, ImLib3D permet de gérer et de traiter de manière totalement transparente les images à valeurs entières compactes (§ B.2.2, p. 163). Ainsi, chaque voxel est codé sur un nombre arbitraire (fixe), réduit, de bits. Ceci permet la gestion efficace et compacte (gain mémoire d'un facteur 16, par rapport à une représentation entière) de cartes de segmentation, et autres données du même type.

### Temps de calcul

Les temps de calcul varient en fonction de la résolution de la segmentation, du nombre d'itérations et du volume de la région segmentée. Une itération en résolution  $(0,25 \text{ mm})^3$  dure environ 8 mn pour l'ensemble du cerveau situé au-dessus de la base du lobe frontal (PC-Linux 2.4 GHz). Soit 20 heures si on considère 150 itérations (N.B. en général, il en faut nettement moins, mais on choisit 150 pour garantir la convergence, voir remarque § 5.1.4). L'encombrement mémoire à cette résolution est de l'ordre de 1.7 Go. En résolution  $(0,5 \text{ mm})^3$ , le temps de calcul d'une itération est de l'ordre de 1,2 mn, soit 3 h, pour la convergence.

## 5.2 Perspectives

Le système de segmentation que nous avons décrit paraît prometteur. Cependant, tel qu'il est implémenté au moment de la rédaction de ce manuscrit, son application se limite à la



segmentation des sillons corticaux. Or, le système est conçu pour permettre la segmentation de bien d'autres structures.

**Modélisation des autres structures cérébrales** Il reste donc à mettre en place les modèles nécessaires. Les structures anatomiques qui nous paraissent les plus importantes à décrire sont les structures internes du cerveau (ventricules, noyaux) et les couches externes au cerveau (méninges, crâne, scalp).

**Evolution de l'intensité** Une autre amélioration à l'étude, qui paraît simple à implémenter, est de réévaluer les paramètres d'intensité du modèle d'acquisition au fur et à mesure de l'avancement de la segmentation. En effet, dans l'implémentation actuelle, ceux-ci sont estimés une seule fois, à l'aide de la technique d'histogramme conjoint. Des erreurs dans cette étape ne peuvent donc pas être remises en cause ultérieurement. La même technique d'histogramme conjoint pourrait être utilisée, mais cette fois, entre l'image estimée  $I$  et l'image à segmenter  $I_0$ .

**Évaluation du biais** Dans tout système utilisant des connaissances *a priori*, celles-ci influencent les résultats. Par exemple, dans notre approche, l'épaisseur de la MG est influencée par la contrainte de distribution relative. Si l'objectif d'une étude est précisément de déterminer l'épaisseur de la MG, alors les résultats auront un biais qui dépendra des poids relatifs de la contrainte image et des contraintes *a priori*. Il serait intéressant de mener une étude systématique sur l'importance de ce biais, en fonction des poids des contraintes et de la forme de leurs distribution de probabilités (variance de la sigmoïde).

**Détection d'anomalies** Des objets présents dans une image, non décrits par le modèle *a priori*, donnent lieu à des différences importantes entre l'image  $I$  reconstruite et l'image acquise  $I_0$ . Ces différences peuvent être exploitées pour détecter des anomalies pathologiques (i.e. non décrits dans le modèle *a priori*). Par exemple, dans des images de sclérose en plaques que nous avons segmentées, les lésions de la matière blanche apparaissent comme des taches dans l'image  $I - I_0$ .

### 5.2.1 Atlas statistique

Pour la segmentation des sillons corticaux, les distributions de probabilités des paramètres des contraintes avaient une expression simple (sigmoïde pour la distribution relative, avec un paramètre d'épaisseur constant). Pour des objets de forme plus complexe, comme les structures internes, il est nécessaire d'effectuer un apprentissage, aboutissant à un atlas statistique des paramètres (figure 5.10).

La construction de l'atlas à partir d'un ensemble d'apprentissage  $A_1, \dots, A_n$  limité, exige d'estimer de manière robuste les densités de probabilité souhaitées. Nous allons illustrer cela à l'aide de l'exemple de la contrainte de distribution relative.

**Apprentissage associé à la contrainte de distribution relative** Nous souhaitons apprendre en chaque point  $p$  de l'image, la probabilité  $P(l|d, k, p)$  d'observer un label  $l$  en  $p$  à une distance  $d$  du label  $k$ . Cette probabilité pourrait s'estimer simplement par

$$P(l|d, k, p) = \frac{1}{n} \sum_{1 \leq i \leq n} \delta(A_i(p) = l, D_{A_i, k}(p) = d) \quad (5.2)$$

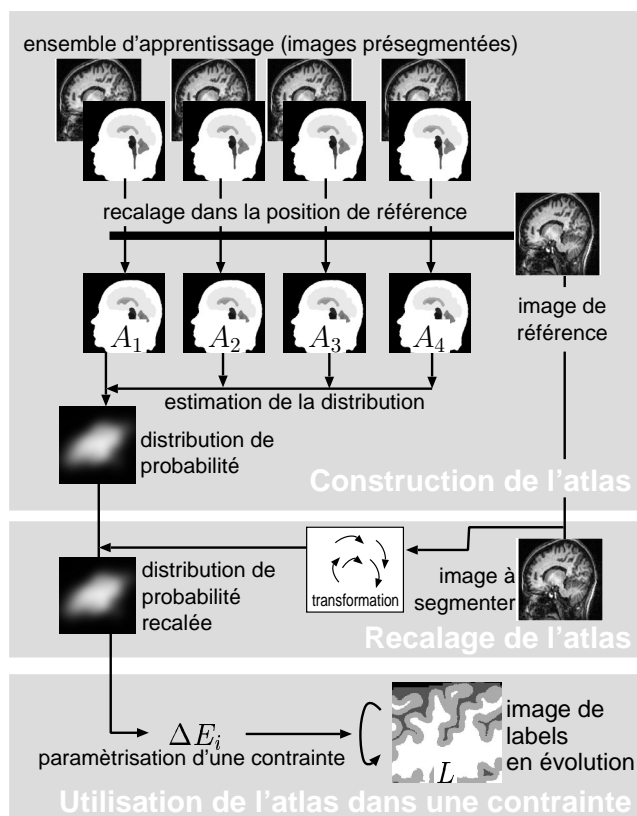


FIG. 5.10 – *Atlas statistique* : Un ensemble d'images pré-segmentées (ensemble d'apprentissage) permet d'estimer les densités de probabilités associées aux différents paramètres des contraintes.

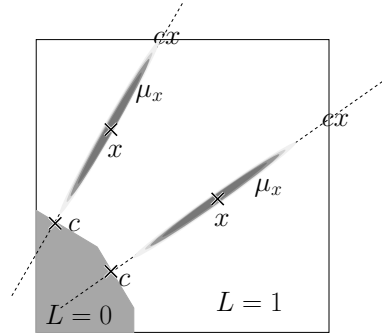


FIG. 5.11 – Pour l'apprentissage de la distribution de probabilité liée à la contrainte distribution relative, des noyaux gaussiens allongés sont utilisés. Ceci permet d'estimer les distributions même si on ne dispose que d'un nombre limité d'échantillons. Le noyau  $\mu_x$  associé à une position  $x$  est allongé dans la direction  $cx$ . Le point  $c$  est la projection de  $x$  sur le label  $L = 0$ .

où  $D_{A_i,k}$  est la distance extérieure au label  $k$  dans l'image d'apprentissage  $A_i$ . Une telle approche nécessiterait un nombre très important d'images d'apprentissage. En effet, dans cette approche, chaque point  $p$  de  $A_i$  contribue uniquement à la densité en  $p$  :  $P(l|d,p)$ . Cette contribution est de  $\delta(A_i(p) = l, D_{A_i,k}(p) = d)$ . Pour densifier  $P$ , nous allons faire contribuer chaque point  $p$  de  $A_i$  sur un voisinage défini par un noyau<sup>5</sup>  $\mu$  (qui peut être gaussien ou non).

$$P^1(l|d,k,p) = \frac{1}{n} \sum_{1 < i < n} \sum_{x \in \Omega} \mu_{i,x}(p-x) \delta(A_i(x) = l, D_{A_i,k}(x) = d) \quad (5.3)$$

Dans le cas de la contrainte de distribution relative, considérée ici, il est intéressant de choisir un noyau ayant une forme allongée (figure 5.11). Les premières expérimentations avec cet estimateur montrent que dès le premier échantillon, on obtient des densités de probabilités exploitables.

### 5.3 Conclusion

Dans cette partie, nous avons proposé une approche de segmentation innovante. Ce nouveau paradigme de segmentation itératif fait évoluer une image de labels  $L$  par pas discrets (basculement d'un label en un point). Notre approche discrète permet d'intégrer à la fois des connaissances *a priori* complexes et une modélisation du processus d'acquisition, fournissant ainsi une segmentation sous-voxel. La description de ces contraintes et de leurs variations dans ce cadre discret est simplifiée par rapport aux approches continues (évolution de surface, level-sets § 3.2.3, p. 51) existantes. La paramétrisation par un atlas permet d'apporter des connaissances anatomiques dans les contraintes. La validité de chaque contrainte et des éventuelles approximations associées, a été démontré par des expérimentations sur des objets simples. Nous avons validé quantitativement l'ensemble de l'approche sur un fantôme numérique, et avons vérifié qualitativement son fonctionnement sur un nombre important d'images (§ 5.1.2, p. 91).

Le système que nous avons décrit est donc fonctionnel. Son application est, à l'heure actuelle, encore limitée à la segmentation des sillons corticaux. Les travaux nécessaires à son extension à l'ensemble du cerveau sont en cours et paraissent prometteurs.

<sup>5</sup>Le noyau est une distribution de probabilité :  $\sum_{\mathbb{N}^3} \mu = 1$ .

Dans notre problématique de détection de changements, cette méthode de segmentation sera utilisée dans une phase de post-traitement. Elle permettra de réduire les fausses détections dans les sillons corticaux.



Troisième partie

Détection de changements



# Chapitre 6

## État de l'art de la détection de changements

### 6.1 Présentation

La détection de changements interimages regroupe un grand nombre de problématiques et de domaines d'application<sup>1</sup>. Nous considérerons ici, principalement trois domaines d'application : la télédétection, les séquences vidéo et l'imagerie cérébrale. Ces trois domaines seront abordés plus en détail aux paragraphes 6.2, 6.3 et 6.4, respectivement.

Au sein d'un même domaine d'application, les objectifs, très variés, déterminent les techniques employées.

**Objectifs possibles** La détection de changements peut se donner pour objectif la détection des changements d'intensité entre images, sans *a priori* sur la taille ou la forme des changements. C'est souvent le cas en télédétection et c'est aussi le cas dans notre application. On supposera, généralement, une certaine cohérence spatiale des changements recherchés. On supposera aussi que les changements n'affectent pas l'image dans son ensemble (changement local).

La détection de changements peut aussi se donner pour objectif de suivre l'évolution ou le mouvement d'un nombre variable d'objets dans une série d'images. Ces approches sont fréquentes dans des séquences vidéos, par exemple dans le suivi de voitures dans des scènes routières. Dans certaines applications, une description plus ou moins précise de l'objet à suivre pourra donner un *a priori* fort utile.

Un objectif différent est la comparaison des caractéristiques (position, taille, forme) d'un objet, extraites à partir des images, pour déterminer et, éventuellement, quantifier un changement. Cette approche est entre autres utilisée sur des lésions de SEP (voir comparaison par segmentation, § 6.4.2) et en télédétection (comparaison de classifications).

Un dernier objectif, très différent de ceux évoqués précédemment, peut être de chercher à déterminer si des images sont semblables, ou non, dans leur globalité. Les domaines d'application sont la détection de changements de scènes dans des séquences vidéo, le watermarking (vérification d'authenticité et protection de droits d'auteur) et la comparaison de documents (recherche de duplicata [9]).

---

<sup>1</sup>Pour le traitement de séquences temporelles de signaux 1D, nous recommandons l'ouvrage classique de Basseville et Nikiforov [3]



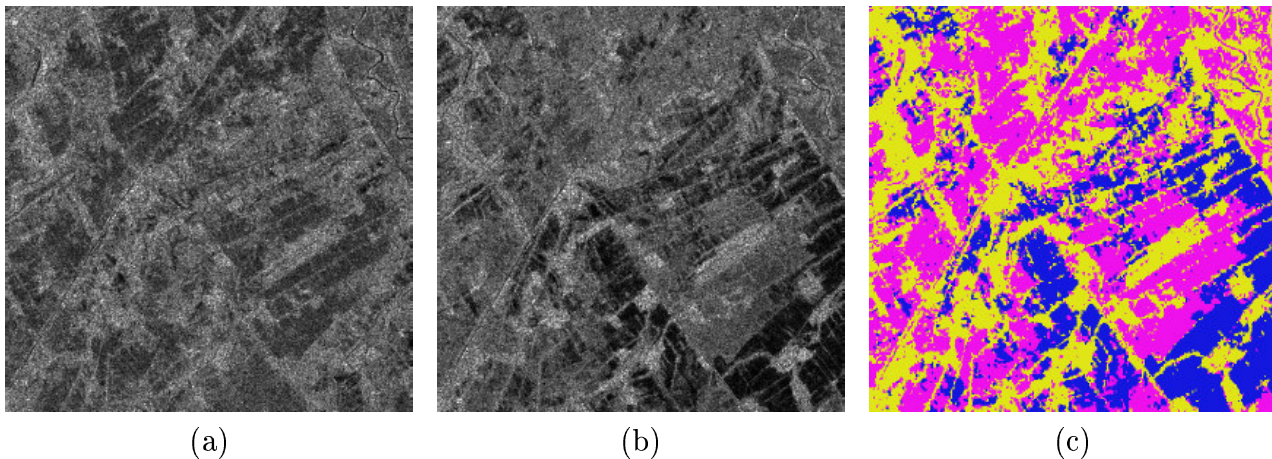


FIG. 6.1 – *Détection de changements (en trois classes) entre deux images RSO (radar), par la méthode de Derrode et al. Les images RSO présentent un fort bruit multiplicatif de speckle. Dans cet exemple, des inondations sont détectées, essentiellement en bas à droite de l'image. (Illustration d'après Derrode et al. [24])*

Nous nous intéresserons ici, plus particulièrement au premier objectif : la détection de changements d'intensité locale.

## 6.2 Détection de changements en télédétection

La télédétection comprend deux principales sources d'images : l'imagerie optique satellitaire et l'imagerie radar. L'imagerie optique est souvent multispectrale, avec des canaux dans les longueurs d'onde visibles et dans l'infrarouge. L'imagerie radar (Radar à Synthèse d'Ouverture, RSO) peut provenir d'un satellite ou d'un avion. Elle présente, d'une part, un fort bruit multiplicatif (figure 6.1), le speckle, mais, d'autre part, a l'avantage d'être utilisable en présence de couverture nuageuse, de jour comme de nuit. Le terme « multitemporel » est employé pour désigner l'acquisition de plusieurs images d'un même lieu. La détection de changements en imagerie multitemporelle [7, 23] trouve des applications importantes dans le suivi et la surveillance de l'urbanisation, de la déforestation, des inondations, des feux de forêt et dans la protection de l'environnement. Sauf cas particuliers, comme des constructions humaines, les régions où surviennent des changements n'ont pas de tailles / formes prévisibles. Il est donc difficile d'introduire des connaissances *a priori*. La détection de changements peut être gênée par des changements d'illumination complexes, de conditions atmosphériques, d'humidité au sol et de calibration des senseurs. Diverses déformations géométriques peuvent intervenir, avec un fort impact sur l'étape de détection [116, 48] (figure 6.2). De nombreuses techniques de recalage sont employées : manuelle (points de contrôle), appariement de points significatifs, recalage affine (on trouvera un état de l'art récent sur le recalage dans [133]).

En imagerie satellitaire optique, l'inclinaison de l'illumination par rapport au soleil (collines, montagnes) influence fortement les intensités mesurées sur les images. Les variations spatiales d'intensité dues à l'inclinaison peuvent être corrigées (correction topographique) à l'aide de modèles numériques d'altitude du terrain [99]. En ce qui concerne la correction de variations globales d'intensité entre acquisitions successives, deux approches très différentes existent. La première, plus répandue, est la recalibration. En effet, les valeurs d'intensité fournies par les capteurs, correspondent à des quantités physiques quantitatives. Certains capteurs

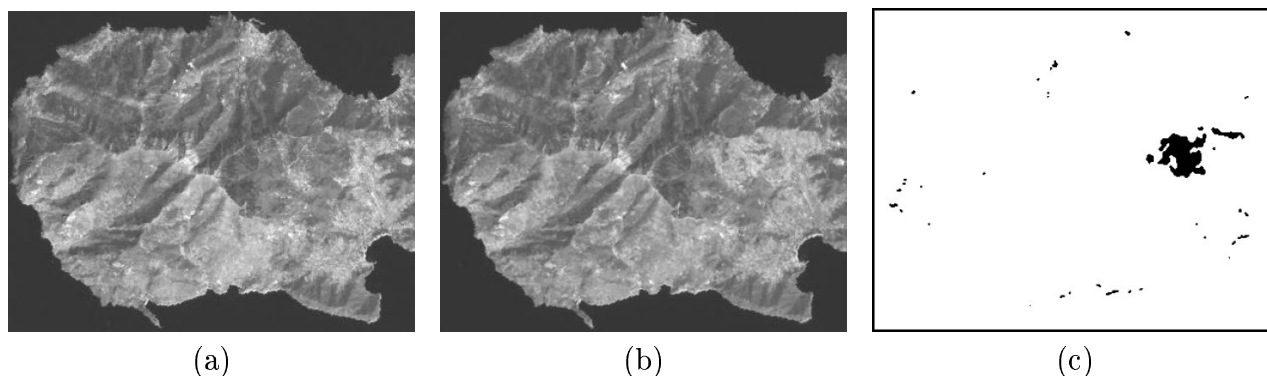


FIG. 6.2 – Détection de changements, par la méthode de Bruzzone et al. , entre deux images Landsat TM de l'île d'Elbe. Une seule bande spectrale est représentée. Des erreurs de détection apparaissent sur les côtes, dus à des légères erreurs de recalage dans des régions ayant un fort contraste. (Illustration d'après Bruzzone et al. [7])

peuvent effectuer une auto-calibration à l'aide de mires embarquées à l'intérieur du satellite. Cependant, même dans ce cas, de légères variations peuvent subsister. Elles sont alors corrigées à l'aide d'un modèle de calibration spécifique à la physique du capteur [46]. La seconde approche, moins répandue, consiste à corriger des différences de calibration entre deux images. Cette seconde approche s'apparente d'avantage à notre problématique. Un modèle paramétrique de la fonction de réflectance bidirectionnelle (BRDF), est construit à partir de l'intensité observée dans l'image [18]. Les différences dans les modèles paramétriques correspondant aux deux images à comparer sont alors minimisées.

Deux approches très différentes sont possibles lors de la comparaison : la comparaison de segmentations obtenues par classification et la comparaison des intensités. Nous nous intéresserons ici à la deuxième. Les méthodes existantes dans ce domaine, emploient généralement l'image de différence (voir [24]), mais il est également possible de calculer le rapport entre les images, ou employer un critère d'information mutuelle. La régularisation des résultats obtenus peut faire appel à des champs de Markov [7] ou à des chaînes de Markov cachées [24].

### 6.3 Détection de changements dans des séquences vidéo

La détection de changements trouve de nombreuses applications dans l'analyse de séquences vidéo : compression de flux vidéo (MPEG), détection de changements de plans, suivi de gestes, vidéosurveillance, surveillance routière, suivi de cibles, etc. Certaines de ces applications exigent un traitement en temps réel.

Des techniques de natures très différentes sont employées pour le suivi d'objets. Une première approche consiste à segmenter l'objet, déterminer ses caractéristiques et ensuite assurer leur suivi dans le temps. Une deuxième approche consiste à effectuer le suivi sans description précise de sa forme, à l'aide de techniques du type contours actifs et level-sets [88]. Nous n'aborderons pas plus en détail ces deux types d'approches, trop éloignées de notre problématique. D'autres approches, plus liées à la nôtre, déterminent les régions de l'image ayant changé (masque de changements), après avoir compensé, entre autres, le mouvement de la caméra ou le mouvement d'ensemble dans les images.

Dans les séquences vidéo, les images sont souvent bruitées et de basse résolution. Elles sont sujettes à des variations d'illumination [114] et à d'autres dégradations. Un autre facteur

à prendre en compte est le mouvement éventuel de la caméra (caméra fixe, variation des paramètres de la caméra - orientation, zoom - ou mouvement général de caméra). La géométrie des déformations induites dans l'image 2D par ces mouvements, est un problème très étudié, dans le cadre de différents modèles de projection. La mise en correspondance géométrique des images (recalage) pourra donc s'effectuer en estimant les variations des paramètres intrinsèques et / ou extrinsèques de la caméra (position, orientation, zoom). Le mouvement apparent dans l'image 2D est de nature différente selon qu'il s'agisse d'objets (scènes) éloignés ou d'objets proches (figure 6.3). L'estimation pourra alors décomposer l'image en régions distinctes [47]. D'autres approches, n'estimant pas directement les paramètres de la caméra, calculent le flot optique entre les images [75]. Celui-ci est semblable aux champs de déformation interimages obtenus par recalage déformable dans le domaine médical (voir § 6.4.2).

Dans des scènes réelles, des changements complexes d'illumination peuvent survenir. L'intensité d'un pixel sur l'image peut alors être répartie entre une composante d'illumination et une composante de réflectance. En supposant que la composante d'illumination a une variation spatiale lente, elle peut être éliminée par filtrage [114, 1]. D'autres approches ne compensent pas les variations d'illumination, mais celles-ci sont prises en compte dans une phase ultérieure de la détection [64].

Une fois les mouvements de caméra et les variations d'illumination compensés, des techniques permettant de décider de la pertinence de changements en chaque région de l'image sont employées. La comparaison pixel par pixel fournit des résultats bruités. Les approches supposent alors, généralement, une certaine cohérence spatiale des changements. Cette connaissance est intégrée à l'aide de diverses approches. Certaines méthodes construisent un modèle statistique de l'intensité des deux images dans une fenêtre entourant chaque pixel [45]. L'utilisation de cette fenêtre permet de débruiter le résultat et ce d'autant plus, que la fenêtre est grande. D'autres méthodes expriment la cohérence spatiale par un *a priori* markovien dans une formulation bayésienne [1, 85, 6] au sein du modèle statistique de détection. Les critères de comparaison sont généralement limités à des modèles (plus ou moins complexes) de l'intensité dans la fenêtre de détection. Plus récemment, Li *et al.* [64] ont proposé un critère intégrant aussi des informations sur la texture. Cette approche semble être plus robuste aux variations d'illumination.

D'autres approches, très différentes, sont possibles. Black *et al.* [5], proposent un modèle génératif pour expliquer des changements constatés. Un modèle expliquant ces changements est construit à partir de plusieurs facteurs : le mouvement apparent, les changements dans les différents types d'illumination, et les changements « iconiques ». Les changements iconiques représentent des événements comme l'apparition ou la disparition d'un objet. Aussi bien le mouvement apparent que les changements iconiques font l'objet d'un apprentissage par ACP. Ce type d'approche n'est possible que si des informations *a priori* fortes sont disponibles sur la scène et son évolution temporelle.

## 6.4 Détection de changements en imagerie cérébrale

Diverses approches de détection de changements, dans des séquences d'images cérébrales, ont été proposées dans la littérature. Les approches existantes utilisent souvent des outils spécifiques consacrés à des applications particulières. Nous proposons, ici, une classification des différentes approches, résumée à la figure 6.4. Une première distinction importante intervient entre les approches cherchant de grands changements structurels et les méthodes se concentrant sur de petits changements, plus localisés.

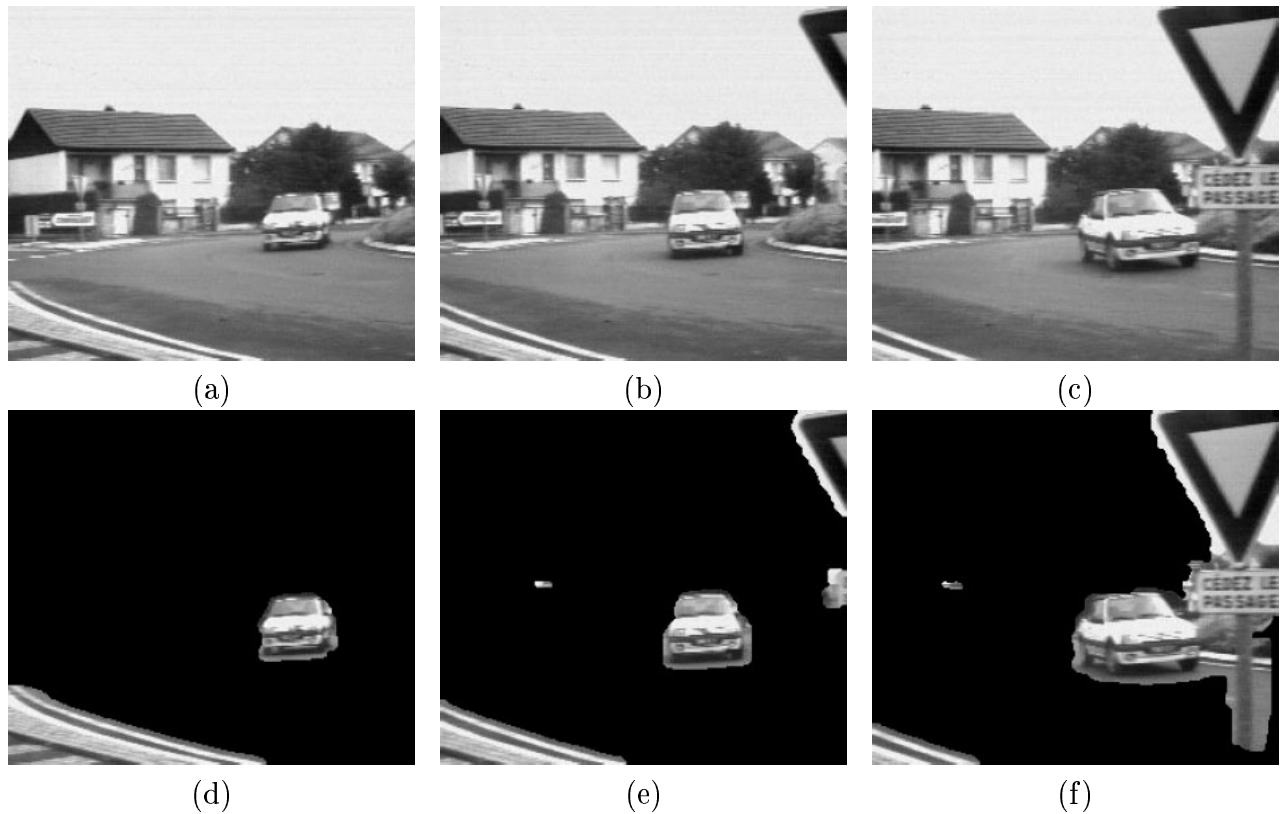


FIG. 6.3 – Détection de changements en présence d'un mouvement de caméra (rotation vers la droite). La compensation du mouvement de la caméra pour les objets de l'arrière plan est correctement géré. La gestion simultanée des objets d'avant plan et d'arrière est un problème délicat (Illustration d'après Odobez et al. [85]).

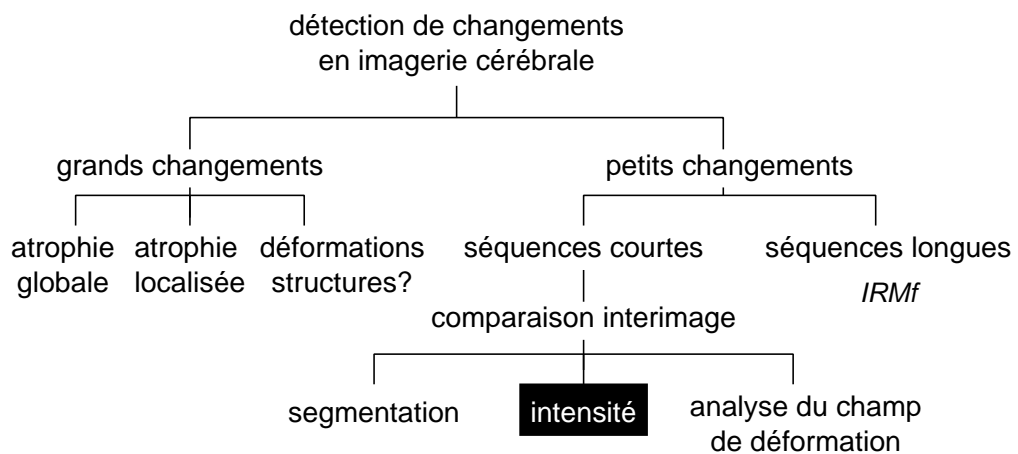


FIG. 6.4 – Classification des méthodes de détection de changements interimages en imagerie cérébrale (voir texte).

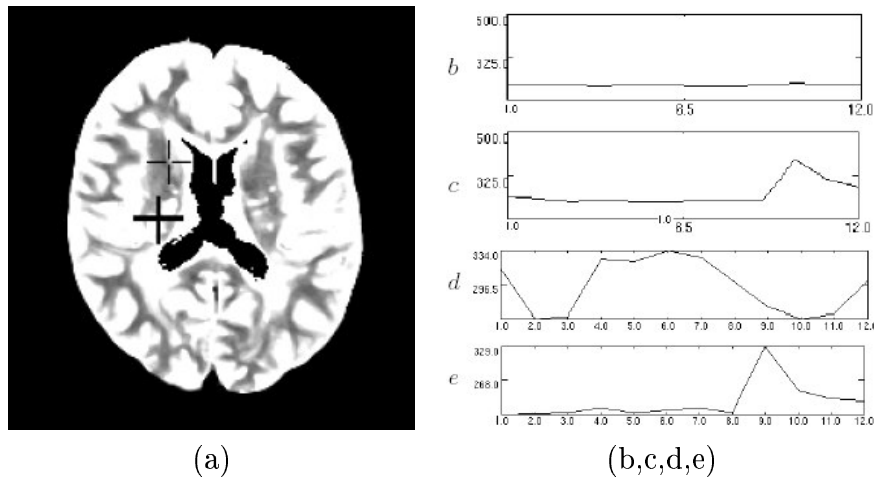


FIG. 6.5 – Gerig et al. [33] proposent une approche de détection de changements en SEP s'appuyant sur l'analyse de la série temporelle en un point donné de l'espace. Plusieurs séries temporelles (b, c, d, e) sont présentées. La série (b) correspond à une position dans la MB, sans lésion, indiqué par la croix fine sur l'image (a). La série (c) correspond à une position sur une lésion en évolution, indiqué par la croix épaisse. Les séries (d, e) correspondent à d'autres évolutions en des points non indiqués sur l'image (a).

Un exemple de détection de grands changements structurels est [102], où les variations du volume global du cerveau sont calculées en estimant le déplacement des surfaces de séparation entre le cerveau et le LCR. Ici, nous nous plaçons dans la deuxième catégorie, analysant les petits changements (éventuellement subtils). On appelle « subtils », les changements difficiles à détecter à première vue par un expert humain.

Les techniques de détection de petits changements peuvent ensuite se classer en deux catégories, suivant la longueur de la séquence d'images traitée : les séquences courtes (en général il s'agira de deux images) et les séquences longues (comme en IRMf). La détection de changements sur des séquences longues s'intéresse d'avantage aux techniques de traitement du signal (1D) appliquées à la série temporelle en une position spatiale (3D) donnée de l'image [3, 30]. De telles approches peuvent être aussi employées dans le suivi d'évolutions de SEP. Par exemple, des tests statistiques sur les séries temporelles sont employés par Gerig *et al.* [33] (figure 6.5). Nous nous focaliserons, ici, uniquement sur les approches traitant des séquences courtes comprenant deux images. Ces approches peuvent être divisées en trois classes (figure 6.4) : les approches segmentant les objets (lésions SEP) avant de les comparer, les approches analysant un champ de déformation interimages pour y déceler des changements et les approches comparant directement l'intensité des images. Elles seront décrites avec plus de détails au paragraphe 6.4.2. Auparavant, nous allons aborder le problème du recalage, primordial pour de nombreuses approches.

### 6.4.1 Recalage interimage

La plupart des systèmes de détection de changements comprennent une phase de repositionnement (recalage) permettant d'obtenir un alignement géométrique précis entre les images, avant de les comparer. Cette première phase (optionnelle), est généralement suivie d'une phase de comparaison interimage. Comme cela a été souligné précédemment à propos des autres domaines d'application, la précision du recalage est très importante pour l'étape suivante de dé-

tection (voir aussi [62]). Le recalage rigide (rotation et translation) est utilisé dans [41, 16, 93]. Cependant, il ne compense pas les erreurs importantes dans les tailles de voxel pouvant apparaître dans les appareils IRM commerciaux (§ 2.1.5, p. 21). Le recalage affine (changements de taille et cisaillement) permet d'obtenir des résultats plus précis [28, 62], mais ne compense pas les déformations complexes. Ces déformations peuvent être d'origine pathologique (atrophie globale du cerveau [102]), d'origine naturelle (gravité [41], changement de pression hydrostatique, déshydratation ...) ou peuvent être dues à des artefacts d'acquisition (shims incorrects ou insuffisants § 2.1.5, p. 21, variation dans la calibration des gradients [28, 102]). Pour compenser ces phénomènes complexes, il est nécessaire d'employer des méthodes de recalage déformable. Le recalage employé, doit être en mesure de compenser des déformations de grande échelle, tout en préservant des déformations très locales, pouvant potentiellement correspondre à des modifications que l'on souhaite détecter. Le recalage déformable a souvent été utilisé pour le recalage inter-sujet [112, 110, 108]. Il a également été employé dans la détection de changements, mais uniquement afin de calculer un champ de déformation analysé par la suite (voir ci-après § 6.4.2). Le recalage déformable n'a pas été utilisé en imagerie cérébrale, à notre connaissance, comme une étape de recalage intra-patient préalable à la détection de changements. En effet, il peut potentiellement détruire des petites modifications interimages. La solution, adoptée ici, est d'employer une technique de recalage *hiérarchique* pouvant recalculer les structures jusqu'à une certaine résolution (échelle) bien contrôlée. Au paragraphe 7.3, p. 119 nous analyserons une technique de recalage itérative, développée par notre équipe [77, 78], permettant d'atteindre cet objectif.

## 6.4.2 Comparaison interimage

Les approches de détection de petits changements sur des séquences courtes (figure 6.4) peuvent être classées dans les trois catégories suivantes.

### Approches de comparaison par segmentation

La comparaison d'objets préalablement segmentés, est une approche de détection de changements envisageable dans certaines applications, comme dans le cas des lésions de SEP. Cette segmentation peut être obtenue manuellement, semi-automatiquement ou pleinement automatiquement. La segmentation manuelle fut la première méthode employée. Il s'agit d'une tâche extrêmement longue, dont la précision, la fiabilité et la reproductibilité ne sont pas assurées. Des méthodes semi-automatiques ont alors été développées [58, 65, 74, 122]. Plus récemment, des méthodes entièrement automatiques ont été proposées [51, 55, 35, 40, 118, 120]. La segmentation automatique de lésions de SEP est un problème difficile. La segmentation de grandes lésions paraît envisageable, mais les petites lésions demeurent difficiles à distinguer du bruit environnant, en particulier dans des modalités où elles ont une valeur d'intensité comparable à celle de structures proches. Une fois les lésions segmentées, leur évolution peut être suivie en déterminant, par exemple, la charge lésionnelle totale. La précision du résultat dépend étroitement de la précision, incertaine, de la segmentation. Ces méthodes ne recalculent pas, en général, les images à comparer, ce qui complique l'évaluation de changements d'intensité au sein d'une lésion.

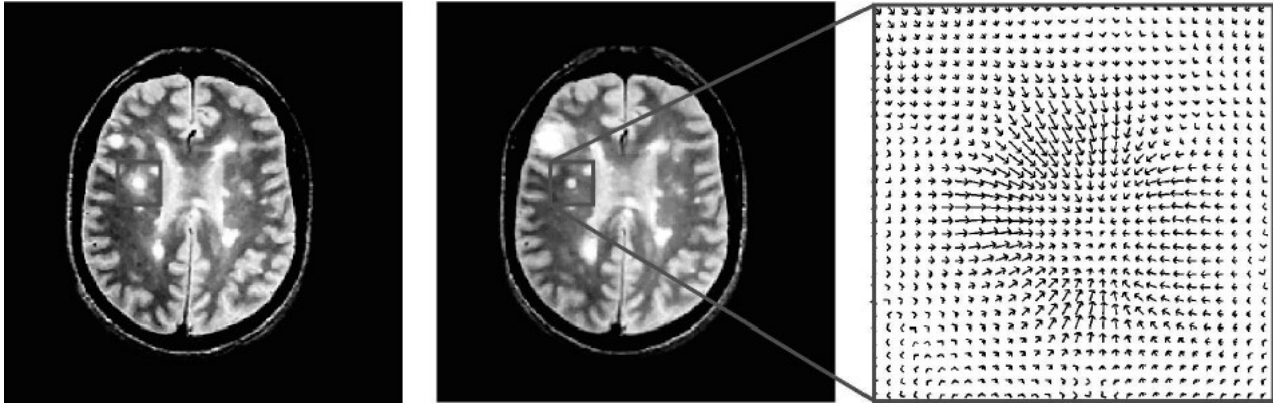


FIG. 6.6 – « Déplacement apparent » dans le champ de déformation calculé par recalage déformable entre les deux images de gauche. L'évolution de lésion apparaît alors comme une contraction dans le champ. La contraction peut être quantifiée à l'aide d'opérateurs tels que le jacobien. (Illustration d'après Rey et al. [93])

### Comparaison par analyse d'un champ de déformation

Ce type d'approches repose sur l'estimation et l'analyse du champ de déformation inter-images. Le champ de déformation est estimé par recalage déformable [77, 108, 110]. Lorsque celui-ci est mené à une résolution suffisamment fine, il déforme la première image, jusqu'à ce qu'elle corresponde étroitement à la seconde. Le champ de déformation, ainsi calculé, contient des informations pertinentes sur les différences entre les deux images (figure 6.6). En particulier, les évolutions de lésions apparaissent comme des contractions ou des dilatations dans ce champ. Cette approche est semblable à l'interprétation du flot optique, employée en analyse de mouvement dans des séquences vidéo [75]. Diverses variantes de ces approches existent en imagerie cérébrale [109, 93, 29, 2, 21]. Par exemple, Thirion *et al.* [109, 93] et Freeborough *et al.* [29] analysent un champ de déformation obtenu par recalage déformable, pour détecter des lésions en évolution ou pour quantifier des variations locales de volume. L'opérateur jacobien permet alors de calculer la variation locale de volume à partir du champ de déformation. Un autre exemple est VBM (Voxel-Based Morphometry [2, 21]), où des déformations de régions segmentées permettent de mesurer une atrophie localisée.

### Comparaison directe d'intensité

Ce dernier type d'approches compare directement l'intensité des images soit voxel par voxel, soit sur une petite région entourant un voxel. Ces approches permettent de détecter tout type de changements, ne faisant aucune hypothèse sur la forme des changements recherchés. Elles exigent tout d'abord de compenser les différences globales d'intensité entre les images (normalisation d'intensité). Cette première étape peut être effectuée avant, pendant, ou après le recalage interimages. Les approches existantes dans ce domaine n'emploient que des fonctions linéaires de correction d'intensité. Cependant, nous avons observé des changements d'intensité fortement non-linéaires entre acquisitions successives (voir § 7.4.1, p. 123). Au paragraphe 7.4.1, p. 123, nous proposons une approche originale de normalisation non-linéaire permettant de compenser ces changements d'intensité. Après recalage et normalisation d'intensité, les images peuvent être comparées. La comparaison peut être effectuée par une approche déterministe ou probabiliste. La plupart des méthodes effectuent une simple soustraction déterministe suivie d'un seuillage manuel ou automatique [86]. Des méthodes probabilistes [45, 62] construisent

un modèle statistique des intensités et du bruit employé pour décider si un changement est significatif ou non.

L'approche que nous proposons ici, repose sur un schéma de détection statistique qui étend des travaux antérieurs [45, 62] en gérant des données multimodales à l'aide d'un modèle gaussien multidimensionnel des intensités des images. La multimodalité permet de réduire sensiblement le taux de fausses alarmes par rapport aux approches monomodales (voir § 8.4.2, p. 136).

## 6.5 Conclusion

Nous avons, dans ce chapitre, décrit brièvement les techniques utilisées pour la détection de changement dans trois domaines d'application. Parmi les domaines non médicaux, la télédétection présente une problématique proche de la nôtre. En effet, dans les applications vidéo, les changements sont, en général, de taille plus conséquente et le fond présente une grande variabilité. Les méthodes récentes, dans ce domaine, se concentrent souvent sur le suivi du mouvement. Il s'agit d'une problématique relativement différente de celle de la détection de changements. La télédétection optique, quant à elle, cherche, comme nous, des changements subtils dans des images plus stables.

Les images des trois domaines d'application considérés, ont des modes d'acquisition (capteurs) et des supports (2D, 3D) très distincts. Aucun lien direct ne peut donc exister entre les artefacts d'acquisition de ces images. Néanmoins, nous avons constaté des analogies entre certains problèmes : distortions géométriques, nécessité du repositionnement et du traitement des variations d'intensité / illumination. Un élément d'explication à ces similitudes est que les appareils d'acquisition sont conçus pour le système visuel humain. Celui-ci est très performant pour ces tâches (repositionnement, compensation de l'illumination et distortions géométriques). Les méthodes que nous proposons au chapitre suivant compensent précisément ces artefacts. Elles pourront donc trouver des applications en dehors du seul domaine de l'imagerie médicale.





# Chapitre 7

## Description de notre approche

### 7.1 Présentation des traitements

La figure 7.1 résume les différentes étapes de traitement, décrites dans les paragraphes suivants. Le déroulement de notre méthode de détection de changements peut être divisé en quatre étapes principales :

1. extraction du cerveau (§ 7.2) ;
2. repositionnement et correction des déformations (§ 7.3, p. 119). Cette étape peut être décomposée en trois phases :
  - (a) recalage affine itératif robuste (§ 7.3.3, p. 121) ;
  - (b) recalage déformable (§ 7.3.4, p. 121) ;
  - (c) rééchantillonnage (§ 7.3.5, p. 122) ;
3. correction des intensités et de l'artefact RF (§ 7.4, p. 122) ;
4. test de détection statistique monomodal ou multimodal (§ 7.5, p. 129).

Environ 12 examens successifs ont été effectués sur chaque patient. À chaque examen, trois images de modalités différentes ont été acquises (voir § 8.1, p. 134). Les étapes d'extraction du cerveau et de recalage sont appliquées à toutes les images (étapes 1 et 2 figure 7.1). Les examens sont ensuite couplés par paires successives : (examen-1, examen-2) (examen-2, examen-3) ... (examen-11, examen-12). Après avoir corrigé les intensités et l'artefact RF (étapes 3 figure 7.1), les images sont finalement comparées à l'aide d'un test de détection statistique (étape 4 figure 7.1).

Pour gérer le nombre important d'images et d'étapes de traitements, un mécanisme de base de données a été implémenté. Tous les traitements sont *entièrement automatiques* : aucune intervention n'est requise de la part d'un utilisateur à aucun moment.

### 7.2 Extraction du cerveau

L'objectif du système de détection est de comparer l'intensité des images en des positions *anatomiques* identiques. Nous avons donc explicitement choisi d'ignorer, en le compensant, tout changement de la taille globale du cerveau. Nous avons aussi observé d'importants mouvements du cerveau à l'intérieur du crâne<sup>1</sup>. Il est donc essentiel de recalculer uniquement les seuls

---

<sup>1</sup>La visualisation des mouvements s'opère à l'aide d'animations. Pour cela, les cerveaux sont recalés entre eux à l'aide de recalage rigide. Dans les animations, on voit alors distinctement le mouvement relatif du crâne par rapport au cerveau.

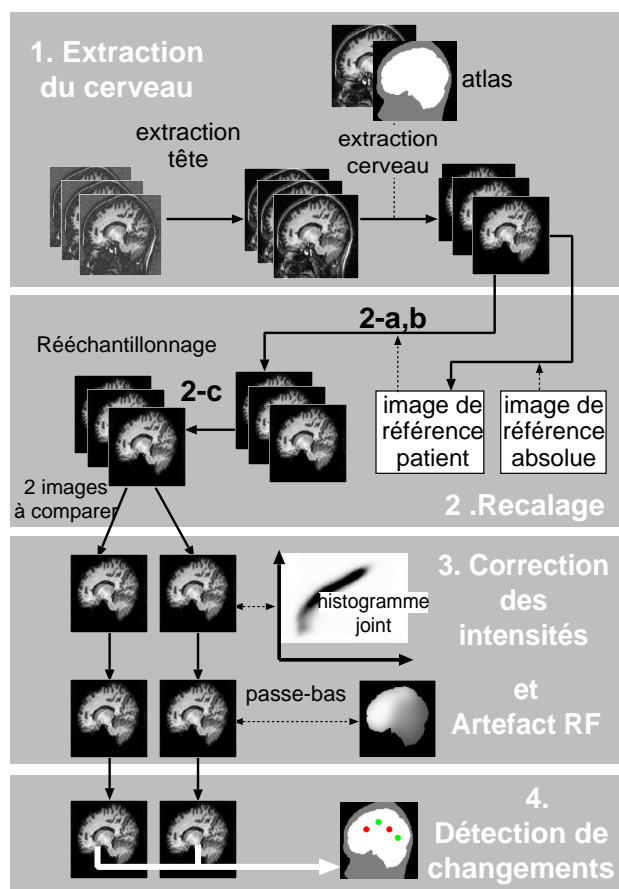


FIG. 7.1 – Vue d'ensemble du processus de détection de changements, montrant les différentes étapes. Pour simplifier, une seule modalité est prise en compte dans ce schéma.

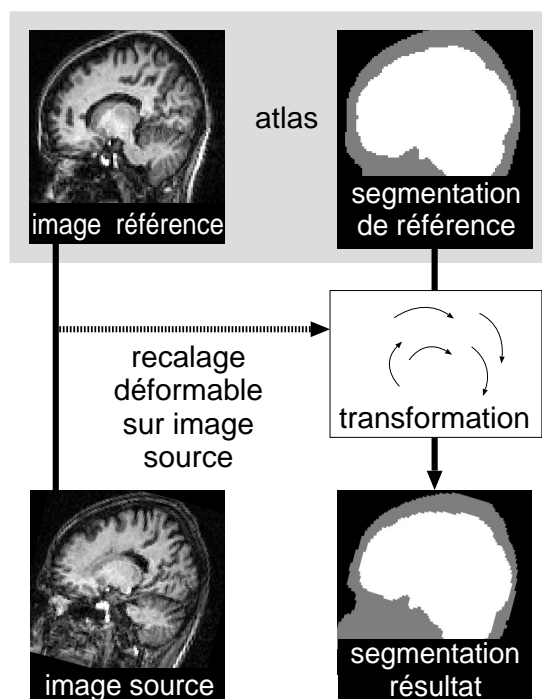


FIG. 7.2 – Extraction du cerveau à l'aide d'un atlas pré-segmenté. Le champ de transformation entre l'image de référence et l'image source (patient) est estimé en utilisant une technique de recalage déformable. Le champ de transformation obtenu permet de plaquer la segmentation de référence (atlas) sur l'image source (patient).

cerveaux entre eux sans le crâne ni le bruit de fond extérieur susceptibles de perturber alors la précision du recalage.

Dans cette étape d'extraction du cerveau, la première phase consiste à séparer l'ensemble de la tête du bruit de fond extérieur. Cette première segmentation n'a pas besoin d'être extrêmement précise mais se doit d'être fiable, étant donné le nombre important d'images devant être traitées automatiquement. Cette segmentation de la tête est obtenue à l'aide de techniques de traitement d'image bas niveau : seuillage automatique par la méthode d'Otsu [87] et opérations morphologiques (érosion, remplissage).

La deuxième phase est l'extraction du cerveau à partir de la tête pré-segmentée à la phase précédente. Une technique de segmentation à base d'atlas est utilisée [78] (figure 7.2). Ici, l'atlas est une image de référence ayant été préalablement pré-segmentée semi-automatiquement à l'aide d'une technique de ligne de partage des eaux [8]. Cette image de référence est recalée l'image à segmenter à l'aide de recalage déformable (§ 7.3.4, p. 121). L'extraction du cerveau souhaitée est alors obtenue en appliquant le champ de déformation (résultat du recalage) sur la segmentation associée à l'image de référence (voir figure 7.2).

## 7.3 Repositionnement et correction de déformations

### 7.3.1 Présentation

La détection automatique de changements entre images est extrêmement sensible aux problèmes de repositionnement. Comme le remarquent Lemieux *et al.* dans [62], dans les régions présentant un fort gradient d'intensité, (sur les bords des lésions, par exemple), un décalage

entre images de moins de 0,05 voxels peut produire des fausses détections. Nous nous trouvons alors clairement dans un contexte sous-voxel et une approche très précise pour le recalage ainsi qu'une interpolation de haute qualité (§ 7.3.5) sont indispensables.

Les recalages rigides (rotation, translation) et affines (agrandissement, cisaillement), généralement utilisés, ne sont pas suffisants pour compenser les déformations complexes qui apparaissent entre acquisitions successives (voir § 6.4.1, p. 112 pour une description de ces déformations). Pour corriger ces déformations, nous proposons une approche de recalage affine itératif robuste, suivie par un recalage déformable hiérarchique (décrit au paragraphe 7.3.4 et dans [77, 78], voir aussi § 8.5, p. 140 pour une discussion sur les compromis impliqués). Une fois déterminée, la transformation géométrique finale, les images sont rééchantillonnées à l'aide d'interpolation B-spline (§ 7.3.5, p. 122, [107]).

Chaque transformation géométrique intermédiaire est effectuée à l'aide d'interpolation B-spline de degré cinq. Ceci ne comprend pas les itérations internes au recalage qui emploient, quant à elles, une interpolation linéaire. En effet, nous avons constaté que l'interpolation B-spline de haute qualité n'est pas nécessaire dans les itérations internes au recalage. Par contre, elle est importante lors du rééchantillonnage final, une fois la transformation géométrique finale calculée. L'utilisation de l'interpolation linéaire dans les étapes internes du recalage permet de gagner en temps de calcul, tout en préservant la qualité sous-voxel du recalage final.

### 7.3.2 Détermination des images de référence pour le recalage

Comme le font remarquer Smith *et al.* [102], il est important que les images à comparer soient traitées de façon équivalente au cours des différentes phases menant à la détection. Si une image particulière d'une série d'un patient était choisie comme position de référence, et que les images de cette série étaient recalées sur cette référence, alors cette image particulière ne serait pas transformée géométriquement. Or, toutes les autres images subiraient une transformation géométrique. Nous proposons donc un schéma de références et de recalages successifs, garantissant que toutes les images soient traitées de façon équivalente. Il garantit aussi que les images finales à comparer sont recalées à l'aide de recalages intra-patient monomodaux. Ce type de recalage est bien plus précis et fiable que les recalages inter-patients ou multimodaux.

Une image de chacune des trois modalités d'un sujet est choisie, en dehors de la base d'images. Ces trois images sont soigneusement recalées entre-elles à l'aide de recalage multimodal affine (étape 1 de la figure 7.3). Ces trois images servent de position de référence absolue pour l'ensemble de la base. Remarquons que le recalage affine permet de compenser des erreurs dans les tailles des voxels qui ne seraient pas compensées par un recalage rigide.

Le recalage de la série d'images correspondant à un patient donné se fait en deux étapes. D'abord, une image arbitraire de chacune des trois modalités de ce patient, est choisie. Ces trois images servent ainsi de référence pour ce patient. Elles sont recalées sur les références absolues respectives à l'aide d'un recalage inter-patient affine monomodal (étape 2 de la figure 7.3). Le recalage inter-patient affine est fiable, mais d'une précision insuffisante. On affine donc le résultat du recalage précédent à l'aide d'un recalage multimodal affine intra-patient supplémentaire (étape 3 de la figure 7.3).

L'avantage de l'ensemble de schéma est que nous pouvons désormais recalculer chaque image patient de la série temporelle à l'aide d'un recalage affine itératif robuste puis d'un recalage déformable, intra-patients et monomodaux (étape 4 de la figure 7.3). Ce type de recalage est plus précis et plus fiable. De plus, toutes les images recalées de l'ensemble de la base multi-patients sont situées alors dans la même position géométrique de référence (absolue).

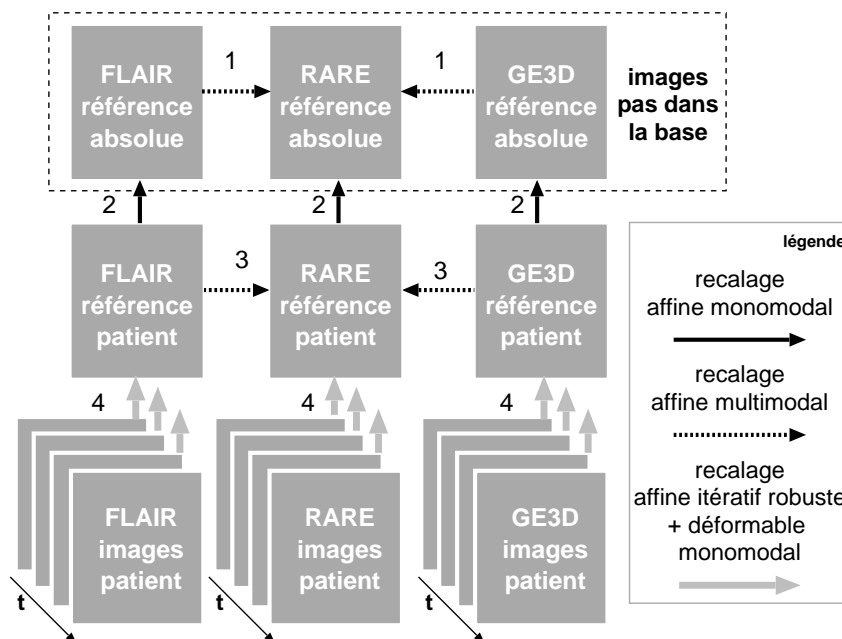


FIG. 7.3 – Images de référence utilisées pour le recalage. Les chiffres (1, 2, 3, 4) désignent l'ordre chronologique des recalages. Cette succession de références (référence absolue et références patient) et de recalages permet de n'employer que des recalages fiables (intra-patient, monomodal) pour les images finales qui sont comparées. Toutes les images se retrouvent, de surcroît, dans la même position.

Dans son implémentation actuelle, tous les algorithmes de recalage utilisent l'information voxel et minimisent une fonction d'énergie (coût) globale à l'aide de l'algorithme standard du simplex. Des fonctions de coût quadratique sont employées pour les recalages monomodaux tandis que le critère de l'information mutuelle est utilisé pour les recalages multimodaux. Le recalage déformable est décrit ci-dessous (§ 7.3.4, p. 121).

### 7.3.3 Recalage affine itératif robuste

L'étape 4 de la figure 7.3, comporte deux phases de recalage : le recalage itératif affine robuste, présenté ici, et le recalage déformable. Cette phase de recalage affine fournit une initialisation nécessaire à la phase suivante de recalage déformable. Le recalage affine corrige diverses erreurs, dont celles dans les tailles des voxels. Cependant, les différences dans les masques du cerveau des deux images à recaler (extraction du cerveau § 7.2, p. 117) peuvent biaiser significativement ce recalage. Pour résoudre ce problème, un affinement itératif du masque du cerveau est introduit dans le processus de recalage. Chaque itération effectue un recalage affine, puis recalcule les masques du cerveau. Le nouveau masque est l'intersection de ceux associés aux images précédentes. En pratique, on n'observe pas de variation significative des masques après deux itérations.

### 7.3.4 Recalage déformable

La méthode de recalage déformable, présentée ici, est une approche multi-résolution développée par notre équipe [77, 78]. Elle démarre à une résolution grossière puis progresse vers une résolution plus fine. Elle estime un champ de transformation paramétrique défini sur une base

multi-résolution de fonctions B-spline. À chaque niveau de résolution (échelle), la méthode minimise une énergie quadratique par une technique d'optimisation de type quasi-Newton. À une échelle  $i$  le champ de déformation est divisé en  $(2)^i$  cellules dans chaque direction. La résolution finale peut être choisie en fonction de la taille des structures que l'on souhaite recalcrer : en arrêtant le recalage à une basse résolution, on compensera uniquement les déformations à grande échelle, alors qu'en continuant le recalage à des résolutions plus élevées, on compensera progressivement davantage de détails entre les images recalées. Le choix de la résolution optimale dans notre application particulière sera discuté au paragraphe 8.5, p. 140.

### 7.3.5 Rééchantillonnage

Comme il a été remarqué précédemment, le résultat du recalage doit fournir une correspondance sous-voxel de haute qualité entre les images à comparer. Une approche performante d'interpolation est donc indispensable. L'interpolateur idéal *sinc* doit toujours, en pratique, être tronqué. Le *sinc* tronqué fournit des résultats incorrects et est inefficace en termes de temps de calcul. Un grand nombre de méthodes d'interpolation ont été proposées (synthèse proposée dans [61]). Récemment, Thevenaz et Unser [107] ont proposé une stratégie d'interpolation innovante et efficace. Habituellement, l'interpolation est exprimée comme la reconstruction d'une image à support continu  $I(x)$  à partir d'une image échantillonnée  $I_k$  en utilisant une fonction d'interpolation  $\phi_{int}$  (*sinc* par exemple) :

$$I(x) = \sum_{k \in \mathbb{Z}^3} I_k \phi_{int}(x - k) \quad (7.1)$$

où  $I_k$  sont les valeurs de  $I(x)$  sur une grille discrète de points. Ce problème a été reformulé dans [107] par :

$$I(x) = \sum_{k \in \mathbb{Z}^3} c_k \phi(x - k) \quad (7.2)$$

où les  $c_k$  sont des coefficients calculés (à partir de l'image), qui ne sont plus les valeurs exactes de l'image sur la grille discrète. Par conséquent, la fonction  $\phi$  n'a plus besoin d'être strictement interpolante. Ce changement de perspective ouvre la porte à de nouvelles stratégies d'interpolation rapide et de grande qualité. L'interpolation est alors décomposée en deux étapes. La première étape calcule les coefficients par un filtrage récursif de l'image. La deuxième étape ressemble à l'interpolation courante, mais utilise un noyau non interpolant. Dans notre système de détection de changements, une transformation géométrique finale est calculée en composant les transformations résultant des différentes étapes de recalage affine et déformable. La transformation est alors appliquée en utilisant un noyau spline d'interpolation de degré cinq [107].

## 7.4 Correction des intensités et de l'artefact RF

Les étapes précédentes ont permis de corriger les différences géométriques entre les images. Nous allons maintenant corriger les différences d'intensité liées aux conditions variées d'acquisition. Nous corrigerons d'abord les différences globales d'intensité à l'aide d'une technique de normalisation d'intensité sur l'histogramme conjoint (§ 7.4.1). Ensuite, nous corrigerons les différences d'intensité diffuses dues à l'artefact RF (§ 7.4.2, p. 129).

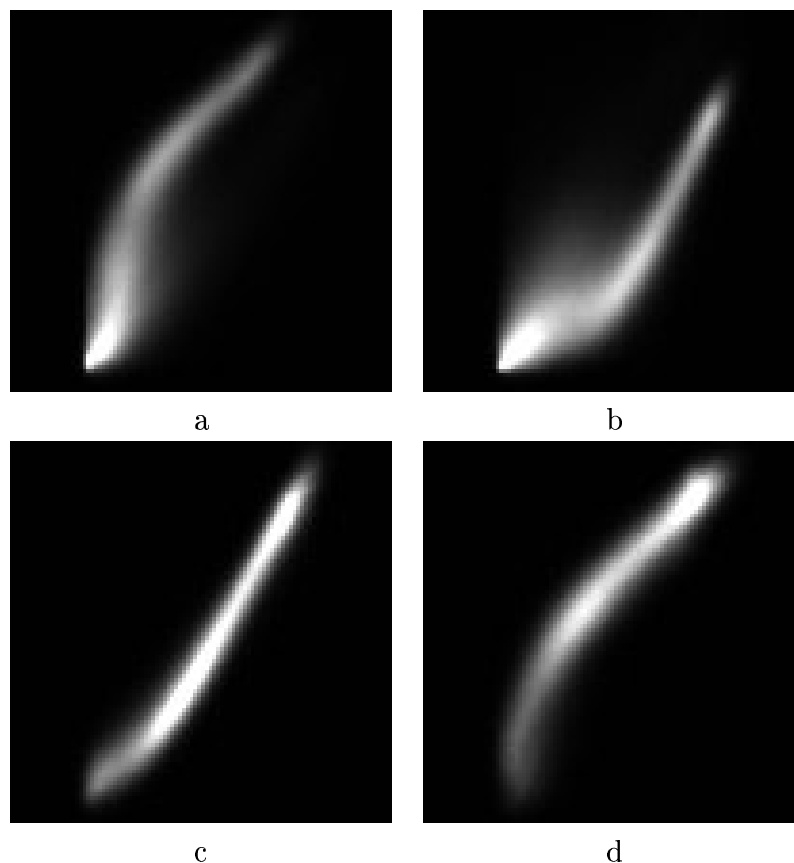


FIG. 7.4 – Quatre histogrammes conjoints entre paires images recalées d'un même patient présentant des non-linéarités importantes. La normalisation d'intensité linéaire, généralement utilisée, ne permet pas de corriger ces défauts. Dans les exemples (a) et (b) les histogrammes conjoints sont calculés sur l'ensemble de la tête (en excluant le fond). Dans les exemples, distincts, (c) et (d) ne sont considérés que les voxels appartenant au cerveau.

### 7.4.1 Normalisation d'intensité sur l'histogramme conjoint

L'intensité sur une image associée à un même tissu peut varier de manière significative entre acquisitions successives d'une même modalité. Ceci est particulièrement gênant lors de la comparaison d'images, mais peut s'avérer également dans d'autres applications nécessitant de connaître l'intensité associée à un tissu (voir utilisation dans la segmentation § 4.6.1, p. 85). La solution employée dans diverses approches [62] est d'utiliser un modèle linéaire décrivant les changements d'intensité. L'intensité de l'image corrigée  $I_{cor}$  est alors considérée comme une fonction linéaire  $f(x) = ax + b$  de l'image  $I_{src}$  originale :  $I_{cor} = f(I_{src})$ . La fonction de transfert d'intensité  $f$  est dans ce cas calculée sur les moyennes et écarts types des images à comparer  $I_{src}$  et  $I_{ref}$ .

Cependant, nous avons constaté que ce modèle était insuffisant. Comme le montrent les histogrammes de la figure 7.4, les variations d'intensité entre deux acquisitions peuvent être fortement non-linéaires. Nous proposons ici une méthode d'estimation d'une fonction de transfert d'intensité  $f$  non-linéaire basée sur une estimation précise de la densité conjointe des intensités des deux images. La connaissance de la fonction de transfert  $f$  nous permet alors d'effectuer une normalisation d'intensité non-linéaire pour la détection de changements<sup>2</sup>.

<sup>2</sup>La fonction de transfert est aussi employée pour déterminer des niveaux d'intensité dans la partie II,



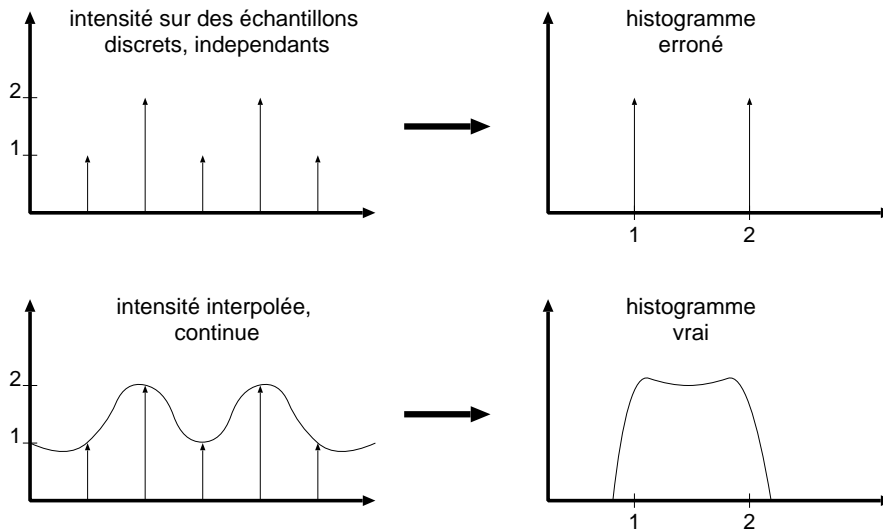


FIG. 7.5 – Exemple 1-D de calcul d'un histogramme à l'aide d'interpolation. Si les échantillons représentent une image continue (Shannon), le calcul de l'historgramme à partir d'échantillons discrets peut donner des résultats erronés. Cet effet est particulièrement marqué si des fréquences élevées sont présentes dans l'image.

### Estimation de probabilités conjointes

L'estimation d'un histogramme ou d'un histogramme conjoint s'effectue habituellement à l'aide de techniques paramétriques ou non paramétriques d'estimation de densités de probabilité [26]. Ces techniques considèrent des séries d'échantillons indépendants et requièrent un grand nombre d'échantillons pour obtenir une estimation acceptable d'une densité de probabilité. L'estimation de densités de probabilité multidimensionnelles (comme les densités conjointes) nécessite un nombre particulièrement élevé d'échantillons. Dans le cas spécifique des histogrammes d'images, l'estimation est habituellement faite en considérant chaque pixel (ou voxel dans le cas 3D) de l'image en tant qu'échantillon indépendant. Le nombre d'échantillons disponibles est alors limité et peut s'avérer insuffisant. Or, dans le cas d'une image, les échantillons  $I_{src}(k)$ ,  $k \in \mathbb{Z}^3$  pris sur une grille discrète de points correspondent à un signal continu dual  $I_{src}(p)$ ,  $p \in \mathbb{R}^3$  (théorème de l'échantillonnage de Shannon). Nous disposons donc d'un nombre *infini* d'échantillons pour estimer la densité de probabilité, ce qui nous permet d'en obtenir une estimation précise et sans bruit. Une deuxième conséquence de cette interprétation est que les techniques d'estimation conventionnelles d'historgrammes à partir des pixels considérés comme indépendants produisent des résultats erronés, surtout dans les régions contenant des hautes fréquences (voir figure 7.5). Une première solution serait d'utiliser les techniques à base de pixels indépendants mais sur des images fortement sur-échantillonnées. Malheureusement, en pratique, cela devient calculatoirement rédhibitoire. Une autre solution consiste à calculer analytiquement l'historgramme de l'image continue ; elle s'avère impossible pour un interpolateur idéal (*sinc*), mais pas pour un interpolateur linéaire mono-dimensionnel. Dans ce cas,  $I_{src}$  et  $I_{ref}$  sont localement linéaires par rapport à la direction considérée  $x$  :  $I_{src}(x) = ax + b$ ,  $I_{ref}(x) = cx + d$ . On a donc localement  $I_{ref} = \frac{c}{a}I_{src} + d - \frac{b}{a}$ . La contribution locale à l'historgramme conjoint  $J(u, v)$  est alors un segment de droite. Remarquons que dans l'approche conventionnelle, un pixel fournissait un seul point à l'historgramme conjoint, alors

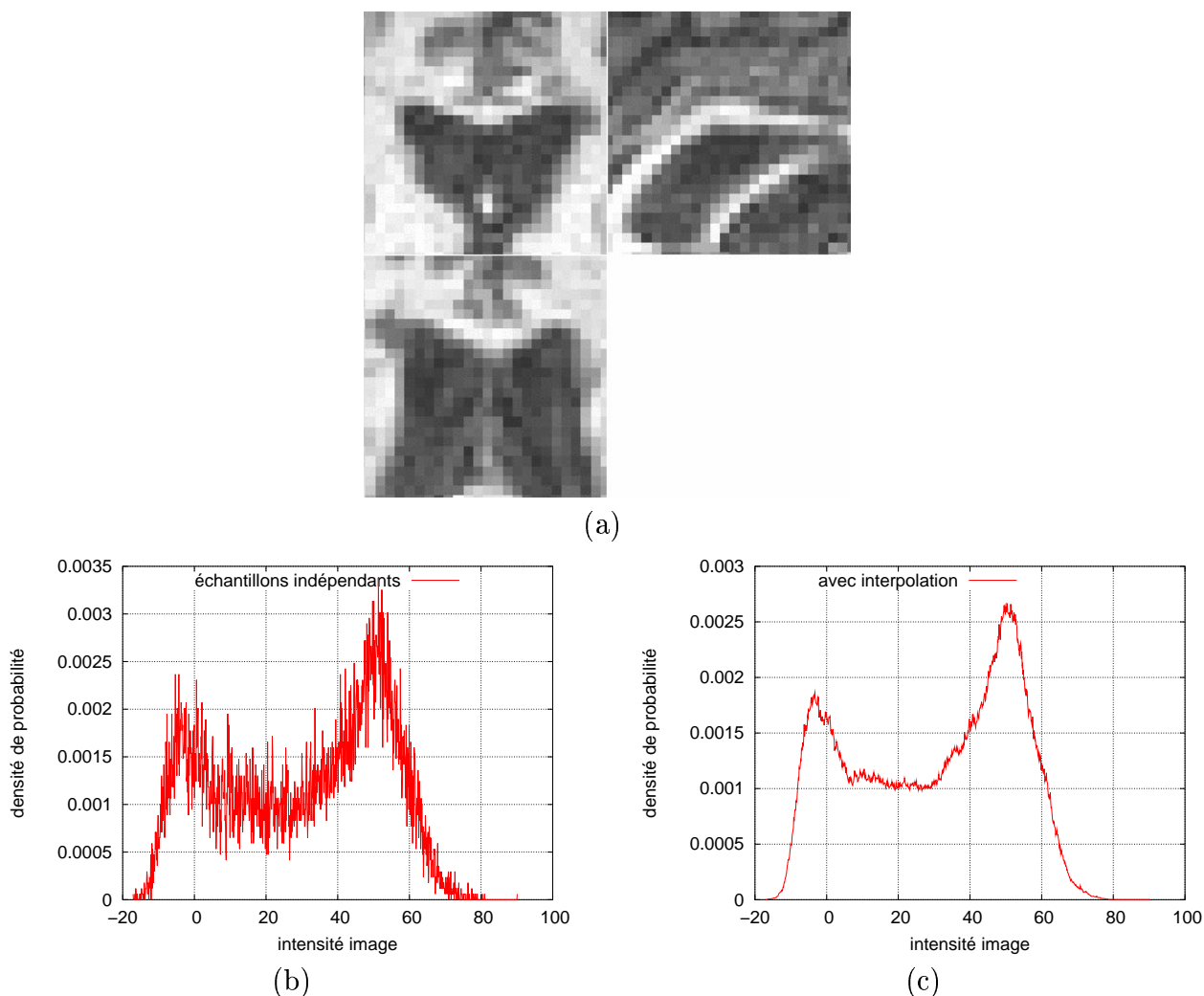


FIG. 7.6 – Histogrammes obtenus à partir d'une portion volumique (a) de cerveau. (b) Histogramme calculé à partir d'échantillons indépendants. (c) Histogramme calculé en utilisant la technique présentée dans le texte. Ce deuxième histogramme est nettement moins bruité. Notez que les techniques d'estimation par noyaux, habituellement utilisées, permettent aussi d'obtenir des histogrammes lisses, mais au prix d'un étalement (lissage) important, impliquant par conséquent d'une perte de précision.

qu'ici, il s'agit d'un segment de droite. L'ensemble de l'histogramme sera alors couvert avec moins d'échantillons. L'interpolation linéaire mono-dimensionnelle est appliquée dans les trois directions et les résultats sont moyennés. Les histogrammes obtenus par la technique présentée ici sont en accord avec ceux obtenus en sur-échantillonnant et s'avèrent de qualité supérieure à ceux obtenus par des approches conventionnelles (voir figures 7.6, 7.7).

### Détermination de la fonction de transfert d'intensité entre deux images

Nous cherchons à estimer une fonction de transfert d'intensité  $f$ , appliquée à une image source  $I_{src}$  qui minimise l'écart entre l'image corrigée  $I_{cor} = f(I_{src})$  et une image de référence  $I_{ref}$ . Nous cherchons donc la fonction  $f$  qui va minimiser

$$E_{image} = \int_{\mathbb{R}^3} C(f(I_{src}(p)), I_{ref}(p)) dp \quad (7.3)$$

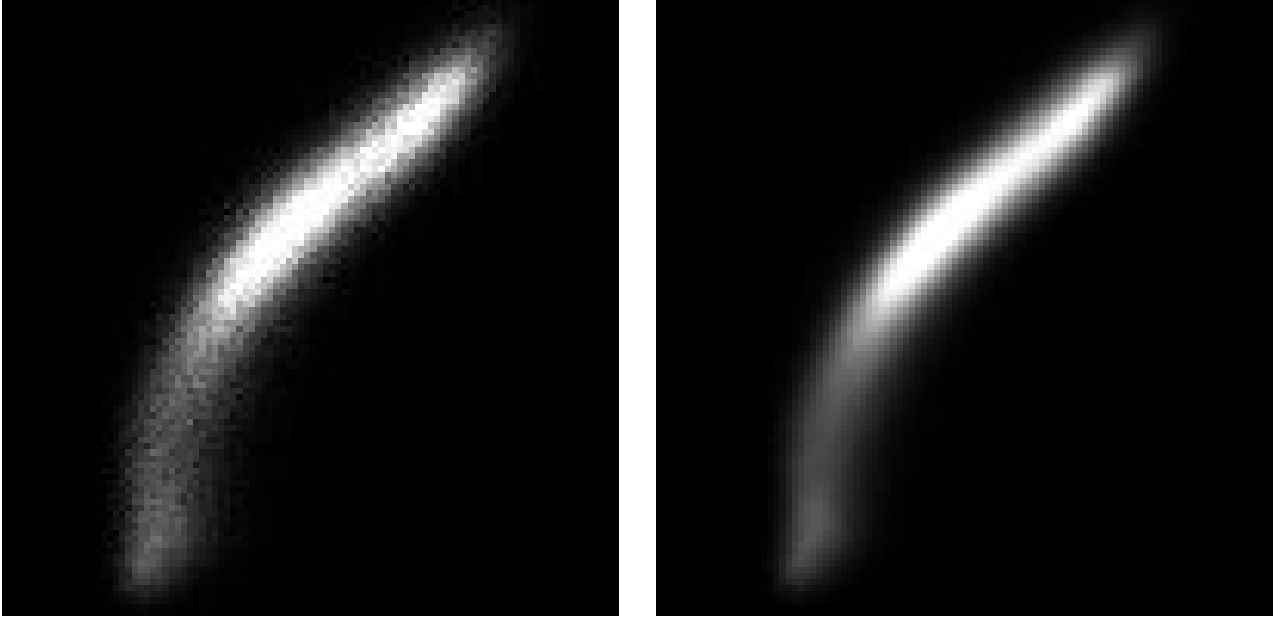


FIG. 7.7 – Histogrammes conjoints sur deux images du cerveau d'un même patient : à gauche, l'histogramme conjoint calculé à partir d'échantillons indépendants ; à droite, l'histogramme conjoint calculé en utilisant la technique présentée dans le texte. Ce deuxième histogramme est nettement moins bruité.

où  $C$  est une fonction de coût (dans le cas quadratique  $C(a, b) = (a - b)^2$ ).

Le calcul effectif de cette minimisation requiert de sommer sur l'ensemble du support de l'image à chaque itération de la minimisation. Cela implique un temps de calcul excessif. Par conséquent, nous allons changer le domaine d'intégration, en le découpant en parties d'intensité constante pour chaque image. Nous obtenons alors :

$$E_{image} = \int_{v_0=-\infty}^{\infty} \int_{v_1=-\infty}^{\infty} J(v_0, v_1) C(f(v_0), v_1) dv_1 dv_0 \quad (7.4)$$

Ceci prouve que toute l'information d'intensité est contenue dans l'histogramme conjoint  $J$  des deux images. La fonction  $f$  minimisant cette erreur pose deux difficultés : elle est bruitée pour les intensités peu présentes et elle possède un biais systématique.

**Lissage** Pour remédier à la première difficulté, nous introduisons un deuxième terme d'erreur dont l'objectif est de régulariser la fonction  $f$  en minimisant sa courbure :

$$E_{lissage} = \int (f'')^2 \quad (7.5)$$

Pour représenter la fonction lissée, ainsi que pour limiter le nombre de paramètres à considérer lors de la minimisation, nous choisissons la représentation paramétrique suivante de  $f$  :

$$f = \beta_m * \sum_{0 \leq i < q} v_i \delta_i \quad (7.6)$$

où  $\beta_m$  est une fonction  $\beta$ -spline de degré  $m$ , les  $\delta_i$  sont des diracs unidimensionnels espacés régulièrement,  $q$  est le nombre de paramètres et  $v_0, \dots, v_{q-1}$  sont les paramètres définissant  $f$ . En pratique, nous utiliserons  $m = 1$ . La fonction  $f$  est alors linéaire par morceaux.

**Biais** Pour comprendre le problème du biais, considérons un exemple. Supposons que  $I_{ref}$  et  $I_{src}$  sont deux images parfaites (sans bruit) du cerveau divisé en trois classes (LCR, MG, MB). Les erreurs de recalage entre  $I_{ref}$  et  $I_{src}$  feront que les voxels de MB de  $I_{src}$  seront en correspondance essentiellement avec des voxels MB de  $I_{ref}$  mais aussi avec des voxels de MG et de LCR de  $I_{ref}$ . La MB ayant l'intensité la plus élevée, la fonction  $f$  calculée précédemment sera systématiquement biaisée, donnant une intensité inférieure à ce qu'elle devrait être. En d'autres termes, la densité de probabilité de  $P(v_1|v_0)$  n'est pas symétrique autour de son maximum. Nous allons donc remplacer cette densité par une densité gaussienne centrée sur la valeur maximale observée  $P(v_1|v_0) : P(v_1|v_0) \simeq \mathcal{N}(\mu_{v_0}, \sigma_{v_0})(v_1)$ . La probabilité conjointe peut s'écrire à l'aide de la formule de Bayes  $P(v_0, v_1) = P(v_1|v_0)P(v_0)$ . L'histogramme conjoint est lié à la probabilité conjointe par un facteur de normalisation  $\gamma : P(v_0, v_1) = \frac{1}{\gamma}J(v_0, v_1)$ . D'où  $J(v_0, v_1) = \gamma P(v_0)\mathcal{N}(\mu_{v_0}, \sigma_{v_0})(v_1)$ . L'erreur image s'écrit alors :

$$E_{image} = \gamma \int_{v_0=-\infty}^{\infty} \int_{v_1=-\infty}^{\infty} P(v_0)\mathcal{N}(\mu_{v_0}, \sigma_{v_0})(v_1)C(f(v_0), v_1)dv_1dv_0 \quad (7.7)$$

En choisissant un coût  $C$  quadratique, nous obtenons :

$$E_{image} = \gamma \int_{v_0=-\infty}^{\infty} P(v_0) \left( (f(v_0) - \mu_{v_0})^2 + \sigma_{v_0}^2 \right) dv_0 \quad (7.8)$$

**Minimisation** Un coefficient  $\alpha$  est introduit pour ajuster le poids relatif du lissage et de l'information image. L'erreur totale à minimiser s'écrit alors :

$$E = \alpha E_{image} + E_{lissage} = \alpha\gamma \int_{v_0=-\infty}^{\infty} P(v_0) \left( (f(v_0) - \mu_{v_0})^2 + \sigma_{v_0}^2 \right) dv_0 + \int (f'')^2 \quad (7.9)$$

Nous utilisons la méthode de minimisation multidimensionnelle du simplexe proposée par Nelder *et al.* [80] déterminer les paramètres  $v_0, \dots, v_{q-1}$  minimisant l'erreur  $E$ . Remarquons que le calcul de l'erreur totale  $E$ , effectué à chaque itération de la minimisation, est très rapide, ne comportant que  $n_{discr} + q$  opérations ( $n_{discr}$  est le nombre de pas choisi pour discrétiser l'intensité de  $I_{src}$ ).

**Résultats** Cette technique de calcul de la fonction de transfert  $f$  a été appliquée sur 76 paires d'images. Quatre exemples sont illustrés à la figure 7.8. La robustesse des résultats a été vérifiée visuellement sur 75 paires d'images. Leur précision a été vérifiée sur des images simulées. Les résultats sont satisfaisants et l'amélioration par rapport aux modèles linéaires est significative. Cet outil paraît prometteur pour diverses applications et plusieurs développements pourraient en étendre la portée. Dans le cas inter-patient, il serait intéressant d'utiliser un atlas de variabilité construit en calculant la variance de l'intensité en chaque point après avoir recalé une série d'images sur une référence. Cet *a priori* simple permettrait d'accorder plus de confiance aux régions à faible variabilité. Un autre développement envisageable serait d'étudier les effets liés à la dispersion du nuage dans l'histogramme conjoint. Ces effets, visibles aux bords de l'histogramme (figure 7.8 : (a,b) en haut, à droite (d) en bas à gauche), proviennent du bruit dans l'image de référence. Par exemple, sur la figure 7.8(d), pour une intensité de  $I_{src}$  de 15, la probabilité maximale  $P(v_1|I_{src} = 15)$  est effectivement en  $I_{ref} \simeq 15$  comme l'indique la fonction de transfert. Or, nous souhaiterions intuitivement avoir une fonction de transfert presque verticale en  $I_{src} = 20$ . L'étude de ce problème est en cours.

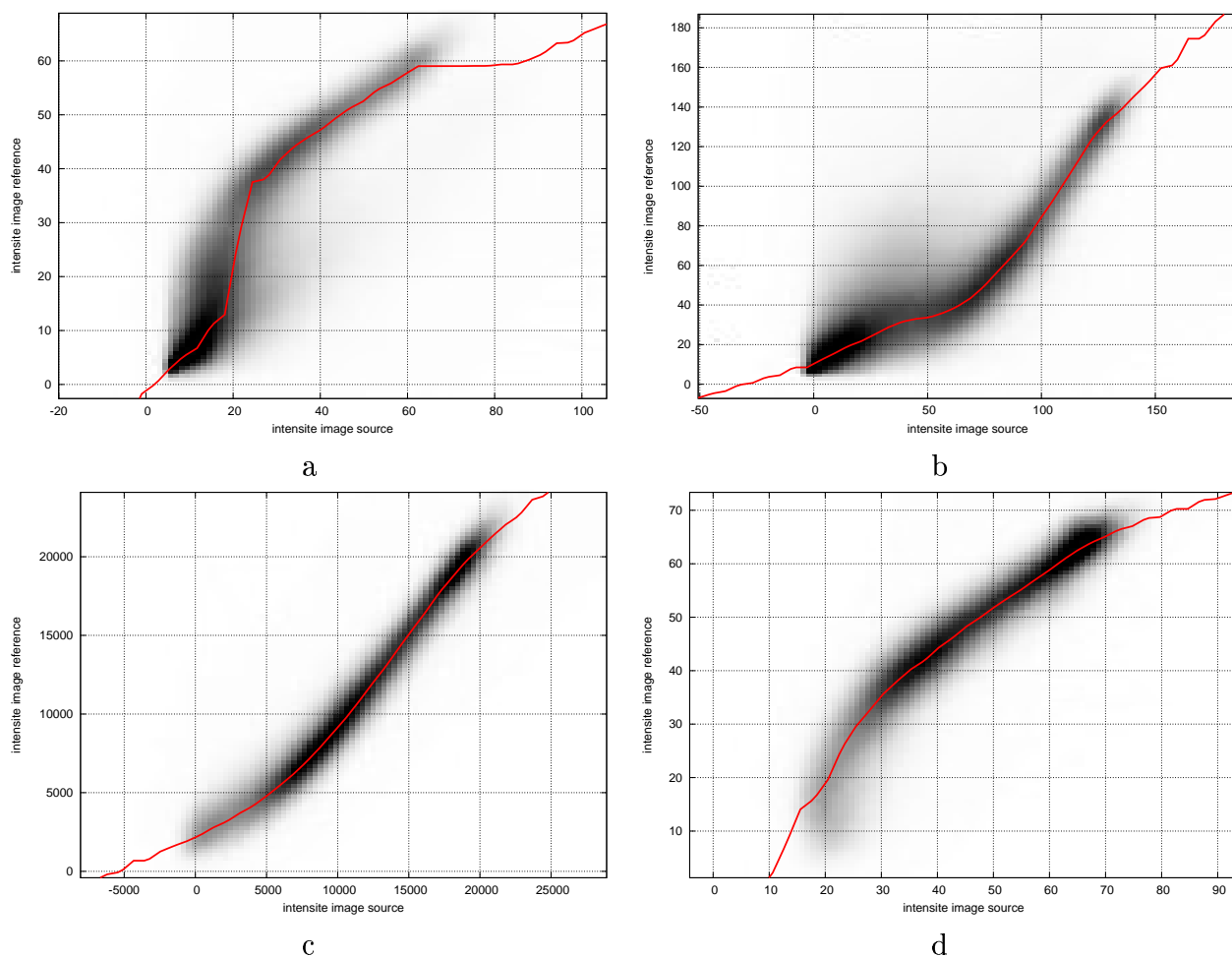


FIG. 7.8 – Fonctions de transfert d'intensité calculées sur les 4 histogrammes conjoints présentés figure 7.4

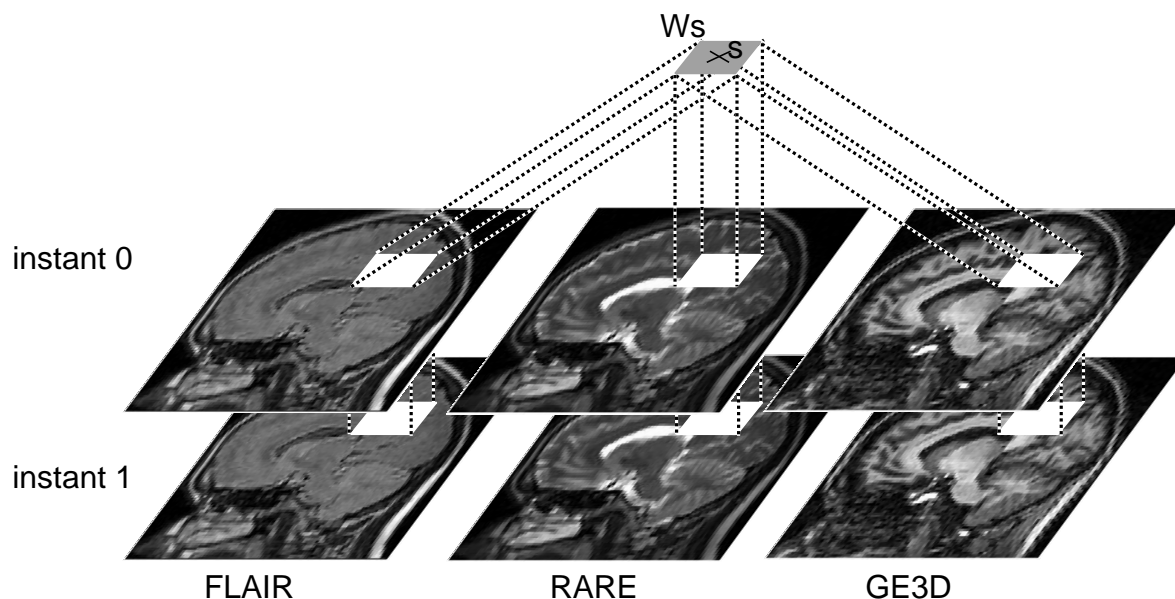


FIG. 7.9 – Pour la détection de changements, on considère un modèle statistique des intensités dans une fenêtre 3D  $W_s$  centrée sur un voxel  $s$

#### 7.4.2 Artefact RF

Après la correction géométrique et la normalisation d'intensité, il subsiste encore quelques artefacts. Parmi eux, le plus important, potentiellement, est la variation diffuse d'intensité résultant des inhomogénéités dans le gain de l'antenne RF (voir § 2.1.5, p. 21). Le problème principal dans l'élimination de cet artefact, est qu'il est difficile de distinguer les variations réelles des images des variations dues à l'artefact [12]. Cependant, dans le cadre de notre approche, une solution simple s'offre à nous. Une fois les images recalées et normalisées (en intensité), une simple soustraction fournit une image composée de bruit, des artefacts résiduels et éventuellement des changements interimages réels. Les artefacts résiduels sont essentiellement composés des artefacts de mouvement et des artefacts de pulsation ayant une moyenne nulle et une fréquence spatiale plutôt élevée. Les changements d'intensité réels, que l'on souhaite détecter, sont de petite taille par rapport à l'artefact RF. L'application d'un filtre gaussien passe-bas ( $\sigma = 15$  mm) sur l'image différence permet d'isoler aisément l'artefact RF par rapport aux autres changements décrits précédemment. L'artefact RF isolé est ensuite soustrait à l'une des deux images.

### 7.5 Modèle statistique pour la détection de changements

Les principaux artefacts étant corrigés, il reste à comparer les images. En d'autres termes, on doit décider en chaque point  $p$  des images, si, effectivement, le changement est significatif ou non. L'approche immédiate serait de soustraire les deux images. Cependant, cela produirait des résultats bruités, ne permettrait pas d'intégrer la multimodalité et ne fournirait pas un critère de décision objectif. La théorie statistique de la détection de changements fournit une approche éprouvée pour la comparaison [53]. On considère une fenêtre  $W_s$ , de taille fixe, centrée sur un voxel  $s$  (figure 7.9). Dans cette fenêtre  $W_s$ , on utilise un modèle statistique simple de la distribution d'intensité. La taille de la fenêtre est choisie à  $3 \times 3 \times 3$  voxels (voir § 8.5,

p. 140 pour une discussion sur les effets de ce paramètre). Le test généralisé du rapport de vraisemblance (generalized likelihood ratio test : GLRT [45, 53]) calcule le rapport entre les vraisemblances de deux hypothèses :

$H_0$  : il n'y a pas de changement entre les images  $I_1$  et  $I_2$  dans la fenêtre  $W_s$

$H_1$  : il y a un changement significatif entre  $I_1$  et  $I_2$  dans  $W_s$

Les vraisemblances des hypothèses sont exprimées à l'aide de leur densité de probabilités (ddp) paramétriques. On interprète l'hypothèse  $H_0$  par le fait qu'il n'y a pas de changement dans les paramètres  $\theta_0$  des ddp de  $I_1$  et  $I_2$ . Dans l'hypothèse  $H_1$ , les paramètres  $\theta_1 \neq \theta_2$  des ddp des intensités dans les images  $I_1$  et  $I_2$  sont distincts.

Dans le cadre du GLRT, l'estimation  $\hat{\theta}$  des paramètres inconnus  $\theta$  est faite par le maximum de vraisemblance [53]. Le rapport généralisé de vraisemblance s'écrit alors :

$$R_{GLRT} = \frac{p(\mathcal{I}_1; \hat{\theta}_1)p(\mathcal{I}_2; \hat{\theta}_2)}{p(\mathcal{I}_1; \hat{\theta}_0)p(\mathcal{I}_2; \hat{\theta}_0)} \quad (7.10)$$

où  $\mathcal{I}_1$  et  $\mathcal{I}_2$  sont les valeurs respectivement de  $I_1$  et de  $I_2$  dans la fenêtre  $W_s$ .

### Détection statistique : le cas monomodal

Nous avons choisi un modèle simple pour les ddp paramétriques : l'intensité dans la fenêtre  $W_s$  est modélisée comme une valeur constante  $\mu$  avec un bruit gaussien, i.i.d., de moyenne nulle et d'écart type  $\sigma$ . Diverses études, en particulier dans la détection de changement dans des séquences vidéo, ont montré que des modèles d'intensité simples (comme le modèle : constante plus bruit gaussien additif) sont plus robustes que des modèles ayant davantage de paramètres [75]. Des modèles d'intensité d'ordre supérieur (i.e. non-linéaires) n'apportent qu'une amélioration négligeable dans les résultats de détection. Le modèle d'intensité simple, que nous avons choisi, reste efficace, même si le bruit n'est pas parfaitement gaussien.

La variance du bruit est supposée constante sur toute l'image et est considérée identique sur les deux images. Elle est estimée à l'aide d'un estimateur standard sur l'image de différence décrite au paragraphe 7.4.2, qui ne contient plus que le bruit résiduel. Le seul paramètre  $\theta$  des ddp restant à estimer au cours de la détection est donc la valeur moyenne :  $\theta = \mu$ . L'estimée par le maximum de vraisemblance de  $\mu$  sur une fenêtre  $W_s$  d'une image  $I$  est  $\hat{\mu} = \frac{\sum_{r \in W_s} I}{n}$ , où  $n$  est le nombre de voxels dans  $W_s$  (ici  $n = 27$ ).

En prenant le logarithme de l'équation 7.10 on obtient le rapport de log-vraisemblance :

$$l = \frac{1}{2\sigma^2} \sum_{p \in W_s} \begin{aligned} & -(I_1(p) - \hat{\mu}_1)^2 - (I_2(p) - \hat{\mu}_2)^2 \\ & + (I_1(p) - \hat{\mu}_0)^2 + (I_2(p) - \hat{\mu}_0)^2 \end{aligned}$$

qui se simplifie en :

$$\begin{array}{c} H_1 \\ \frac{\sqrt{n}}{2\sigma} |\hat{\mu}_2 - \hat{\mu}_1| > \lambda \\ H_0 \end{array} \quad (7.11)$$

L'hypothèse  $H_1$  est choisie si  $l > \lambda$  et l'hypothèse  $H_0$  est choisie si  $l < \lambda$ , où  $\lambda$  est un seuil.

### Détection statistique : le cas multimodal

Le cas monomodal de détection de changements présenté ci-dessus, peut être aisément étendu au cas multimodal. Lorsqu'on dispose de  $m$  modalités, la valeur en chaque point de l'image sera considérée comme un vecteur à  $m$  dimensions. La probabilité d'observer les valeurs d'intensité  $I(p), p \in W_s$  pour une moyenne  $\mu$  s'écrit alors :

$$p(\mathcal{I}; \mu) = \left( \frac{1}{\sqrt{(2\pi)^m |C|}} \right)^n e^{-\frac{1}{2} \sum_{p \in W_s} (I(p) - \mu)^T C^{-1} (I(p) - \mu)} \quad (7.12)$$

où  $n$  est le nombre de voxels dans  $W_s$  et  $C$  est la matrice de covariance du bruit. Comme pour la variance dans le cas monomodal, la matrice de covariance est aussi supposée constante sur toute l'image et est estimée à l'aide d'estimateurs ergodiques standards [26, 53]. Le rapport de log-vraisemblance multimodal s'écrit alors :

$$l = -\frac{1}{2} \sum_{p \in W_s} \begin{aligned} & (I_1(p) - \hat{\mu}_1)^T C^{-1} (I_1(p) - \hat{\mu}_1) \\ & + (I_2(p) - \hat{\mu}_2)^T C^{-1} (I_2(p) - \hat{\mu}_2) \\ & - (I_1(p) - \hat{\mu}_0)^T C^{-1} (I_1(p) - \hat{\mu}_0) \\ & - (I_2(p) - \hat{\mu}_0)^T C^{-1} (I_2(p) - \hat{\mu}_0) \end{aligned} \quad (7.13)$$

Remarquons que le « poids » relatif attribué à chaque modalité par cette approche dépend de la matrice de covariance du bruit. Le poids des modalités bruitées sera donc automatiquement diminué par cette approche.

#### 7.5.1 Post-traitements

L'étape de détection statistique de changements décrite précédemment indique quels voxels ont changé. Dans notre système de détection de changements, nous souhaitons détecter des *événements* pouvant correspondre au changement de plusieurs voxels adjacents (voir § 8.4, p. 135). On regroupe alors les voxels voisins ayant changé à l'aide d'une méthode de groupage simple : on part de la valeur  $x$  de vraisemblance maximale qui est ensuite diminuée itérativement. Chaque diminution de  $x$  correspond à des voxels soit ajoutés à un groupement existant s'ils en sont voisins, soit ajoutés à un nouveau groupement. À chaque groupement est attribué la vraisemblance du voxel ayant la plus forte vraisemblance dans le groupement. Les groupements sont alors triés dans l'ordre de vraisemblance décroissante. L'utilisateur verra finalement une série de groupements, triée par ordre décroissant de confiance, représentant les événements détectés. Il pourra aisément naviguer entre les différentes détections (figure 8.1). La tâche d'évaluation et de rejet des fausses détections est alors très rapide, prenant généralement moins d'une minute pour chaque paire d'images. Afin de faciliter cette tâche, nous avons choisi de ne conserver que les 30 premiers groupements ; les suivants étant considérés comme n'ayant pas été détectés.





# Chapitre 8

## Protocole expérimental, résultats

Le système de détection de changements, présenté au chapitre précédent, a été évalué par deux neurologues sur une base de plus de 200 images réelles. Il a aussi été évalué en utilisant des évolutions simulées. Nous présentons, ici, la logique de notre méthodologie.

L'évaluation de la performance d'un système de détection de changements est une tâche difficile. Il n'est pas possible d'accéder à une vérité terrain solide et de nombreux changements subtils sont sujets à débat, même après analyse détaillée par des experts. De fortes variabilités dans la charge lésionnelle totale évaluée par différents experts ont été observées, entre autres par Mitchell *et al.* [74]. Nous avons aussi observé d'importants désaccords entre experts.

Nous proposons deux approches pour l'évaluation :

- 1 l'utilisation d'évolutions simulées sur des paires d'images de sujets sains (§ 8.3) ;
- 2 la comparaison entre les résultats de détection obtenus par un expert et ceux obtenus par le système automatique (§ 8.4).

L'utilisation de données simulées a l'avantage de fournir une vérité terrain bien contrôlée, qui permet d'évaluer précisément l'impact des différents paramètres. Cependant, il est largement reconnu que les données simulées sont souvent biaisées. Elles ne reproduisent pas toujours de manière réaliste tous les facteurs complexes qui peuvent apparaître dans des données réelles. De plus, des changements réels du cerveau peuvent apparaître chez des patients sains (voir § 6.4.1, p. 112). Les évolutions de lésion simulées ont donc été utilisées pour déterminer comment les caractéristiques d'une évolution influencent notre système de détection. En particulier, nous avons évalué l'influence de la taille d'une évolution.

En complément, le protocole expérimental décrit au paragraphe 8.4, évalue le système de détection dans une application réelle. Comme il n'est pas possible de déterminer la vérité terrain, nous avons opté pour une approche dans laquelle un premier expert détecte manuellement les changements dans les paires d'images (comme il l'aurait fait dans sa pratique clinique), ensuite le système automatique accomplit la même tâche, et finalement, un deuxième expert arbitre entre les deux. Cette méthodologie mesure les gains qui seraient obtenus en assistant le neurologue avec notre système de détection automatique. L'absence de vérité terrain implique que cette méthodologie n'est pas parfaite. Par exemple, si une évolution est manquée à la fois par l'expert-1 et par le système automatique, elle n'apparaîtra pas dans l'évaluation. Comme l'explique le paragraphe 8.4, les deux experts ont ainsi des rôles très différents.

## 8.1 Patients

Les images utilisées dans cette évaluation proviennent de 8 patients atteints de SEP rémittente. Les examens ont eu lieu une fois tous les deux mois au cours d'une période de deux ans. À chaque examen, des images dans les trois modalités (FLAIR, RARE, GE3D) ont été acquises.

## 8.2 Imagerie

Les images IRM 3D isotropes ont été acquises sur un imageur corps entier standard (Tomikon S200 Bruker, Karlsruhe, Allemagne) en utilisant une sonde de crâne. Trois acquisitions ont été effectuées à chaque examen. La première est une acquisition 3D RARE transverse (FOV = 25.6 cm ; taille de la matrice  $128 \times 128$  ; épaisseur de coupe 2 mm ; temps d'inversion 1.3 secondes), la seconde est une acquisition 3D *fluid attenuated inversion recovery* (FLAIR) (FOV = 25.6 cm ; taille de la matrice  $128 \times 128$  ; épaisseur de coupe 2 mm) et la troisième est une acquisition 3D pondérée en T1 (GE3D) (FOV = 25.6 cm ; taille de la matrice  $128 \times 128 \times 128$ ). Aucune mesure n'a été prise pour repositionner précisément les patients entre les examens.

## 8.3 Évolutions simulées

Les lésions simulées ont un profil gaussien ayant été additionné aux images IRM réelles dans les trois modalités. L'intensité du profil gaussien est choisie identique à celle observée sur des lésions de SEP réelles. Plusieurs tailles de lésions ont été simulées. La taille de la lésion est déterminée par l'écart type de la gaussienne  $\sigma_r$ , ce qui correspond approximativement au rayon de la lésion à mi-intensité. Les évolutions ont été considérées comme des lésions apparaissant entièrement entre deux examens. Chaque lésion est ajoutée à une image d'un témoin sain, puis cette image est comparée à une autre image du même témoin, acquise deux mois auparavant. La chaîne entière de traitements a été appliquée à toutes les images. Plusieurs expériences ont été effectuées, chacune avec une ou plusieurs lésions simulées. Les lésions simulées ont été suffisamment écartées pour ne pas interférer entre elles.

$\sigma_r$ (voxels)	0.25	0.5	0.6	0.7	0.75	1
$p_{det}$	0%	14%	50%	100%	100%	100%

Le tableau précédent montre comment la probabilité de détection  $p_{det}$  des lésions simulées dépend de leur taille  $\sigma_r$  (exprimée en voxels : 1 voxel = 2 mm). On constate que les lésions sont correctement détectées si elles sont plus grandes que  $\sigma_r > 0.6$  voxels. On considère ici qu'une lésion est détectée si sa vraisemblance  $l$  est supérieure à la vraisemblance de toutes les fausses alarmes.

Des expériences montrent que la taille de la fenêtre  $W_s$  est le facteur principal déterminant la taille minimale d'évolution qui peut être détectée. La fenêtre de détection a un effet de lissage : de petites fenêtres permettent de détecter de petites évolutions, mais donnent des résultats bruités (fausses alarmes), de grandes fenêtres peuvent cacher des petites évolutions mais donnent des résultats plus fiables. Une taille de  $3 \times 3 \times 3$  voxels pour la fenêtre paraît offrir un bon compromis, dans notre application de détection d'évolutions de lésions de SEP.

Le système a aussi été évalué avec de grandes lésions simulées pour déterminer la taille maximale d'évolution pouvant être gérée. Nous avons constaté que les évolutions d'une taille inférieure à  $\sigma_r = 7$  étaient correctement gérées. Pour des évolutions plus grandes, des petites

détections parasites peuvent apparaître autour de l'évolution principale. Ceci est essentiellement dû à une sur-compensation lors du filtrage de la correction RF et lors du recalage déformable. Cette taille maximale, de  $\sigma_r = 7$  voxels, correspond à de très grandes évolutions, d'une taille comparable à celle des ventricules. Des tailles d'évolutions aussi importantes, se situent clairement en dehors du cadre de l'application considérée ici (voir § 6.4, p. 110). Nous avons donc, grâce à ces simulations, validé le fait que l'étape de recalage déformable n'affecte pas les évolutions que nous considérons.

## 8.4 Évaluation expérimentale

### 8.4.1 Protocole expérimental

L'évaluation expérimentale proposée ici évalue la *détection d'évolution de lésions* : nous ne cherchons pas à quantifier le *volume* des évolutions des lésions détectées (voir aussi § 8.5). Pour l'expert, la définition exacte de ce qu'est une évolution de lésion est subjective. Elle se fonde sur ses connaissances *a priori* des configurations de lésions. Pour notre système de détection automatique, une « évolution de lésion » est un groupement résultant de l'étape de détection statistique de changements (§ 7.5.1).

Nous avons choisi de considérer des groupements détectés plutôt que des voxels détectés, pour deux raisons. Premièrement, l'expert interprète les résultats en indiquant si une *lésion* a évolué ou non. Il ne considère pas les voxels individuellement. Il serait alors difficile de comparer l'évaluation de l'expert avec celle du système automatique si celui-ci fournissait des résultats en termes de voxels détectés. Deuxièmement, pour l'expert, la détection de changements, voxel par voxel, impliquerait qu'il considère *chaque* voxel individuellement. Ce serait une tâche extrêmement pénible et en pratique irréalisable. Il a été démontré que la délimitation manuelle ou semi-automatique de lésions n'est pas fiable (une forte variabilité inter-opérateur de la charge lésionnelle totale a été, par exemple, observée dans [74]). Il apparaît alors clairement que la délimitation manuelle *d'évolutions* de lésions est encore moins fiable.

Le protocole d'évaluation que nous avons employé se décompose en trois étapes et requiert l'intervention de deux experts :

1. étape 1 : l'expert-1 marque les lésions en évolution sur les paires d'images successives de l'ensemble de la base ;
2. étape 2 : le système automatique marque les lésions en évolution sur les paires d'images successives de l'ensemble de la base ;
3. étape 3 : l'expert-2 évalue les détections de l'expert-1 et celles du système automatique. L'expert-2 décide alors quelles détections sont valides.

La première étape, la détection par l'expert-1, est très longue. L'expert-1 doit analyser chaque paire d'images et détecter les évolutions en comparant visuellement chaque coupe de la paire d'images. Ce procédé est semblable à celui couramment utilisé par un clinicien. Ici, l'expert-1 a été assisté par un recalage rigide et une normalisation d'intensité linéaire. Il a visualisé les images sur un logiciel de visualisation multiplanaire. Remarquez que l'expert-1 n'a pas utilisé les pré-traitements proposés dans notre système de détection automatique (recalage déformable, normalisation par histogrammes conjoints etc.). Ces traitements font partie intégrante du système automatique et notre objectif est précisément de les évaluer. La dernière étape, la validation par l'expert-2, nécessite nettement moins de temps. L'expert-2 doit uniquement parcourir les évolutions déjà détectées et décider lesquelles sont valides. Il n'a pas à parcourir l'ensemble du volume de l'image.

La correspondance entre les évolutions marquées par l'expert-1 et les groupements détectés par le système automatique est effectuée manuellement. Dans quelques très rares cas, une évolution unique détectée par l'expert-1 correspond à plusieurs groupements. Dans ces cas précis, les groupements superflus sont marqués comme étant de fausses détections (faux positifs).

Après l'évaluation par l'expert-2, une évolution détectée par le système automatique peut être identifiée comme étant une :

**-évolution de lésion valide** : confirmée par l'expert-2 ;

**-évolution non lésionnelle valide** : confirmée par l'expert-2 comme étant une évolution physiquement présente, mais n'étant pas une évolution de *lésion* (exemple : expansion ventriculaire) ;

**-fausse détection** : rejetée par l'expert-2.

La figure 8.1 présente l'interface graphique utilisée pour l'évaluation. Au-dessus des animations entre images à comparer sont superposées les détections de l'expert-1 et celle du système automatique. L'expert-2 peut naviguer efficacement entre les différentes détections pour les valider ou les rejeter.

Le seuil  $\lambda$  du système automatique peut être ajusté permettant un compromis entre la sensibilité de la détection et la probabilité d'erreurs. Ce compromis est ajusté pour chaque paire  $(a, b)$  d'images à comparer. La probabilité de détection  $p_{det}(\lambda, (a, b))$  est définie pour chaque paire d'images comme le rapport entre le nombre de détections valides pour le seuil  $\lambda$  et le nombre total de détections valides (y compris celles de l'expert-1). Soit  $n_{err}(\lambda, (a, b))$  le nombre de fausses alarmes pour le seuil  $\lambda$ . La courbe décrit la relation entre  $n_{err}$  et  $p_{det}(n_{err}, (a, b))$  est comparable à une courbe COR et est caractéristique des performances d'un détecteur. La probabilité de détection moyenne  $p_{det}(n_{err})$  sur toutes les paires d'images est illustrée à la figure 8.2. Le résultat final du système de détection est une liste triée des groupements détectés pour chaque paire d'images (décrit au paragraphe 8.5).

## 8.4.2 Résultats expérimentaux

Trois méthodes de détection automatique ont été évaluées : la détection monomodale, la détection multimodale et la détection multimodale employant les résultats de la segmentation corticale. Le protocole d'évaluation complet, exposé précédemment, a été effectué dans le cas des deux premières méthodes. La dernière méthode, employant la segmentation, a été appliquée *a posteriori*, sur les résultats obtenus par la méthode multimodale simple. Sa performance est obtenue en réévaluant celle de la détection multimodale simple. Les performances de ces trois méthodes sont décrites par des courbes de type COR (figure 8.2).

Nous allons d'abord décrire l'intégration des connaissances apportées par la segmentation dans la détection, puis nous comparerons les résultats des trois différentes méthodes avant de décrire les caractéristiques des évolutions et la fiabilité des traitements.

**Intégration des connaissances apportées par la segmentation** Le modèle statistique décrit au paragraphe 7.5 permet de décider si un changement d'intensité a eu lieu dans une fenêtre  $W_s$ . Il détecte tout type de changement d'intensité et n'utilise aucune connaissance spécifique à l'évolution de lésions en sclérose en plaques. La segmentation décrite dans la partie II apporte des connaissances sur les tissus présents dans la fenêtre  $W_s$ , permettant une détection plus spécifique et donc plus performante des évolutions de lésions de SEP.

Nous avons réévalué *a posteriori*, de manière heuristique, les valeurs de log-vraisemblance attribuées à chaque groupement par la méthode de détection multimodale simple. Nous les

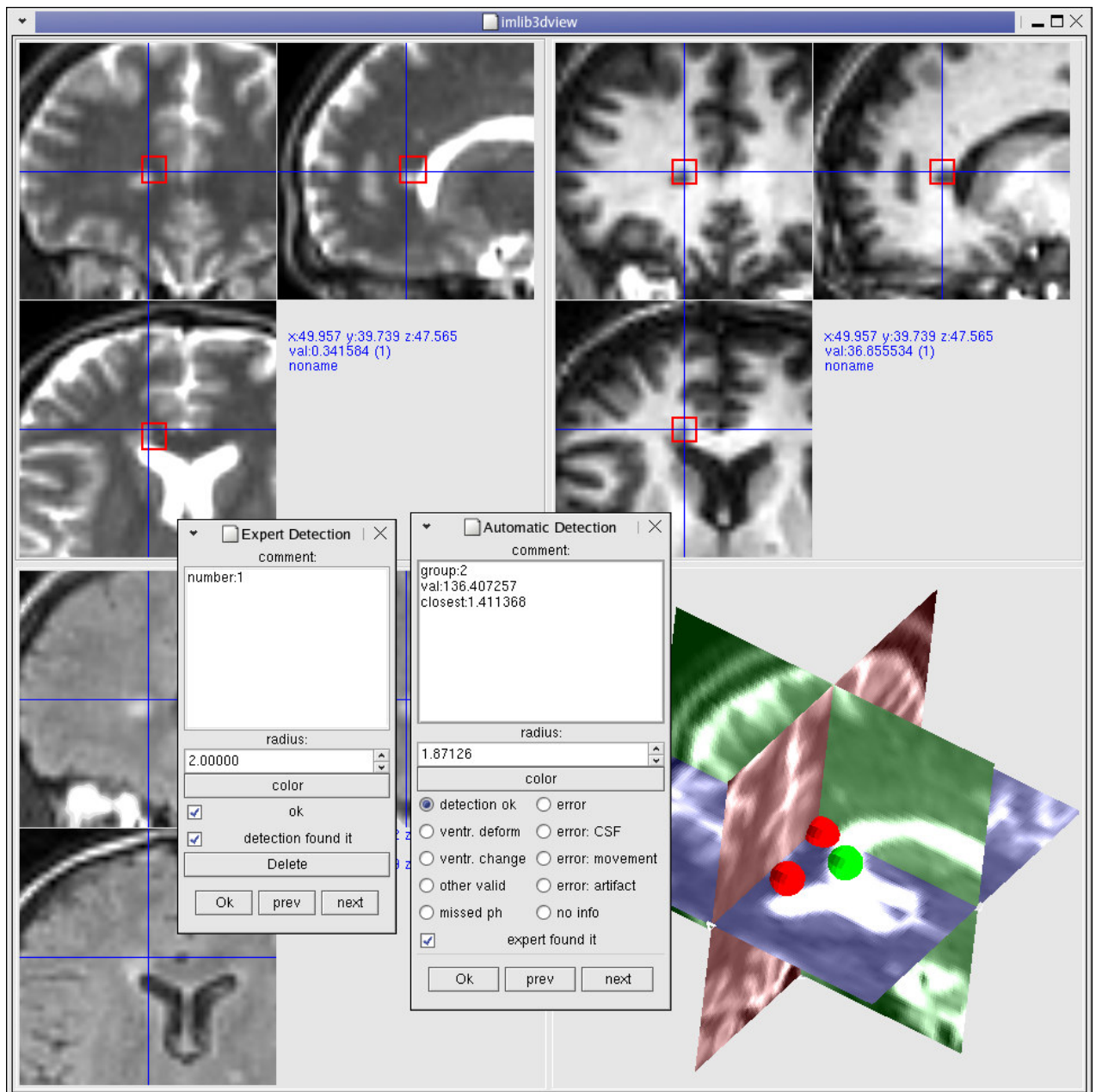


FIG. 8.1 – Interface graphique pour l'évaluation des performances du système de détection. Trois vues multiplanaires affichent des animations alternant entre les deux images à comparer. En haut à gauche les détections de l'expert-1 sont superposées sur l'animation des images à comparer. Sur la vue en haut à droite sont superposées les détections réalisées par le système automatique. Sur celle en bas à gauche est affichée une vue 3D qui aide à se positionner dans l'espace. Cette interface graphique est une extension dynamique (plugin) du visualiseur de ImLib3D (voir § B.5.2, p. 169).

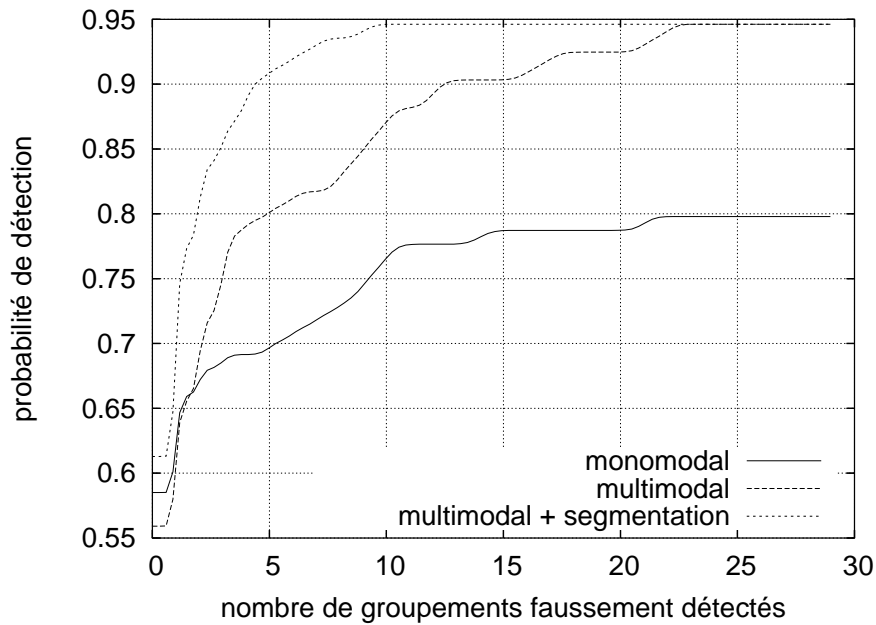


FIG. 8.2 – Performance du système de détection automatique décrite à l'aide de courbes de type COR, indiquant la probabilité de détection en fonction du nombre de fausses alarmes. À titre de comparaison, le taux des détections valides pour l'expert-1 a été évalué à 42%. La multimodalité et la segmentation éliminent des fausses détections, permettant ainsi d'atteindre des taux de détection plus élevés. La segmentation du cortex fournit des informations sur les tissus impliqués dans une évolution permettant de préciser la vraisemblance d'une évolution détectée.

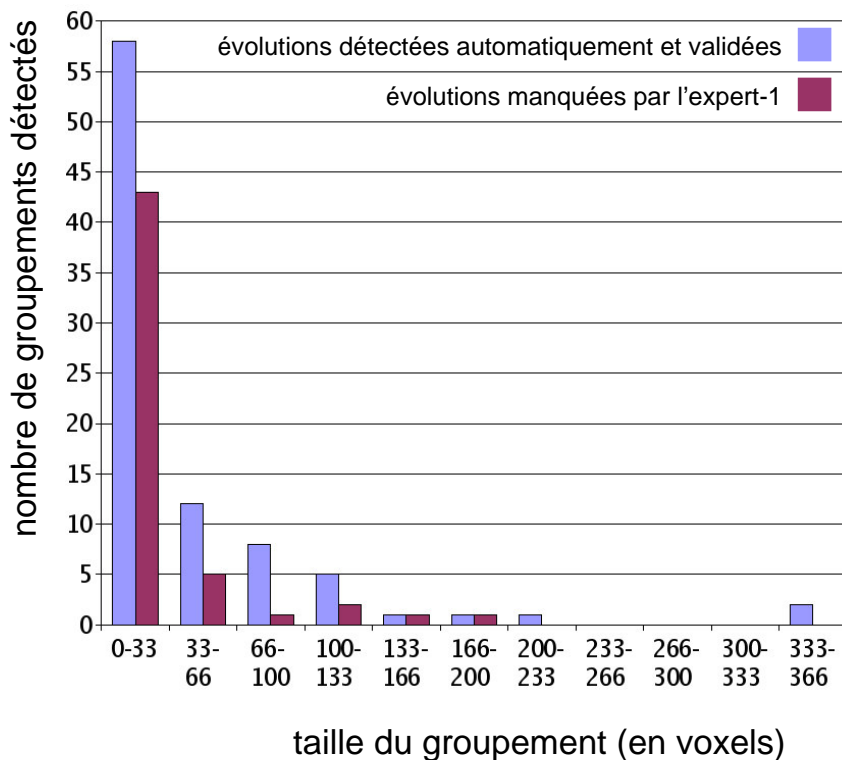


FIG. 8.3 – Tailles des évolutions détectées automatiquement. Chaque voxel fait  $2\text{ mm} \times 2\text{ mm} \times 2\text{ mm}$ . (voir § 8.4.2, p. 136).

avons multipliées par un coefficient  $c$  tenant compte des tissus présents dans la fenêtre  $W_s$  :

$$c = 1 + p_{MB}(1 - p_{LCR}) \quad (8.1)$$

où  $p_{MB}$  et  $p_{LCR}$  sont les proportions moyennes de MB et de LCR sur les segmentations des images  $I_1$  et  $I_2$ . Les ventricules et les régions avoisinantes ne sont pas encore traités par notre approche de segmentation (§ 5.2, p. 99). Nous leur avons donc attribué un coefficient  $c = 2$  en utilisant un atlas présegmenté. Les détections (groupements) évaluées au paragraphe précédent ont été retriées en fonction des nouvelles valeurs de vraisemblance.

**Performance des trois méthodes de détection automatique** La figure 8.2 décrit la performance des trois méthodes de détection automatique. Elle donne la probabilité moyenne de détection en fonction du nombre de groupements faussement détectés. L'expert-1 a détecté les évolutions avec une probabilité de 0.42. En d'autres termes, plus d'une lésion n'a pas été décelée par l'expert-1. 37% des détections de l'expert-1 n'ont pas été validées par l'expert-2, ce qui correspond vraisemblablement à un nombre moyen de fausses alarmes, par paire d'image, faible<sup>1</sup>. Le système automatique a donc été plus performant que l'expert-1 même pour un nombre de fausses alarmes faible. Le système automatique permet aussi d'atteindre des probabilités de détection bien plus grandes (95% ) pour un nombre de fausses alarmes plus élevé (voir figure 8.2).

Les résultats, présentés à la figure 8.2, démontrent aussi que les connaissances apportées par la segmentation, permettent de réduire considérablement le nombre de fausses alarmes. À partir de 10 fausses alarmes, la détection de toutes les évolutions est presque certaine (95%). L'amélioration des performances est due principalement à la diminution de la vraisemblance attribuée aux évolutions se situant près du centre des sillons corticaux. Remarquons que nous avons uniquement retrié les 30 premières détections, sans que l'expert-2 ait procédé à une nouvelle évaluation. Une réévaluation complète permettrait éventuellement d'atteindre un taux de détection supérieur à 95%.

**Caractéristiques des évolutions** Les lésions que nous avons observées étaient de tailles et de formes très variées. Les évolutions de lésion n'étaient pas limitées à des formes sphériques en contraction ou en expansion. Notre système de détection ne fait aucune hypothèse sur ce point. Au total, 93 évolutions de lésions valides ont été détectées, soit par l'expert-1, soit par le système automatique, sur l'ensemble de la base. L'expert-1 a décelé un total de 35 évolutions de lésion valides. La figure 8.3 donne une description des tailles des évolutions valides détectées par le système automatique. Les volumes exprimés représentent une estimation approximative<sup>2</sup> calculée à l'aide d'un seuil unique, réglé manuellement, sur l'ensemble de la base. La figure 8.3 met également en évidence le fait que les évolutions correctement détectées par l'expert-1, ainsi que celles qu'il n'a pas décelées, ont des tailles comparables à celles de l'ensemble des évolutions. Les évolutions manquées par l'expert-1 ne sont ni particulièrement plus petites ni plus grandes que l'ensemble des évolutions. Parmi les 5% d'évolutions ne figurant pas dans

<sup>1</sup>Pour chaque paire d'images les détections du système automatique sont triées suivant leur vraisemblance. Aucun critère semblable ne permet de trier les détections de l'expert-1. Il est donc impossible de tracer une courbe de type COR pour l'expert-1. L'évaluation précise du nombre de fausses détections pour l'expert-1 est impossible à réaliser en toute rigueur. Le point de détection de l'expert-1 est donc situé sur un point inconnu de la droite horizontale  $p_{det} = .42$  de la représentation COR. Cependant, en analysant les caractéristiques des erreurs de l'expert-1, on peut supposer que son taux de faux alarme est faible.

<sup>2</sup>Les problèmes et imprécisions liées à cette opération sont décrits au paragraphe « quantification du volume des évolutions » § 8.5, p. 140 et sont aussi évoqués § 8.4.1, p. 135.



les 30 premières détections du système automatique, la majorité était de très petite taille (volume total entre 2 et 4 voxels). Ces lésions sont d'une taille équivalente à celle d'une lésion simulée, dont le profil gaussien aurait un écart type  $.4 < \sigma_r < .65$ , ce qui est cohérent avec les résultats obtenus sur les lésions simulées. Les évolutions non détectées restantes, manquées par le système automatique, présentaient des variations d'intensité très faibles. L'expert-2 n'était pas en mesure d'affirmer avec certitude s'il s'agissait réellement d'évolutions. Les évolutions identifiées par l'expert-1 mais rejetées par l'expert-2 sont difficiles à caractériser, les divergences ayant des causes très diverses (voir § 8.5, p. 140).

**Fiabilité des traitements** La chaîne entière de traitements a été appliquée à plus de 200 images sources de modalités différentes sans aucune intervention d'un opérateur. La robustesse et la fiabilité des traitements sont deux caractéristiques essentielles d'un tel système automatisé.

## 8.5 Discussion

**Erreurs de détection de l'expert** Comme le prouvent les résultats expérimentaux, un neurologue, non assisté par un système de détection automatique, peut ne pas déceler un nombre important (ici, 58%) d'évolution de lésions valides. Le parcours manuel du volume entier de deux images pour y repérer des changements est une tâche longue et fastidieuse. Le neurologue peut alors parcourir trop rapidement certaines régions et ainsi ne pas remarquer une évolution. Par exemple, certaines lésions proches des sillons corticaux peuvent parfois être confondues avec des portions de MG en  $T_1$ . De plus, l'expert peut être induit en erreur par certains artefacts d'acquisition, comme les changements de contraste, alors que ceux-ci sont corrigés dans le système automatique. Ainsi, sans normalisation non-linéaire, une lésion au profil gaussien peut sembler changer de taille, alors qu'il ne s'agit que d'un changement de contraste.

**Un outil interactif pour le neurologue** Le système présenté ici, fournit un outil pouvant assister le neurologue dans la recherche d'évolutions de lésions. Le système présente au neurologue les 30 premières évolutions détectées automatiquement, triées par ordre décroissant de vraisemblance. Le neurologue peut alors rapidement parcourir ces évolutions et rejeter les fausses détections. Le parcours de ces 30 évolutions détectées nécessite quelques minutes (entre 1 et 3) pour une paire d'images à comparer, alors que la comparaison manuelle de deux volumes requiert jusqu'à 20 mn et même parfois plus.

**Identification des problèmes et visualisation** Un des problèmes essentiels rencontrés lors de la conception de ce système, a été d'identifier précisément quels étaient les artefacts à l'origine des erreurs de détection. Le système visuel humain et les connaissances de l'expert compensent naturellement un grand nombre d'artefacts, rendant par conséquent plus difficile l'identification des causes exactes des erreurs de détection. Des erreurs de positionnement sous-voxel sont particulièrement difficiles à observer. Une inspection détaillée, assistée de plusieurs outils de visualisation, a été nécessaire. Nous avons remarqué que des animations affichant successivement les images recalées représentaient un outil efficace de visualisation (certaines de ces animations peuvent être consultées sur [http://ipb.u-strasbg.fr/ipb/gitim/research/research\\_a.html](http://ipb.u-strasbg.fr/ipb/gitim/research/research_a.html))

**Compromis de la détection** Un système de détection de changements doit à la fois compenser des changements indésirables (artefacts) et conserver les changements réels, authentiques. Ce compromis peut être délicat. Nous avons donc dû considérer l'impact sur les changements réels, de chaque étape de traitement compensant des changements indésirables. La validité de nos choix a été confirmée par l'évaluation portant sur les lésions simulées (§ 8.3, p. 134). Nous allons évoquer ici, cette problématique à propos des étapes de recalage déformable, de normalisation d'intensité et de correction des inhomogénéités RF.

Le recalage déformable multi-échelle que nous utilisons nous permet de choisir l'échelle la plus fine que nous souhaitons recaler. À une échelle  $i$  le champ de déformation est divisé en  $(2)^i$  cellules dans chaque direction. En choisissant  $i = 3$ , nous obtenons un champ de basse résolution (divisée en  $(2^3)^3 = 512$  cellules), aux variations lentes, n'ayant pas d'impact significatif sur des objets de dimension réduite comme les lésions (ceci a été vérifié expérimentalement au paragraphe 8.3). En choisissant une résolution plus basse, certaines déformations indésirables (§ 6.4.1, p. 112) ne seraient pas corrigées. En revanche, une résolution plus élevée réduirait la taille maximale des évolutions acceptables avant l'apparition de détections parasites provoquées par ce recalage déformable.

La normalisation d'intensité porte sur l'ensemble de l'image (histogramme conjoint). Les changements d'intensité locaux, comme des lésions, n'ont pas d'impact significatif sur les histogrammes calculés sur l'ensemble de l'image.

Le filtre passe-bas, corrigeant les inhomogénéités RF, a une réponse impulsionnelle très étendue. Il n'affecte donc pas les changements locaux sur les images.

**Apport de la multimodalité** L'utilisation simultanée de plusieurs modalités à chaque acquisition a grandement amélioré la performance du système de détection, augmentant de 79% à 95% la probabilité de détection (figure 8.2). En effet, la multimodalité fournit une information redondante alors que les artefacts, eux, ne sont pas corrélés entre modalités. Les images RARE multi-coupes ont des artefacts résiduels distribués d'une manière très différente de ceux présents dans les images réellement 3D (GE3D). Cette redondance dans l'information permet l'élimination d'un grand nombre de fausses alarmes, dans la détection multimodale et donc l'augmentation du taux de détection pour un taux de fausse alarme donné (figure 8.2). L'information multimodale aide aussi le neurologue à confirmer ou à rejeter un changement, lors de la détection manuelle.

**Apport de la segmentation** Les connaissances anatomiques apportées par la segmentation ont grandement amélioré la performance du système de détection. Presque toutes les évolutions (95%) sont détectées à partir de dix fausses alarmes (figure 8.2). Le gain est essentiellement dû à l'atténuation de la vraisemblance attribuée à des changements aux centres des plis corticaux. L'intégration des connaissances apportées par la segmentation à l'aide d'un coefficient, permet d'éviter une décision binaire (tout ou rien). Ceci est intéressant pour des évolutions de lésions proches de (voir intérieures à) la matière grise.

**Quantification du volume des évolutions** Le système décrit ici se donne pour objectif de déterminer si des changements, même subtils, sont survenus. Nous avons constaté (conformément à d'autres équipes, voir § 8.4.1) le manque de précision de la quantification d'évolutions. Le bruit et les artefacts résiduels rendent la validation (choix binaire) de petites évolutions très difficile. La quantification du volume de ces changements est encore bien plus délicate. De plus, le profil d'intensité d'une lésion peut changer de manière très complexe, ce qui rend

difficile de donner une définition satisfaisante et fiable de ce qu'on entend par le *volume* d'une évolution. Néanmoins, en acceptant ces imprécisions, notre méthode peut être utilisée très simplement pour une analyse quantitative (voir figure 8.3 et § 8.4.2). Cependant, ces résultats pourront s'avérer en désaccord avec ce que le neurologue considère, subjectivement, comme étant le volume d'une variation.

# Chapitre 9

## Conclusions et perspectives

### 9.1 Bilan et conclusions générales

Avant de conclure ce manuscrit, nous proposons un bilan de nos contributions à la segmentation corticale et à la détection de changements.

#### 9.1.1 Bilan des contributions à la segmentation corticale

L'approche de segmentation décrite dans la partie II, nous semble être la principale contribution en amont de ce travail de thèse. Nous décomposons cette contribution en quatre aspects.

**Représentation et évolution de la segmentation** Notre approche emploie une représentation à support discret (éventuellement sous-voxel) évoluant par pas discrets. Elle nous permet d'intégrer des connaissances *a priori*, complexes, représentées par des contraintes éventuellement paramétrées par un atlas. Ces connaissances peuvent être difficiles à intégrer dans les approches continues ou surfaciques existantes (modèles déformables, level-sets), comme le prouvent les travaux en cours sur ce sujet.

**Modèle d'acquisition et contrainte image** Nous avons proposé une approche multi-échelles d'intégration des informations image dans le processus de segmentation. Celle-ci modélise rigoureusement les effets de volumes partiels. De nombreuses approches de segmentation existantes emploient directement les informations d'intensité sans tenir compte de ces effets, importants dans notre application. Aucune, à notre connaissance, n'intègre une description du processus d'acquisition.

**Contrainte de distribution relative** Nous avons proposé une contrainte décrivant la distribution relative de différents tissus. Elle est particulièrement adaptée à la description de la structure des sillons corticaux. Cette contrainte est plus générale, et plus souple, que des concepts analogues, comme les surfaces couplées, proposées dans la littérature [130].

**Contrainte d'épaisseur** Nous avons proposé une contrainte décrivant l'épaisseur d'un objet. À cette fin, nous avons proposé une définition, originale et utilisable, de l'épaisseur d'un objet en chacun des points de sa surface.

#### 9.1.2 Bilan des contributions à la détection de changements

Nous décomposons les contributions à la détection de changements, appliquée à l'imagerie cérébrale, en six aspects.

**Identification des problèmes gênant la détection** Une première étape de ce travail a été l'identification des artefacts gênant la détection. En particulier, parmi ceux-ci, deux aspects importants n'avaient pas été identifiés lors des travaux antérieurs : l'existence de déformations complexes sur les images et les différences d'intensité non-linéaires. L'identification a été rendue possible grâce à l'observation approfondie de la base d'images de SEP à l'aide d'outils de visualisation développés à cette fin.

**Utilisation du recalage déformable** Une fois identifiées, nous avons corrigé les déformations complexes en utilisant un recalage déformable. La méthode de recalage, développée par notre équipe, nous a permis de compenser sélectivement les déformations de grande échelle. Les travaux antérieurs n'avaient jusqu'alors utilisé que des recalages rigides et affines, insuffisants pour compenser les déformations présentes dans les images.

**Utilisation d'informations multimodales** Les approches existantes de détection de changements d'intensité emploient des images d'une seule modalité. Dans notre approche, l'information multimodale a permis une amélioration sensible des résultats de détection.

**Utilisation d'un cadre probabiliste** Nous avons proposé l'adaptation d'une approche de détection statistique existante (GLRT [45, 53]), dans ce cadre particulier. Les approches existantes de détection d'évolutions de lésions basées sur l'intensité, comparent quant à elles les images par soustraction (à l'exception de Lemieux *et al.* [62]). Notre approche a aussi permis l'intégration d'informations multimodales. L'estimation d'une matrice de covariance du bruit permet de pondérer automatiquement chaque modalité en fonction de sa qualité (variance du bruit).

**Normalisation non-linéaire d'intensité** Nous avons proposé une approche corrigeant les variations non-linéaires d'intensité entre images. Nous avons démontré que la normalisation linéaire utilisée par les approches existantes est nettement insuffisante dans certains cas.

**Utilisation de la segmentation du cortex** L'information apportée par la segmentation des plis corticaux a permis une amélioration sensible des résultats de la détection. A notre connaissance, aucune approche de détection de changements n'a encore utilisé ses informations.

### 9.1.3 Conclusion

Nous avons présenté un système complet de détection de changements. Notre démarche a été menée avec l'objectif concret d'aboutir à un système fonctionnel performant. Elle a consisté à identifier les problèmes des méthodes existantes et à proposer des outils permettant de les résoudre. Nous avons employé ou adapté des outils existants lorsque cela s'est avéré possible (recalage déformable, modèle statistique, ...) et lorsque cela ne l'était pas, de nouveaux outils théoriques (segmentation, normalisation d'intensité, ...) ont été proposés. Nous avons choisi d'approfondir certains thèmes en fonction de l'objectif applicatif. Certains développements théoriques pouvaient sembler intéressants en soi (recalage déformable, modèle statistique, modélisation du bruit, ...); cependant, nous avons jugé qu'ils n'auraient pas apporté d'améliorations significatives à la performance du système de détection si d'autres problèmes, plus sérieux, n'étaient pas résolus auparavant. L'exception est la segmentation. Après avoir résolu les problèmes de détection les plus graves, nous avons constaté que l'amélioration de la performance de détection nécessitait de compenser des artefacts d'acquisition dont l'apparence peut être spécifique à l'appareil IRM employé. Nous avons alors choisi une autre voie,

consistant à apporter des connaissances supplémentaires. La multimodalité était la première étape et a prouvé son intérêt. La segmentation des sillons corticaux a été la deuxième étape. Les systèmes existants de segmentation ne sont pas en mesure, à l'heure actuelle, de fournir une segmentation détaillée des sillons corticaux sur des images basse résolution ( $2mm^3$ ). En développant notre système, nous avons constaté qu'il portait des éléments prometteurs et avons décidé d'approfondir ce domaine bien au-delà de ce qui était strictement nécessaire à l'application à la détection. Nous pensons que ce choix se justifie *a posteriori* par les résultats et les potentiels du système de segmentation.

## 9.2 Perspectives

**Description des évolutions recherchées** Une voie importante à explorer dans l'amélioration de la détection est l'apport de connaissances spécifiques aux évolutions que l'on souhaite observer. Le coeur du système que nous avons décrit ici, est un système général de détection de changements d'intensité. Une première étape dans la construction d'un modèle des évolutions est l'apport des connaissances par la segmentation du cortex. Celle-ci a permis de spécifier dans quels tissus il était possible d'observer des évolutions de lésions de SEP. D'autres apports de connaissances sont envisageables. La modélisation des formes des lésions par des approches d'apprentissage est une voie possible. Le modèle doit pouvoir représenter des tailles et des formes de lésions d'une grande variabilité (pour la SEP [62]). Cette approche exige une importante base de lésions segmentées dans des contrastes et des résolutions proches de celles utilisées lors de l'application.

**Fusion de données** L'apport de connaissances, évoqué au paragraphe précédent, nécessite d'intégrer des informations provenant de sources différentes dans un système de décision unique. Le test statistique présenté au paragraphe 7.5, p. 129 pourrait être étendu dans ce sens. Ceci permettrait, entre autres exemples, de fusionner des informations provenant de la segmentation et des informations de changement d'intensité. Cette opération est actuellement menée de manière heuristique (§ 8.4.2, p. 136).

**Modèles statistiques du bruit** Le modèle utilisé actuellement pour le bruit dans l'image est simple (bruit gaussien de variance constante) mais robuste (§ 7.5, p. 129). Un modèle plus sophistiqué doit pouvoir représenter des artefacts d'aspect très complexe pouvant apparaître sur les images. En particulier, les artefacts de mouvement (§ 2.1.5, p. 20) paraissent difficiles à modéliser. Cependant, ces derniers sont généralement localisés dans des régions prédictibles (à hauteur des yeux, par exemple). Une première solution simple serait alors de moduler la variance du bruit dans ces régions, à l'aide d'un atlas de variance [62]. Une autre solution serait un apprentissage du bruit par des techniques de réduction de dimension, comme l'analyse par composantes principales (ACP). Les deux solutions proposées précédemment exigent un apprentissage sur un ensemble d'images de patients sains. Le résultat de cet apprentissage sera spécifique à la séquence et à l'appareil employé. Ceci limiterait son utilisation pratique.

**Quantification du volume des évolutions** Comme cela a été précisé au paragraphe précédent (8.5), la quantification d'évolutions de lésions est délicate. Cependant, si l'on considère des images de résolution plus élevée, ou si on se limite à des évolutions de taille importante, la quantification peut être envisageable. La détection de changements, proposée ici, peut alors permettre d'identifier les évolutions à quantifier, la quantification se déroulant dans

une deuxième étape. Rey *et al.* [93] proposent, par exemple, de segmenter des lésions en évolution à l'aide de modèles déformables. Dans leur approche, les évolutions sont identifiées en analysant le champ de déformation interimages. D'autres approches sont envisageables. Il peut être intéressant de modéliser le profil d'intensité des lésions en évolution considérées. Dans un exemple très simple, les lésions seraient approchées par des profils d'intensité gaussiens, pour être ensuite comparées.

# Quatrième partie

## Annexes





# Annexe A

## Réflexions sur le développement logiciel et la recherche

Les réflexions présentées ici ne font pas partie intégrante de la contribution scientifique de ce travail de thèse. Néanmoins, il nous paraît important de s'interroger sur le fonctionnement de la discipline à laquelle nous participons et de proposer des initiatives pour contribuer à son évolution. C'est dans ce cadre que nous proposons à la communauté de traitements d'images médicales un système logiciel complet `ImLib3D`, librement disponible. Avant de décrire `ImLib3D` au paragraphe B.1, p. 156 nous allons présenter, ici, les réflexions qui ont motivé ce travail.

Le traitement d'images est une discipline scientifique comprenant à la fois des aspects théoriques, des aspects expérimentaux et des applications pratiques. L'expérimentation et l'application reposent, en traitement d'images, sur des systèmes logiciels dont la complexité croît avec l'avancement de la discipline. Cette complexité pose des défis fondamentaux.

Après une description des menaces que cette complexité fait peser sur le traitement d'images, nous argumenterons sur la nécessité d'une ouverture et d'une large diffusion des développements logiciels associés à la recherche.

### A.1 Reproductibilité et démarche scientifique

Le système des publications, avec évaluation par des pairs, est, aujourd'hui, un des principaux garants de la validité de la démarche scientifique. Mais ce système, bien établi, est aussi utilisé par les institutions (universités, centres de recherche, organismes de financement) pour juger le travail des chercheurs. Publier devient alors un enjeu déterminant pour l'attribution de ressources et pour les carrières de la recherche. La forte concurrence induite, si elle peut être stimulante dans certains cas, suscite une surenchère dans les publications. Les résultats publiés sont alors souvent exposés de manière très favorable, et les problèmes occultés. Dans ce cadre, le travail d'évaluation devient particulièrement délicat, et repose principalement sur deux axes : l'analyse critique et la reproductibilité des résultats.

#### Évaluation par analyse critique

Par analyse critique, nous entendons le processus d'évaluation qui consiste à évaluer un article sans expérimentation. Si cette approche permet de repérer des erreurs de méthodologie, elle reste fortement subjective. Le passage d'un cadre théorique à un cadre expérimental soulève très souvent de nombreux problèmes, imprévisibles, subtils, mais rédhibitoires. Il est très

difficile de prévoir comment une méthode peut réagir à diverse formes de perturbations dans les données. Il n'est pas aisé d'envisager tous les cas particuliers pouvant dérouter une méthode. Sauf pour des articles purement théoriques (comme la démonstration d'un théorème), il paraît donc peu fiable de baser l'évaluation sur la seule analyse critique.

### Évaluation par reproduction expérimentale

La reproductibilité des résultats est un des fondements de la méthodologie scientifique. Dans de nombreuses disciplines, cela implique l'utilisation de matériel onéreux et de procédures lourdes, elle peut même y faire l'objet d'une publication en soi. En traitement d'images, la reproduction expérimentale nécessite la copie du système logiciel, ce qui n'a qu'un coût négligeable. Malheureusement, le logiciel implémentant une méthode décrite dans une publication n'est que rarement librement disponible. La reproduction des résultats nécessite donc, le plus souvent, de ré-implémenter les méthodes décrites. Elle se heurte alors sur plusieurs obstacles majeurs :

**Ré-implémentation : un effort insurmontable** L'obstacle le plus important à la ré-implémentation d'un logiciel est son coût, en terme de quantité de travail. Un système de traitement d'images moderne peut être constitué de plusieurs dizaines de milliers de lignes de code. En intégrant les divers outils nécessaires, il peut même s'agir de plus d'un million de lignes. Ceci peut représenter plusieurs mois de travail et, en pratique, aucun relecteur n'est prêt à fournir un pareil effort.

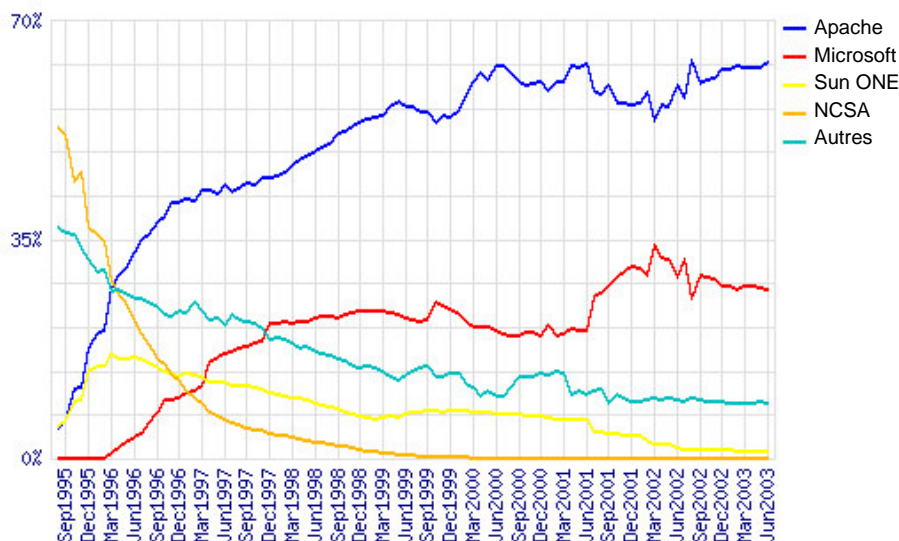
**Ré-implémentation : manque de détails** Une publication, même si elle décrit de manière détaillée une méthode, ne peut pas en être une description complète. L'expérience montre qu'un grand nombre de petits détails doivent être résolus pour en assurer le fonctionnement effectif. Ces détails, réunis, forment une partie essentielle d'un travail, mais n'ont pas leur place dans le cadre synthétique et l'espace limité d'une publication.

**Le jeu de données/images** Même si le relecteur réussit à obtenir ou à ré-implémenter le logiciel associé à un article, il lui manque encore les données nécessaires pour l'évaluer et surtout pour reproduire les résultats publiés. En imagerie médicale, par exemple, la création d'une base de donnée d'images d'évaluation (comme celle utilisée dans le travail de cette thèse, voir § 8.1, p. 134) représente un travail considérable et un investissement financier non négligeable. Un autre facteur limitant la reproductibilité des résultats est l'utilisation pratique des méthodes. Souvent, les méthodes requièrent le réglage d'un nombre considérable de paramètres, difficiles à déterminer.

### Conclusion partielle

Les considérations précédentes tendent à montrer que dans son état actuel, la discipline de traitement d'images peut être ouverte à des dérives inquiétantes, et que la validité des publications y est sujette à caution. Une illustration en est la disparité entre la simplicité des techniques utilisées dans des applications réelles et l'évolution spectaculaire de la complexité des techniques apparaissant dans les publications.

Face à ces dérives, une solution serait de considérer l'implémentation logicielle comme partie indissociable d'une publication et du travail scientifique. Le paradigme du logiciel libre



TAB. A.1 – Les logiciels libres ont pris une position dominante dans l’infrastructure logicielle d’Internet. Par exemple, à partir de 1996, Apache devient le premier serveur Web (source : netcraft.com)

offre un cadre adapté, dynamique et éprouvé, présenté au paragraphe A.2. Les freins à son extension sont analysés au paragraphe suivant (A.3).

Si la diffusion libre des logiciels associés à un travail paraît nécessaire, elle n’est pas suffisante. Le partage des données ainsi que l’organisation de cadres d’évaluation standardisés pour des méthodes ayant les mêmes objectifs, sont d’autres aspects essentiels.

## A.2 Logiciels libres

« Logiciel libre » est un terme générique pour désigner des logiciels dont le code source est redistribué librement, avec divers types de licences destinées à empêcher toute ré-appropriation susceptible de les priver de leur liberté<sup>1</sup>. Les logiciels libres ne sont plus une simple curiosité, ils sont devenus un des principaux acteurs de l’informatique professionnelle actuelle. Des logiciels libres tels BIND (DNS,  $\simeq$  100% de parts de marché), Apache (serveur Web, voir tableau A.1) et SendMail (serveur mail,  $\simeq$  80%) forment les briques de base de l’infrastructure logicielle de l’Internet. Des systèmes d’exploitation libres, tels GNU-Linux, gagnent en importance<sup>2</sup> et commencent même à s’introduire dans le marché des ordinateurs personnels. Le moteur de recherche Google, par exemple, repose sur un cluster de plus de 10 000 serveurs sous Linux<sup>3</sup>. Divers pays (Brésil, Pérou, Chine ...) ont adopté, ou sont en voie d’adopter, des mesures législatives incitant ou imposant aux institutions, et parfois aux entreprises d’Etat l’utilisation de logiciels libres<sup>4</sup> [66].

L’essor des logiciels libres a intrigué de nombreux observateurs. Les motivations économiques et sociologiques des divers acteurs ont fait l’objet de nombreuses études (références dans [60]). Les mécanismes sociologiques (économie de la réputation, capital symbolique) en œuvre dans les communautés de logiciel libres [56, 91] présentent de fortes ressemblances avec

<sup>1</sup><http://www.gnu.org/philosophy/free-sw.fr.html>

<sup>2</sup>En 2002, Linux possédait 49% des parts sur le marché des serveurs contre 25% pour MS Windows [43].

<sup>3</sup><http://www.google.fr/press/highlights.html>

<sup>4</sup><http://news.com.com/2100-1001-272299.html?legacy=cnet>

ceux agissant dans le champ de la recherche scientifique [54].

### Accessibilité et flexibilité

Contrairement à une idée reçue, la gratuité des logiciels libres n'est pas leur principal attrait. De nombreux utilisateurs professionnels, particulièrement dans l'industrie, et souvent dans la recherche, sont prêts à payer le prix de logiciels tels que MATLAB.

Les avantages majeurs des logiciels libres sont leur accessibilité et leur flexibilité. Pour évaluer ou utiliser un logiciel libre, il suffit de le télécharger et de l'installer. Lorsqu'un chercheur ou un ingénieur a ponctuellement besoin d'un outil, il n'a pas besoin de demander une autorisation, de faire un bon de commande, et d'attendre la livraison. Une fois le logiciel libre téléchargé, il peut aisément l'intégrer en partie ou en totalité au sein de ses propres développements. De plus, il peut le décortiquer, l'analyser et s'en inspirer. Ces aspects sont très importants. Il est fréquent, en informatique, d'apprendre par l'exemple, cela n'étant possible qu'avec le code source à disposition. Par ailleurs, si le logiciel libre ne répond pas exactement aux besoins de l'ingénieur, des modifications / corrections, souvent très simples, peuvent aisément être apportées au logiciel afin de mieux l'adapter. Par exemple, un logiciel propriétaire est disponible uniquement sous forme binaire, on ne peut donc l'installer que sur les plateformes (systèmes d'exploitation) prévues par le vendeur. Dans le cas du libre, il est possible de les recompiler, quitte à y faire de légères modifications.

### Qualité logicielle

Contrairement à une autre idée reçue, les logiciels libres sont souvent à la fois plus fiables et plus performants que leurs équivalents non libres. Les études quantitatives précises se multiplient dans ce sens<sup>5</sup>, certaines comparant, à l'aide de protocoles expérimentaux rigoureux, les taux de pannes survenant sur des configurations équivalentes, dans des environnements réels.

L'univers du logiciel libre est en foisonnement permanent. De nouveaux projets se créent puis meurent en permanence. Les logiciels que nous évoquons ici sont ceux ayant atteint une certaine maturité, par la persévérance de leurs développeurs, et la sélection sévère imposée par les utilisateurs.

Une des explications de la qualité des logiciels libres est à chercher dans le fonctionnement dynamique des communautés « Open Source ». Les utilisateurs, communiquant par e-mail ou au travers de forums ouverts, signalent les défaillances et participent à leur réparation [91, 123]. L'enthousiasme généré par l'environnement coopératif ouvert est un facteur important de cette dynamique.

Les développeurs d'un produit libre sont engagés dans une concurrence basée sur la réputation [54], et accordent une grande importance à la fiabilité de leur produit. Leur code source étant ouvert et visible par tous, il est exposé à la critique (tout comme une publication scientifique est évaluée par des pairs). Le développeur peut donc difficilement se permettre de produire du code de mauvaise qualité. La situation est toute autre dans certains environnements non libres, où la pression commerciale peut pousser à la production rapide de code de mauvaise qualité.

Dans le cas particulier de la recherche, la publication de leurs développements en « Open Source » incite les chercheurs à faire attention à la qualité et à la documentation de leur production logicielle. Si cela demande un effort supplémentaire, le résultat est une plus grande

---

<sup>5</sup>[http://www.dwheeler.com/oss\\_fs\\_why.html](http://www.dwheeler.com/oss_fs_why.html)

ré-utilisabilité. Au lieu de produire du code expérimental « jetable », le chercheur pose alors des briques sur lesquelles d'autres pourront construire.

### Éthique

Un autre argument majeur en faveur du logiciel libre est l'éthique. Au lieu de développer pour leur bénéfice propre ou celui de l'institution dans laquelle ils travaillent, les producteurs de logiciels libres diffusent au plus grand nombre le fruit de leur travail<sup>6</sup>. Tout comme le chercheur peut être motivé par l'avancement de la science, la large diffusion d'un logiciel libre participe au progrès technique. Les chercheurs du secteur public sont rémunérés par le travail du contribuable, qui attend d'eux la plus grande efficacité dans l'avancement de la recherche. Restreindre la diffusion de leur production est un frein important à l'avancement de la recherche, ce qui est contraire à la mission qui leur est implicitement confiée.

Par ailleurs, l'accroissement des inégalités au niveau mondial, maintient une majorité de la population à l'écart du progrès technique. Les institutions de recherche des pays pauvres n'ont pas les moyens de payer les licences souvent coûteuses des logiciels de la recherche. Partager la production logicielle scientifique contribue à moins les marginaliser sur le plan scientifique, et leur permet d'accéder plus facilement aux résultats de la recherche en termes d'applications.

## A.3 Freins à l'extension des logiciels libres dans la recherche

Si une grande partie des logiciels libres trouvent leur origine dans le milieu universitaire ou de la recherche, il reste que de nombreux laboratoires sont réticents à la diffusion libre de leur production logicielle.

Les équipes de recherche sont souvent en compétition pour les mêmes ressources limitées (financements, publications), et ne souhaitent alors pas diffuser leur travail de peur de favoriser les concurrents. Mais inversement, la diffusion libre d'un logiciel performant peut aussi contribuer fortement au rayonnement du laboratoire.

Les restrictions budgétaires et les incitations de la part des pouvoirs publics, poussent les laboratoires à vouloir générer des ressources propres en commercialisant leur production logicielle. Cependant, il n'est pas certain qu'un groupe de recherche dispose des moyens humains<sup>7</sup> nécessaires à la réussite commerciale d'un logiciel. La diffusion de la production logicielle, dans un cadre libre, demande aussi un effort non négligeable, même s'il est bien moindre que dans le cadre commercial. Il s'agit de mettre en forme le code source, de le documenter, de le maintenir et d'interagir avec les utilisateurs. Comme il a été dit précédemment, une partie de ce travail est très utile pour le laboratoire lui-même, en ce qu'il permet de pérenniser les développements.

---

<sup>6</sup>Ceci n'est pas toujours entièrement désintéressé : le capital symbolique accumulé en participant au monde du libre peut souvent être utilisé pour obtenir des carrières bien rémunérées dans le secteur marchand.

<sup>7</sup>Ingénieurs, commerciaux et service après vente.



# Annexe B

## ImLib3D

ImLib3D est une plate-forme de traitement d'images volumiques composée d'une librairie et d'un logiciel de visualisation interactif séparé. Elle est distribuée librement (<http://imlib3d.sourceforge.net>), en « Open Source », et est placée dans un cadre de développement distribué coopératif (sourceforge.net). Elle a été créée et est gérée par l'auteur, avec des contributions de T.Vik. ImLib3D a été conçue pour la recherche et le développement en traitement d'images volumiques (3D, et 3D+t). Elle vise à être facile à utiliser par des scientifiques n'ayant pas nécessairement une connaissance pointue en programmation<sup>1</sup>. En revanche, elle ne vise pas à fournir un outil adapté au clinicien<sup>2</sup>. L'ensemble du travail de cette thèse a été implémenté à l'aide de ImLib3D.

ImLib3D repose sur une conception orientée objet et utilise des concepts modernes s'inspirant de la librairie standard du C++. Ces fondements conceptuels seront brièvement présentés aux paragraphes suivants, avant d'aborder la présentation technique de ImLib3D.

Dans la communauté de traitement d'images cérébrales, plusieurs initiatives de logiciels librement disponibles sont à noter. Statistical Parametric Mapping<sup>3</sup> est un logiciel basé sur Matlab<sup>®4</sup>, très répandu, en particulier dans l'imagerie fonctionnelle. Matlab est un environnement propriétaire proposant un langage de programmation interprété, orienté vers le prototypage rapide d'applications scientifiques. Il est moins adapté au développement d'applications complexes, en termes de performances et de conception logicielle, qu'un langage de programmation moderne, comme le C++. Une autre initiative, très récente est ITK<sup>5</sup>. La première version, développée par des industriels sous contrat du NLM<sup>6</sup>, est apparue en 2002. Elle est fondée sur des concepts similaires à ceux de ImLib3D (généricité, orientée objet, ... voir ci-après). Il s'agit d'un projet de taille conséquente, conçu avec une méthodologie rigoureuse. Le financement de ITK prend fin en 2003 et la gestion de cette initiative, dans l'avenir, paraît incertaine. Si cette initiative est prolongée et si elle reste « Open Source », elle pourrait prendre une ampleur considérable dans le milieu du traitement d'image médicales. Dans le domaine

---

<sup>1</sup>L'utilisation exige néanmoins la connaissance de la syntaxe du C++. Il a été choisi dans ImLib3D d'utiliser un C++ conforme au standard. La modification du C++ (par l'utilisation de macros, ou d'un préprocesseur) est une possibilité intéressante permettant de simplifier d'avantage la syntaxe[19], mais exige aussi l'apprentissage d'un « nouveau » langage de programmation.

<sup>2</sup>Un outil destiné au clinicien pourrait être conçu comme une couche logicielle s'appuyant sur ImLib3D. Les besoins, les concepts, et la terminologie utilisés par les cliniciens sont très éloignés de ceux utilisés dans la communauté de traitement d'images. Un outil commun ne paraît donc pas satisfaisant.

<sup>3</sup>SPM[30] : <http://www.fil.ion.ucl.ac.uk/spm>

<sup>4</sup><http://mathworks.com>

<sup>5</sup>Insight Toolkit : <http://itk.org>

<sup>6</sup>National Library of Medicine du National Institutes of Health



non médical, remarquons Vigna<sup>7</sup> (traitement 2D) et Olena<sup>8</sup> [19] (traitement N-D), ayant des approches génériques comparables à la nôtre.

## B.1 Fondements conceptuels

La conception de ImLib3D repose sur trois fondements : la généricité, les itérateurs et la conception orientée objet. Ces concepts sont brièvement présentés dans les paragraphes suivants. Le concept de l'extensibilité et son implémentation dans ImLib3D seront décrits plus loin (§ B.5, p. 169). D'autres concepts de génie logiciel employés dans ImLib3D, tels que les batteries de tests automatisées, ne seront pas décrits ici.

### B.1.1 Généricité

**Motivations** Le stockage et la manipulation de données sont des aspects fondamentaux de la programmation. Le programmeur structure ses données dans des tableaux (1D, 2D . . .), des listes chaînées, des tableaux associatifs, etc., selon ses exigences en termes de performances et de place mémoire. La gestion de ces structures de données, appelées conteneurs, a longtemps été laissée à la charge du programmeur (bibliothèques *ad hoc* dans certains langages compilés)<sup>9</sup>. Le programmeur avait alors le choix entre des conteneurs spécifiquement typés (en C/C++ : un tableau *d'entiers*, une liste chaînée *de flottants*), ou des conteneurs non-typés (un tableau de pointeurs *void*, une liste chaînée contenant des pointeurs *void*). Ces deux solutions ne sont pas satisfaisantes. La première implique la réécriture des conteneurs pour chaque nouveau type de données à stocker. Dans le cas de la seconde solution (les conteneurs non-typés) le trans-typage doit être effectué explicitement par l'utilisateur et il n'y a, en général, pas de vérification de type<sup>10</sup>, ce qui est extrêmement dangereux. De surcroît, cette solution laisse à l'utilisateur la gestion de la mémoire (allocation, désallocation des données stockées), ce qui, en pratique, est très fastidieux et peu fiable. Les problèmes que nous venons d'illustrer pour les conteneurs, sont plus généraux, ils existent aussi pour les traitements. D'une manière plus générale, la dépendance à des types (ou des constantes) fixes pose d'importants problèmes pour la conception d'un logiciel.

**Description** Dans la programmation générique, les classes, structures ou fonctions génériques sont paramétrées par des types<sup>11</sup> qui ne seront précisés qu'à l'utilisation. Par exemple, le concepteur crée une classe représentant une liste chaînée dont le type de données reste indéfini. Ensuite, l'utilisateur instancie cette classe générique pour créer une liste chaînée de flottants (par exemple). Le concept de la généricité<sup>12</sup>, et son introduction dans le C++ par

---

<sup>7</sup><http://kogs-www.informatik.uni-hamburg.de/~koethe/vigna>

<sup>8</sup><http://www.lrde.epita.fr/olena>

<sup>9</sup>Pour les langages non compilés, des vérifications de type sont possibles à l'exécution, et les remarques qui suivent ne s'appliquent pas.

<sup>10</sup>La vérification de type peut être très coûteuse à la fois en mémoire et en temps de calcul.

<sup>11</sup>La généricité peut porter sur des types, mais aussi sur des valeurs constantes. Par exemple, si on définit un vecteur de dimension fixe, la valeur constante de cette dimension peut être un argument template.

<sup>12</sup>La généricité ne doit pas être confondue avec le polymorphisme. Dans la conception objet, le polymorphisme permet effectivement de manipuler de manière homogène des objets d'une classe dérivant d'un même parent. La généricité permet, elle, de manipuler des objets de familles totalement distinctes, sans avoir à faire de transtypage (up-cast, down-cast), et sans avoir à créer des hiérarchies artificielles (cf. conteneurs en Java). Les deux concepts sont distincts mais complémentaires.

les templates et STL<sup>13</sup>, ont marqué une évolution importante dans la conception de la programmation. STL fournit une librairie de classes génériques avec, entre autres, un cadre unifié, extrêmement bien conçu, de gestion des différentes formes de conteneurs (tableaux, listes chaînées, tableaux associatifs, etc.), et les moyens de parcourir ces conteneurs (itérateurs, voir paragraphe suivant). Par ailleurs, la programmation générique a montré de nombreux potentiels<sup>14</sup>, comme en témoigne le foisonnement d'applications dans boost<sup>15</sup>. Si la conception et l'implémentation de librairies génériques restent des tâches complexes<sup>16</sup>, leur utilisation, par le programmeur est très simple et transparente.

**Généricité dans ImLib3D** Une image volumique à support discret est naturellement interprétée comme un conteneur à support tridimensionnel. En traitement d'images il est courant de manipuler des images pouvant avoir des valeurs réelles, de valeurs complexes (domaine de Fourier), de valeurs entières (images de label), de valeurs binaires, etc. Dans ImLib3D, ces images sont implémentées dans la même philosophie que les conteneurs génériques de STL. Les fonctions (processeurs) de traitement d'image sont aussi génériques, et peuvent donc être réutilisées pour plusieurs types d'images différentes.

## B.1.2 Itérateurs

**Itérateurs : Introduction** Lors de sa conception, un conteneur peut être vu sous deux angles : le stockage des données (évoqué précédemment), et le parcours des données. Dans la conception classique, chaque conteneur était parcouru par une approche spécifique. Par exemple, un tableau était parcouru en balayant ses indices  $i$  et en indexant le tableau (`tab[i]`), une liste chaînée était parcourue, elle, en étudiant successivement le nœud courant puis en passant au nœud suivant. Le concept d'itérateur permet d'unifier le parcours des différents types de conteneurs avec une même syntaxe<sup>17</sup>. Un itérateur est un objet indiquant la position courante dans un conteneur. Il permet d'accéder au contenu à cette position (déréférencement) et fournit les méthodes permettant de modifier sa position (avancer, reculer, avancer de plusieurs éléments, etc.). Chaque conteneur, et ses itérateurs associés, ont des capacités et une complexité algorithmique spécifiques associées à chaque opération. Par exemple, avancer un itérateur de  $n$  éléments a une complexité algorithmique de  $O(1)$  pour un tableau, de  $O(n)$

<sup>13</sup>La Standard Template Library (STL), est une librairie de conteneurs génériques initialement proposée par A. Stepanov en 1992 et qui a été incorporée au standard ANSI/ISO définissant le C++ en 1994. Voir [105, 79].

<sup>14</sup>À titre anecdotique, on peut démontrer que l'exécution par le compilateur de l'instanciation récursive de templates est équivalente à une machine de Turing. Il est alors théoriquement possible de faire exécuter par le compilateur un programme arbitraire. Ceci montre les potentiels mais aussi les dangers de l'utilisation de templates. Il peut devenir tentant, dans certaines situations, d'utiliser les templates dans des contextes pour lesquels il n'ont pas été prévus. Ceci peut aboutir à du code excessivement complexe, ou bien à une modification potentiellement déroutante de la syntaxe du C++.

<sup>15</sup>Boost <http://www.boost.org> est un répertoire de librairies C++, soumises à une révision par pairs stricte. Ces librairies ont pour vocation d'être intégrées dans des versions futures du standard définissant le langage C++.

<sup>16</sup>Il s'agit de concevoir du code qui puisse fonctionner pour un grand nombre de types différents, ce qui exige beaucoup de soin.

<sup>17</sup>En plus de permettre une syntaxe unifiée, le concept d'itérateur permet également de dissocier les algorithmes des conteneurs. Par exemple, une fonction qui trie (quicksort) une suite de données pourra travailler uniquement sur les itérateurs, sans se soucier du type de conteneur associé. Cette approche n'est cependant pas utilisée dans ImLib3D, car la plupart des processeurs de traitement d'image nécessitent des informations sur le conteneur (l'image) qui ne sont pas présentes dans des itérateurs.

pour une liste chaînée et de  $O(\log_2(n))$  pour un tableau associatif (implémenté en STL sous forme d'un arbre binaire).

**Syntaxe en C++/STL** Un exemple simple d'itérateur est le pointeur en C/C++. Considérons un pointeur  $p$  sur un élément d'un tableau. L'accès au contenu à la position indiquée par  $p$  se fait par le dérérérencement  $*p$ . Le passage à l'élément suivant se fait par l'incréméntation  $p++$ . Cette syntaxe est reprise et généralisée dans STL pour les itérateurs sur toutes les formes de conteneurs (tableaux, listes chaînées, etc.). La généralisation de cette syntaxe est rendue possible grâce à la redéfinition des opérateurs du C++.

### B.1.3 Conception orientée objet

**Motivations** Le génie logiciel, dont la conception orientée objet est un des aspects importants, est né d'un constat pratique. La gestion de la complexité d'un logiciel est le principal enjeu du développement logiciel. En d'autres termes, s'il est facile d'écrire un programme implémentant une fonctionnalité précise, il est très difficile d'écrire un système logiciel de plus grande taille qui puisse évoluer. Un petit programme ( $< 3\ 000$  lignes) peut être appréhendé dans son ensemble par un individu, et aucune méthodologie particulière n'est nécessaire à sa conception<sup>18</sup>. Lorsque la taille et le nombre de développeurs croissent, on atteint rapidement un point de blocage. Le temps alors passé à comprendre, reconcevoir, et réimplémenter, dépasse de loin le temps passé à ajouter de nouvelles fonctionnalités. Une méthodologie devient donc indispensable.

**Méthodologie objet** La méthodologie objet repose, entre autres, sur le concept de modularité. Un logiciel est vu comme un ensemble de boîtes noires (objets) communiquant entre elles. La communication entre boîtes noires doit être réduite à son strict minimum, et se fait au travers d'interfaces bien définies et documentées (figure B.1). Néanmoins, l'approche orientée objet n'est pas, en soi, une garantie de bonne conception. Des hiérarchies de classes trop complexes, et trop imbriquées (over-engineering), sont des erreurs fréquentes. La reconception et la réécriture font donc toujours partie du cycle de vie d'un logiciel. Une bonne conception permet d'allonger la durée de vie avant reconception, et de diminuer la quantité de code devant être réécrit à chaque cycle.

## B.2 La librairie ImLib3D

ImLib3D est composée d'une librairie et d'un visualiseur décrit au paragraphe B.3, p. 167. Deux autres éléments, les extensions utilisateur (§ B.5, p. 169) et l'interface ligne de commande (§ B.4, p. 169), seront aussi décrits ultérieurement (voir figure B.2).

La librairie, décrite ici, fournit tous les composants nécessaires pour écrire des programmes de traitement d'images volumiques. Le **premier** composant est formé des classes d'images et toute l'infrastructure logicielle qui facilite leur gestion. Nous décrirons d'abord les fonctionnalités communes à toutes les images (§ B.2.1), puis les différentes classes d'images proposées (§ B.2.2), et finalement les itérateurs facilitant le parcours des images (§ B.2.3). Le **deuxième** composant est le système de gestion des processeurs de traitement d'images qui fournit une

---

<sup>18</sup>La difficulté d'enseigner la programmation orientée objet provient de là : les étudiants ont rarement eu à gérer des logiciels dépassant quelques centaines de lignes.

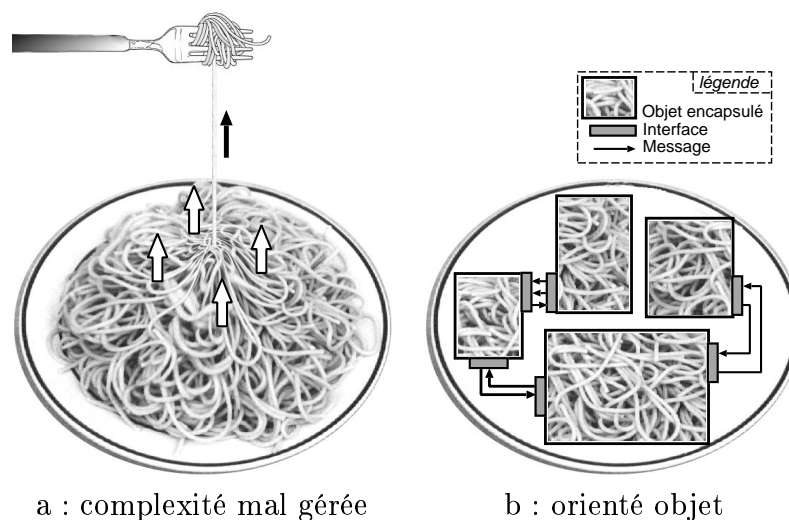


FIG. B.1 – Métaphore du plat de spaghetti. (a) Un logiciel peut rapidement devenir très complexe (comme un plat de spaghetti collant). Lorsqu'on cherche à modifier un élément (retirer un spaghetti) tout est si imbriqué, que cela a des répercussions sur l'ensemble du logiciel. Dans la conception orientée objet (b) on encapsule la complexité dans des boîtes noires communiquant par des interfaces bien définies. Les répercussions d'une modification sont alors mieux contrôlées.

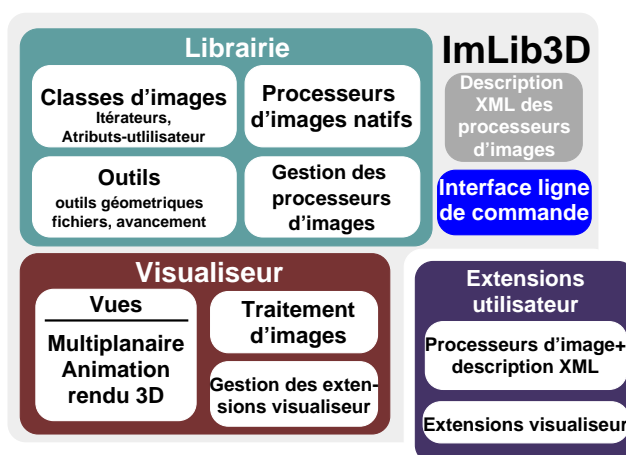


FIG. B.2 – Vue d'ensemble des différents éléments de ImLib3D

interface unifiée et documentée à toutes les opérations de traitement d'images. Il sera décrit au paragraphe B.2.4. Le **troisième** composant, les processeurs de traitement d'images natifs fournis avec ImLib3D, seront décrits au paragraphe B.2.5. Nous finirons la description (§ B.2.6) par le **quatrième** composant de la librairie, constitué d'un ensemble d'outils divers mis à la disposition des utilisateurs de ImLib3D.

### B.2.1 Les fonctionnalités communes à toutes les images

ImLib3D propose un grand nombre de classes d'images partageant certaines fonctionnalités communes. Une partie de ces fonctionnalités, comme la gestion de la taille de l'image, les accès fichier, les masques et les propriétés utilisateur est décrite par la classe `Image3D`. Toutes les images dans ImLib3D dériveront (§ B.2.2) de cette classe (non générique).

**Manipulation élémentaire des images** La création d'images et l'accès à leurs éléments se fait très simplement à l'aide d'opérateurs intuitifs. Voici un exemple :

```
// création d'une image à valeurs flottantes de taille 100x100x100
Image3Df monImage(100,100,100);
monImage(5,2,3)=100; // change la valeur au point (5,2,3)
cout << monImage(5,2,3); // affiche la valeur au point (5,2,3)
```

**Fichiers** Les formats de fichiers d'images médicales volumiques présents dans le commerce (DICOM, ANALYZE [AVW]) ont un codage trop rigide pour permettre l'enregistrement de tous les types différents d'images pouvant apparaître dans ImLib3D (et les informations associées). Un format de fichier spécifique basé sur le XML est donc proposé. Le XML est un langage de description extensible, hiérarchique, et très souple qui permet d'ajouter simplement des informations structurées. ImLib3D permet également à l'utilisateur, s'il le souhaite, de lire et d'écrire des fichiers au format AVW.

```
Image3Df monImage("fichier.im3D");
monImage.WriteToFile("fichier2.im3D");
Image3Df monImage2;
monImage2.ReadFromFile("fichier2.im3D");
```

**Masques** Il est fréquent en traitement d'images de vouloir associer des régions géométriques à une image. On voudra, par exemple, définir des régions d'intérêt (ROI) sur lesquelles pourront se restreindre certains traitements. Dans ImLib3D, toute image est optionnellement associée à un masque. Le masque `Mask3D` est une image à valeurs entières (0-255).

**Propriétés dynamiques utilisateur** Souvent, un utilisateur souhaite associer à une image des informations telles que des données sur un patient ou des préférences d'affichage. La solution de l'héritage proposée par l'approche objet est parfois insuffisante<sup>19</sup>, ou trop lourde à mettre en œuvre. La classe `Image3D` permet à l'utilisateur d'ajouter dynamiquement des attributs nommés de n'importe quel type à une image. Elle fournit un mécanisme pour gérer ces propriétés automatiquement (entrée/sorties fichier, opérateurs de recopie etc.) et vérifie la cohérence des types :

<sup>19</sup> Lorsque des propriétés doivent être ajoutées dynamiquement, par exemple à l'exécution dans une interface graphique, l'héritage n'est pas envisageable.

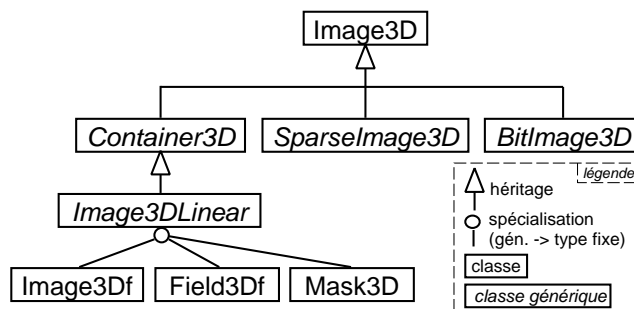


FIG. B.3 – Hiérarchie simplifiée des principales classes d'images.

```

Image3Df image(100,100,100);
image.SetProperty("nom", "Toto");
image.SetProperty("age", 35);
image.SetProperty("position tumeur", Vect3Df(53,25,55));
cout << image.Property<Vect3Df>("position tumeur");

```

**Interpolation** Dans de nombreuses situations, particulièrement lors de transformations géométriques (rotation, transformation affine ...), on souhaite prendre la valeur d'une image sur des points qui ne sont pas situés exactement sur la grille discrète. Les interpolateurs implémentés sont de type plus proches voisins, linéaire, sinus cardinal [61] et B-spline. Ce dernier est particulièrement performant [107] (voir aussi § 7.3.5, p. 122). Les images et les interpolateurs étant génériques, il est possible d'associer un interpolateur à toute image supportant les opérations linéaires (par exemple, des images à valeurs flottantes, des images à valeurs complexes, des champs de vecteurs, ou tout autre type défini par l'utilisateur). Voici un exemple pour un champ de vecteurs :

```

Field3Df champ(10,10,10);
...
champ.SetInterpolator(new SplineInterpolator3D<Vect3Df> >(3));
cout << champ.Value(1.1, 2.2, 3.3);

```

## B.2.2 Les différentes classes d'images

ImLib3D utilise des conteneurs génériques, semblables à ceux de STL, pour représenter des images de types différents. Nous allons commencer leur description par une vue d'ensemble de la hiérarchie des principales classes (figure B.3).

De cette classe parent `Image3D` dérivent trois sous-classes. La plus utilisée, `Container3D`, décrit des images classiques, où les données sont stockées dans un tableau tridimensionnel d'éléments<sup>20</sup>. `Container3D` est une classe générique. Les deux autres sous-classes, les images creuses `SparseImage3D`, et les images compactes `BitImage3D`, ont des modes de stockage des données différents, peu encombrants en mémoire.

### Images classiques

Les images classiques dérivées de `Container3D` peuvent être instanciées pour des types de données très divers. Des exemples courants sont les images à valeurs flottantes `Image3Df` et les

<sup>20</sup>Après expérimentation, le stockage des données image en une seule région contiguë de mémoire est apparue optimale.

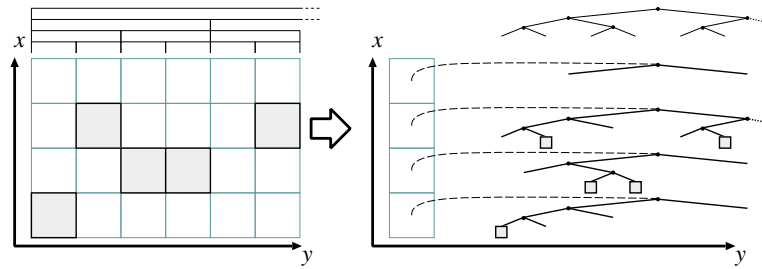


FIG. B.4 – Image creuse (en 2D pour l'illustration). Seuls les éléments non nuls de l'image sont stockés. Un tableau d'arbres binaires (conteneur associatif `std::map` dans *STL*) permet un accès à un point aléatoire en un temps  $O(\log_2(n))$  où  $n$  est la taille de l'image dans une direction (ici  $y$ ). En 3D il s'agit d'un tableau bidimensionnel d'arbres binaires. L'accès séquentiel aux éléments non nuls de l'image se fait en un temps linéaire.

images à valeurs entières `Mask3D`<sup>21</sup>. Parmi les images classiques, nous pouvons distinguer celles supportant des opérations linéaires, images flottantes, images complexes (`Image3Dcomplex`), champ de vecteurs (`Field3Df`), etc. D'autres types d'images sont simples à dériver de `Container3D`<sup>22</sup>. L'utilisateur peut aussi très facilement spécialiser `Container3D` pour ses propres besoins. Par exemple, une image, où chaque élément serait un modèle statistique (moyenne, variance) pourrait être utilisé comme suit :

```
struct ModeleStatistique {float moyenne,ecartType;};
...
Container3D<ModeleStatistique> atlas(50,50,50);
atlas(5,2,3).moyenne=10;
atlas(5,2,3).ecartType=2;
```

## Images creuses

Les images classiques peuvent devenir très volumineuses. Dans certaines applications, comme dans la segmentation proposée dans la partie II, seul un nombre limité de points sont non nuls. Il est alors possible de réduire fortement l'encombrement mémoire en stockant uniquement les valeurs non nulles des images. Les images creuses `SparseImage3D` dans `ImLib3D` ont une structure creuse (disposition d'éléments nuls et non nuls) fixe, qui peut être partagée par plusieurs images creuses. Chaque image creuse ne contient plus (pour simplifier) qu'un tableau des valeurs non nulles. L'accès séquentiel aux éléments non nuls est donc très rapide. La structure creuse (figure B.4) permet aussi l'accès à une position arbitraire de l'image. Pour un point où la valeur de l'image est nulle, l'accès se fait en un temps constant  $O(1)$ ; pour les points non nuls l'accès se fait en un temps logarithmique. Ces opérations sont transparentes pour l'utilisateur, qui manipule l'image de la même manière que les images ordinaires. La classe `SparseImage3D` est générique permettant la création d'images creuses ayant des éléments de tout type.

```
Mask3D elementsNonNuls("elementsNonNuls.im3D");
SparseStructure3D structureCreuse=SparseStructure3D::Create(elementsNonNuls);
```

<sup>21</sup> Les valeurs de `Mask3D`, sur un octet, sont limitées entre 0 à 255.

<sup>22</sup> Par exemple, dans `ImLib3D` il existe une image représentant une subdivision régulière de l'espace utile pour résoudre le problème des  $k$  plus proches voisins `Neighbors3D`. Chaque élément de cette image est un tableau redimensionnable (`std::vector`) contenant des points tridimensionnels.

```

SparseImage3D<float> imageCreuse1(structureCreuse);
SparseImage3D<int > imageCreuse2(structureCreuse);

```

### Images compactes

Les images génériques classiques permettent de créer des images de tout type. Cependant, la taille minimale d'un élément est un octet. Lorsque l'utilisateur souhaite manipuler des images à valeurs entières ayant un nombre restreint de valeurs (images binaires par exemple), il est possible d'avoir une représentation plus compacte. La classe `BitImage3D` permet de stoker chaque voxel sur un nombre arbitraire de bits (identique pour toute l'image).

```

// création d'une image dont les valeurs sont représentées sur 2 bits
// les groupes de 2 bits sont arrangés sur des octets
BitImage3D<byte,2,byte> image(50,50,50);
image(5,2,3)=2;

```

### B.2.3 Itérateurs : le parcours des images

En traitement d'images, une opération essentielle est le parcours des images. Les itérateurs fournissent un cadre simple et standardisé dans ce but. Ils allègent le code, améliorent sa lisibilité et peuvent permettre des gains de performance. Au lieu d'écrire une triple boucle imbriquée pour parcourir l'image, une seule boucle suffit :

```

Image3Df image(100,100,100);

// parcours d'une image par la méthode classique
for(int z=0;z<image.Depth() ;++z)
  for(int y=0;y<image.Height();++y)
    for(int x=0;x<image.Width() ;++x)
      image(x,y,z)=rand();

// parcours d'une image à l'aide d'un itérateur
for(Image3Df::iterator p=image.begin();p!=image.end();++p)
  *p=rand();

```

Comme l'illustre cet exemple, l'utilisation des itérateurs permet aussi d'encapsuler (cacher les détails de l'implémentation) le fonctionnement de l'image. Dans la méthode classique, l'utilisateur doit avoir des connaissances sur des détails du fonctionnement interne de la classe afin de savoir dans quel ordre il est plus efficace d'écrire les trois boucles<sup>23</sup> (x, y, z ou z, y, x).

ImLib3D fournit différents types d'itérateurs pour parcourir les images de diverses manières (figure B.5). Il existe deux versions de l'itérateur permettant le parcours lexicographique de toute l'image. L'une `iteratorFast` est très rapide (pas d'indice à gérer). L'autre `iteratorXYZ`, légèrement plus lente, permet de suivre la position (x, y, z) sur laquelle se trouve l'itérateur. D'autres itérateurs permettent de parcourir des portions restreintes d'une image (régions parallélépipédiques, ou définies par un masque), ou de parcourir l'image dans un ordre spécial (cubes concentriques).

Les itérateurs de ImLib3D sont compatibles avec STL. Il est donc possible de les utiliser avec les algorithmes STL standards (`sort`, `find`, `for_each`, `mismatch`...).

<sup>23</sup>L'ordre des boucles a un impact considérable sur la performance dû à l'utilisation de la mémoire cache.



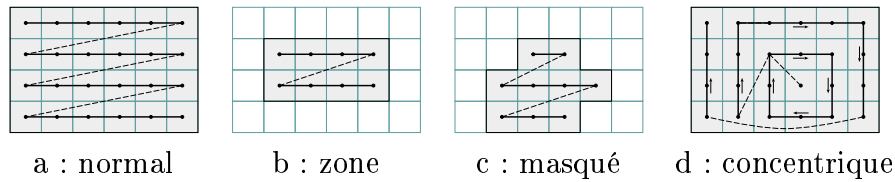


FIG. B.5 – Trois types d’itérateurs parcourant des régions différentes d’une image (en 2D pour l’illustration). (a) Itérateur parcourant l’ensemble de l’image dans l’ordre lexicographique. (b) Itérateur parcourant une portion parallélépipédique de l’image. (c) Itérateur parcourant une région définie par le masque de l’image. (d) Itérateur parcourant l’image par cubes concentriques successifs, à partir d’un point.

## B.2.4 Gestion des processeurs de traitement d’images

Nous avons décrit, dans les paragraphes précédents, les différentes opérations de base permettant de manipuler des images dans ImLib3D. Ces opérations peuvent être composées pour implémenter des algorithmes de traitement d’image. Il s’agit alors de gérer le nombre important d’algorithmes différents qui peuvent exister dans une boîte à outils de traitement d’images. Cette gestion permettra de manipuler, et d’étendre, ces processeurs au travers de diverses interfaces.

Nous définissons le concept de « processeur de traitement d’image » (ou pour raccourcir « processeur d’image ») pour désigner l’implémentation d’un algorithme. Concrètement, un processeur d’image est composé d’une fonction C++ (habitant le namespace IP3D) et d’une description complète de cette fonction<sup>24</sup>. La description, structurée dans un répertoire hiérarchique XML, fournit des informations détaillées sur tous les aspects de la fonction : son nom, sa généricité, une documentation courte, une documentation plus longue, une description complète de chaque argument (nom, type, valeur par défaut, ...). Cette description est utilisée dans différents contextes (figure B.6). Les processeurs d’images respectent des conventions de syntaxe strictes, garantissant l’uniformité de leur utilisation. Ils pourront être appelés dans différents contextes (visualiseur, ligne de commande, programme C++). Le mécanisme précédent permet à l’utilisateur d’ajouter très simplement ses propres processeurs d’images (voir § B.5).

## B.2.5 Processeurs natifs

ImLib3D fournit un nombre important de processeurs de traitement d’image. Ils sont énumérés au tableau B.1, leur description détaillée dépasse le cadre de cette introduction. Certains processeurs (`ConvenienceProcessors`) correspondent à des manipulations qui se font très simplement en C++, mais qui sont aussi nécessaires dans les autres interfaces (graphique, ligne de commande).

<sup>24</sup>Nous avons expérimenté diverses approches pour la gestion des processeurs d’images, dont certaines utilisant des hiérarchies de classes pour représenter les processeurs. Finalement, il apparaît que ces approches ajoutent beaucoup de complexité et ne résolvent pas les problèmes de manière satisfaisante. Notre conclusion est qu’un processeur d’image est finalement un concept très large. Il est l’équivalent conceptuel d’une fonction. Il est cependant apparu qu’il était important de disposer d’une description permettant à d’autres modules de gérer ces fonctions convenablement.

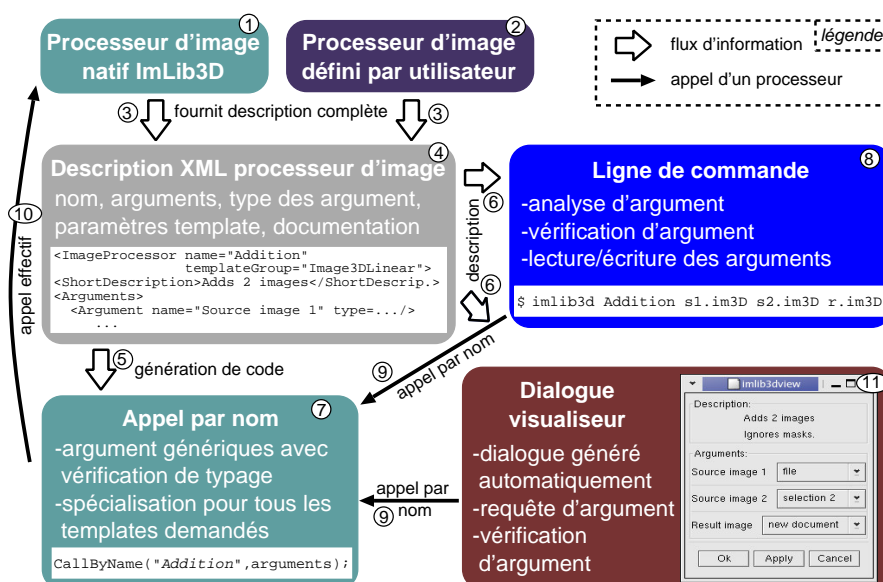


FIG. B.6 – Chaque processeur de traitement d'image, qu'il soit natif (1), ou une extension de l'utilisateur (2), est décrit (3) de manière détaillée. La description (4), en XML, fournit toutes les informations nécessaires pour permettre de générer automatiquement (5) le code nécessaire pour appeler le processeur. Ces informations sont aussi suffisantes pour générer automatiquement (6) des dialogues interactifs du visualiseur et une interface ligne de commande. L'appel d'un processeur par son nom, en fournissant une liste d'arguments génériques (7), nécessite d'avoir préalablement spécialisé (5) tous les processeurs génériques pour tous les types souhaités par le concepteur du processeur. Par exemple, lorsque l'utilisateur demande, sur la ligne de commande (8) d'additionner deux images complexes ( $\mathbb{N}^3 \rightarrow \mathbb{C}$ ), il est nécessaire que le processeur *Addition* ait été préalablement compilé (5,7) pour le type complexe (la compilation à la volée poserait trop de problèmes). L'appel en ligne de commande (8) utilise (6) la description XML du processeur pour gérer les arguments fournis par l'utilisateur et vérifier leur cohérence. Les arguments saisis sont ensuite mis dans une liste générique, et le processeur d'image est appelé par son nom (9). Le processeur correspondant, avec la bonne spécialisation est ensuite effectivement appelé (10). Le même schéma est utilisé pour l'appel d'un processeur à partir d'un dialogue interactif du visualiseur (11).

<b>Arithmetic</b>	TransformAffine	WriteToFileIPB*
Addition	Scale	ReadTransfoFromFileIPB*
AdditionWithConstant	Flip	ImageFromRawData
Difference	SwapAxes	RawDataFromImage
DifferenceWithConstant	WrapTranslate	WriteToFileAVW
Multiplication		ReadFromFileAVW
MultiplicationWithConstant	<b>LinearFiltering</b>	ReadFromFileBruker
Division	Convolution	ExportImageSlice
MaxImage	SeparableConvolution	
Abs	ConvolutionFFT	<b>TestPatterns</b>
	FFTLowPassFilterApodizedIdeal	Sphere
<b>ImageStats</b>	FFTLowPassFilterButterworth	Cone
Average	FFTLowPassFilterChebyscheff	NoiseUniform
AverageAndVariance	PartialSum	NoiseGaussian
RobustAverageAndVariance	CumulativeSum	Explode
Median	BoxFilter	Target
	GaussianApproxFilter	Bump
<b>MorphologicalOperators</b>	SymmetricBinomialFilter	Ramp
MedianFilter		Parallelogram
SharpeningFilter	<b>ImageAs1DVector</b>	RectangularGrid
Erosion	ScalarProduct	ColoredGrid
Dilation		
Opening	<b>ShapeAnalysis</b>	<b>Normalization</b>
Closing	CenterOfGravity	NormalizeAverageAndVariance
DistanceTransform	PrincipalAxes	NormalizeJointHistogram
ConnectedComponentLabelling	BoundingBox	ComputeJointHistogramImage
FillHoles		
MakeValueList	<b>VisualizationAids</b>	<b>FFTOperators</b>
LargestConnectedComponent	TransformRectangularGrid	FFTDouble
<b>VoxelCoding</b>	AnimateField	FFTFloat
VoxelCode		FFTInverseDouble
SSCode	<b>ConvenienceProcessors</b>	FFTInverseFloat
BSCode	ImageTypeConversion	FFTDoubleToComplex
ShortestPathExtraction	Crop	FFTFloatToComplex
Skeleton	Pad	FFTInverseComplexToDouble
	InsertImage	FFTInverseComplexToFloat
<b>Estimator</b>	ResizeImageSupport	
NoiseVarianceEstimation	FindFirstDifference	<b>BrainProcessors*</b>
	FindMinMax	ExtractHeadMask*
<b>Threshold</b>	SetAllVoxels	SegmentationFromAtlas*
SimpleThreshold	AddMask	SegmentHeadAndBrain*
SimpleThresholds2	GetMask	CenterCenterOfGravity*
LimitThreshold	ApplyMask	ChangeDetectionConstantPlusNoise*
OtsuThresholds	SetDefaultInterpolator	
UniModalThreshold	SetProperty	<b>Registration*</b>
	ExtractFromComplexImage	RigidRegistration*
<b>Transform</b>		AffineRegistration*
TransformBSpline3D*	<b>FileConversion</b>	DeformableRegistration*
TransformWithInverseField	ReadFromFileIPB*	

TAB. B.1 – *Processeurs de traitement d'image existants regroupés hiérarchiquement. Les processeurs marqués d'un astérisque sont des extensions utilisateur. Les autres processeurs sont natifs à ImLib3D.*

### B.2.6 Outils

**Surveillance de l'avancement d'une tâche** Le traitement de certaines images tridimensionnelles peut être très long. L'utilisateur s'inquiète alors, et souhaite connaître l'état d'avancement du traitement, et le temps restant. `ImLib3D` fournit une approche de surveillance à base de fils<sup>25</sup> très simple à utiliser et n'ayant pas d'impact sur la performance du programme<sup>26</sup>. L'utilisateur doit simplement indiquer une variable qu'il souhaite surveiller. Généralement il s'agira d'un itérateur :

```
Image3Df::iterator p;
ImageProgress avancement("état d'avancement:",p);
for(p=image.begin();p!=image.end();++p)
{
    ...
}
```

Cet exemple imprimera à l'écran le message « état d'avancement : xx% », toutes les secondes (valeur paramétré), en fonction de la position de l'itérateur dans l'image.

**Outils divers** `ImLib3D` fournit une série de fonctions utilitaires pour la manipulation des chaînes de caractères (utilisant `std::string`), pour la manipulation de fichiers et pour l'affichage de courbes.

**Outils géométriques** `ImLib3D` fournit aussi un ensemble de classes facilitant les manipulations géométriques : des vecteurs bidimensionnels et tridimensionnels, une classe pour les transformations affines, des régions parallélépipédiques et quelques autres outils simples.

## B.3 Le visualiseur

`ImLib3D` propose un visualiseur extensible permettant d'afficher, de traiter et d'éditer des images volumiques. Les images sont représentées dans une classe `Document` ayant des spécialisations génériques permettant de manipuler un grand nombre de types d'images différentes.

L'espace d'affichage est composé d'un nombre variable de vues (`View`). Trois grands types de vues sont actuellement implémentés. Les vues multiplanaires, les animations multiplanaires et le rendu 3D surfacique. L'utilisateur peut lier certaines vues entre elles (avec différents adaptateurs, si les tailles des images sont différentes). La navigation dans une vue sera alors synchronisée avec certaines autres vues.

### B.3.1 Vues

**Vue multiplanaire** L'affichage multiplanaire permet de naviguer dans un volume tridimensionnel en visualisant trois coupes perpendiculaires en un point désigné par le curseur. Elles sont re-dimensionables, et l'utilisateur peut facilement zoomer sur le point désigné. Lorsque le type de l'image le permet (type linéaire), les vues multiplanaires permettent d'afficher les images en utilisant l'interpolation linéaire, améliorant ainsi nettement la qualité de l'affichage.

<sup>25</sup>Les fils (en anglais : threads) sont des processus légers qui s'exécutent parallèlement, et qui partagent l'espace d'adressage mémoire du processus principal.

<sup>26</sup>L'approche classique exige d'ajouter des tests à l'intérieur d'une boucle potentiellement critique.

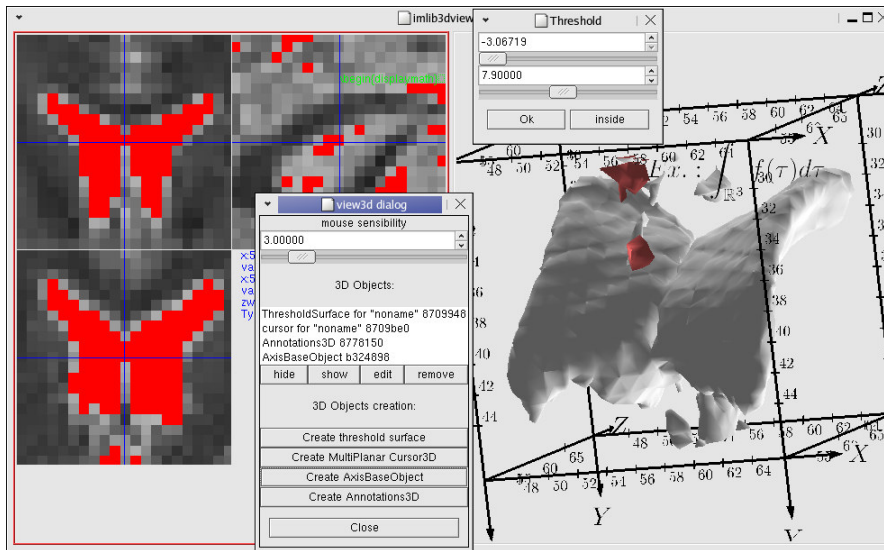


FIG. B.7 – À gauche une vue multiplanaire agrandie des ventricules. À droite une vue 3D du volume seuillé (rouge) dans la vue multiplanaire.

Parmi de nombreuses fonctionnalités signalons l’affichage avec des palettes de couleurs extensibles, le seuillage interactif, le réglage de contraste, l’affichage par superposition du masque associé, l’affichage d’histogrammes, l’affichage de coupes 1D de l’image et la sélection interactive de régions parallélépipédiques.

**Animations multiplanaires** L’affichage d’animations à partir d’une série d’images est particulièrement utile pour l’analyse de séries temporelles et pour visualiser des changements entre deux images. L’animation se présente sous la forme d’une vue multiplanaire où défilent les images successives. La plupart des opérations des vues multiplanaires décrites au paragraphe précédent sont accessibles.

**Vue 3D** Cette vue fournit une visualisation surfacique en 3D isométrique simple. Elle ne prétend pas être un système de visualisation 3D complet. Elle permet d’afficher des surfaces correspondant à des images seuillées interactivement dans des vues multiplanaires<sup>27</sup>. Elle permet aussi d’afficher en 3D des annotations (y compris du texte avec de formules en  $\text{\LaTeX}$ ), une représentation du curseur correspondant aux trois plans de coupe d’une vue multiplanaire et des axes gradués.

### B.3.2 Autres fonctionnalités

**Traitement d’image** Tous les processeurs de traitement d’images sont accessibles à partir du visualiseur. Cela comprend les processeurs définis dans ImLib3D et ceux définis comme des extensions par l’utilisateur. Un dialogue complet, comprenant une documentation et une saisie interactive des arguments, est généré automatiquement pour chaque processeur.

**Annotations** Un système souple d’annotations permet d’ajouter des commentaires et des dessins (texte, sphères de couleur ...) à toute position de l’espace. Ces annotations pourront

<sup>27</sup>Nous utilisons optionnellement la librairie `gts` (GNU Triangulated Surface Library : <http://gts.sourceforge.net>) pour la simplification et la manipulation de maillages.

être enregistrées avec l'image.

**Dessin** Un système simple de dessin interactif volumique est proposé dans les vues multi-planaires. L'utilisateur peut choisir une plume de taille, d'aspect et d'intensité variables et peut défaire (undo multiple) ses modifications. Cet outil permet de segmenter manuellement des structures anatomiques.

## B.4 Interface ligne de commande

ImLib3D permet l'appel de tous les processeurs de traitement d'image à partir de la ligne de commande. Cette interface est utile pour expérimenter, pour effectuer des opérations simples, pour enchaîner des séquences simples d'opérations (scripts) et pour interagir avec d'autres logiciels.

```
| imlib3d Addition image1.m3D image2.im3D resultat.im3D  
| imlib3d ExportImageSlice resultat.im3D 2 50 resultat.ppm
```

Une liste de tous les processeurs est obtenue par `imlib3d --list`. La documentation d'un processeur est affichée par `imlib3d NomDuProcesseur`.

## B.5 Extensibilité d'ImLib3D

Pour maintenir la cohérence et la propreté d'un système logiciel, il est important que ses utilisateurs puissent aisément ajouter des fonctionnalités sans avoir à modifier le logiciel lui-même. Ces fonctionnalités peuvent répondre aux besoins particuliers de l'utilisateur. Il peut aussi s'agir des fonctionnalités expérimentales ayant pour vocation d'être intégrées ultérieurement dans le logiciel lorsqu'elles seront validées et stabilisées. Dans ImLib3D deux types d'extensions sont possibles. L'ajout de nouveaux processeurs de traitement d'image (§ B.5.1) reposant sur une documentation XML détaillée des fonctionnalités, et l'extension du système de visualisation.

### B.5.1 Extensions des processeurs de traitement d'image

Pour ajouter un processeur de traitement d'image, l'utilisateur écrit les fonctions correspondantes ainsi que la description XML associée. Un utilitaire de ImLib3D permet alors de générer un programme exécutable<sup>28</sup>, qui sera placé dans un chemin déterminé par une variable d'environnement. À l'exécution, les différentes interfaces de ImLib3D reconnaîtront automatiquement les nouveaux processeurs.

### B.5.2 Extensions du visualiseur

Le travail de recherche en traitement d'images est grandement facilité par des outils de visualisation spécifiques au problème considéré. L'ajout dans le visualiseur de ImLib3D de systèmes de visualisation avec des applications très spécifiques n'est pas souhaitable. La conception orientée objet du visualiseur permet à l'utilisateur de créer de nouvelles classes de visualisation (vues) répondant à ses besoins spécifiques, en les dérivant des classes existantes.

---

<sup>28</sup>Il est prévu de permettre aussi l'extension par librairie dynamique, plus performante, mais moins portable.

Le visualiseur permet d'ajouter ces vues, au moment de l'exécution, en s'appuyant sur des bibliothèques dynamiques (DLL). Les extensions du visualiseur ont été utilisées dans ce travail de thèse dans la partie segmentation (§ 5.1.1, p. 90, et figure 5.1.1) et dans la partie détection (§ 8.4, p. 135, et figure 8.1).

## B.6 Perspectives et conclusions

Dans cette partie, nous avons décrit un cadre logiciel reposant sur des fondements conceptuels solides. Ce logiciel est proposé librement à la communauté de traitement d'images volumiques qui lui a montré son intérêt<sup>29</sup>. L'ensemble du travail décrit dans cette thèse, ainsi que d'autres travaux dans notre laboratoire reposent sur ImLib3D. Les principaux concepts de ImLib3D ont donc été modélés et validés par la pratique. La bibliothèque a atteint un niveau de maturité permettant de passer à une étape de plus ample diffusion, et à une interaction accrue avec d'autres initiatives semblables dans le domaine. Les bases paraissent établies pour la recherche de nouveaux contributeurs sur le modèle proposé par E. Raymond [91].

Plusieurs perspectives s'offrent alors. La construction progressive d'une bibliothèque comprenant les principaux algorithmes de traitement d'images médicales utilisés par la communauté. Ceci permettrait la création d'une plate-forme commune pour l'évaluation et la comparaison des algorithmes proposés.

La consolidation de l'infrastructure logicielle existante par une modularisation accrue paraît souhaitable. En particulier, le système de gestion de processeurs d'image est un concept qui paraît être suffisamment général pour devenir un module séparé. Par ailleurs, une documentation plus technique donnant une vue d'ensemble aiderait des contributeurs potentiels à s'insérer dans le code.

---

<sup>29</sup>Plus de 7500 téléchargements à l'écriture de ce manuscrit.

# Annexe C

## Segmentation : variations d'énergie des contraintes

### C.1 Approximation de la variation de distance en un point

$$\Delta^a D_k(x)$$

Nous souhaitons approcher le changement  $\Delta D_k(x)$  de la distance  $D_k$  extérieure à  $O_k$  (voir § 4.5, p. 70 et § 4.5.1, p. 71). Une illustration de la variation exacte de cette distance est présentée à la figure C.1. Nous ne calculons la variation que sur des points  $x$  dont le plus proche élément de surface  $s \in S_k$  est voisin de  $p$ . Ceci assure que la distance  $D_k$  en  $x$  est effectivement affectée par le changement en  $p$ . Soit  $S_{k,p} \in S_k$  l'ensemble d'éléments de surface de  $S_k$  voisins de  $p$ . On définit de la même manière  $S'_{k,p} \in S_k$ , après le changement. En se limitant ainsi aux éléments de surface voisins de  $p$ , on peut calculer la variation de la distance à la surface  $S_{k,p}$  :  $D_{S'_{k,p}}(x) - D_{S_{k,p}}(x)$  (voir § 4.5, p. 70 pour une définition de la distance à une surface). On fait donc l'approximation que la variation de distance en un point  $x$  est entièrement décrite par ce changement de surface voisinant  $p$  :

$$\Delta^a D_k(x) = D_{S'_{k,p}}(x) - D_{S_{k,p}}(x) \quad (\text{C.1})$$

Ce calcul est efficace en pratique. La variation de distance est précalculée sur un voisinage  $N^{26}$  pour toutes les configurations possibles d'éléments de surface autour de  $p$ , avant et après le changement. Un voxel ayant 6 faces, il y a donc  $2^{2*6} = 4096$  possibilités (voir figure C.2). La variation  $\Delta^a D_k(x)$  en un point  $x$  est calculée à partir de la variation sur les voisins correspondant de  $N^{26}(p)$ .

### C.2 Différentielle d'une distance

Soit  $A$  et  $B$  deux points de  $\mathbb{R}^3$ . La distance euclidienne  $d(A, B)$  est

$$d(A, B) \triangleq (\overrightarrow{AB} \cdot \overrightarrow{AB})^{\frac{1}{2}} \quad (\text{C.2})$$



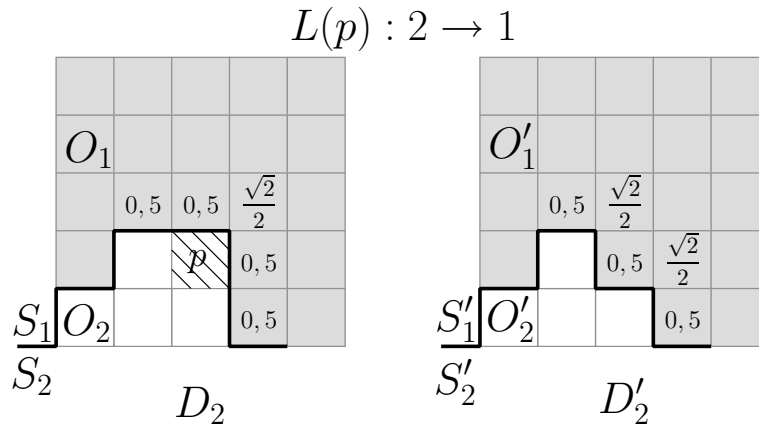


FIG. C.1 – Le changement de label au point  $p$  provoque un changement de la carte de distance. Les valeurs exactes de la distance extérieure à  $O_2$  sont affichées sur des points proches de  $p$ , avant et après le changement. Nous allons ici, approcher ce changement  $\Delta^a D_2(x)$  pour certains points  $x \in O_1$ .

$$\begin{array}{ccc}
 \begin{array}{|c|c|c|} \hline \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 1+\frac{\sqrt{2}}{2} \\ \hline 0,5 & 0,5 & \frac{\sqrt{2}}{2} \\ \hline \frac{\sqrt{2}}{2} & 0,5 & \frac{\sqrt{2}}{2} \\ \hline \end{array} & - & \begin{array}{|c|c|c|} \hline \frac{\sqrt{2}}{2} & 0,5 & \frac{\sqrt{2}}{2} \\ \hline \frac{\sqrt{2}}{2} & 0,5 & 0,5 \\ \hline 1+\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \hline \end{array} \\
 D_{S'_{2,p}} & & D_{S_{2,p}} \\
 \end{array} = \begin{array}{|c|c|c|} \hline 0 & \frac{\sqrt{2}-1}{2} & 1 \\ \hline \frac{1-\sqrt{2}}{2} & 0 & \frac{\sqrt{2}-1}{2} \\ \hline -1 & \frac{1-\sqrt{2}}{2} & 0 \\ \hline \end{array} \\
 & & \Delta^a D_2$$

FIG. C.2 – Variation de distance dans un voisinage autour de  $p$ , en considérant uniquement les éléments de surface adjacents à  $p$ .

On note le vecteur unitaire  $\vec{u}_{AB} \triangleq \frac{\vec{AB}}{d(A,B)}$ . On peut alors différencier la distance :

$$\begin{aligned}\delta d(A, B) &= \frac{\delta(\vec{AB} \cdot \vec{AB})}{2(\vec{AB} \cdot \vec{AB})^{\frac{1}{2}}} \\ &= \frac{2\delta(\vec{AB}) \cdot \vec{AB}}{2d(A, B)} \\ &= \delta(\vec{AB}) \cdot \vec{u}_{AB}\end{aligned}\tag{C.3}$$

On a donc finalement :

$$\delta d(A, B) = (\delta B - \delta A) \cdot \vec{u}_{AB}\tag{C.4}$$

où  $\delta A$  et  $\delta B$  sont les déplacements infinitésimaux de  $A$  et  $B$ .



# Annexe D

## Publications de l'auteur

### Revue internationale

M. Bosc, F. Heitz, J.-P. Armspach, I. Namer, D.Gounot, L. Rumbach « *Automatic Change Detection in Multi-Modal Serial MRI : Application to Multiple Sclerosis Lesion Evolution* » NeuroImage, 20(2) : 643-656, Octobre 2003

### Conférences internationales avec actes et comité de lecture

M. Bosc, F. Heitz, J.-P. Armspach « *Statistical Atlas-Based Sub-Voxel Segmentation of 3D Brain MRI* » accepté à IEEE International Conference on Image Processing (ICIP'03), 14-17 Septembre 2003, Barcelone, Espagne

M. Bosc, T. Vik, J.-P. Armspach, F. Heitz « *ImLib3D : A High Performance, Open Source, Medical Image Processing Framework in C++* » accepté à Medical Image Computing and Computer Assisted Intervention (MICCAI'03), 16-18 Novembre, Montréal, Canada



# Bibliographie

- [1] T. Aach and A. Kaup. Bayesian algorithms for adaptive change detection in image sequences using markov random fields. *Signal Processing : Image Communication*, 7 :147–160, August 1995.
- [2] J. Ashburner and K.J. Friston. Voxel-based morphometry : the methods. *NeuroImage*, 11 :805– 821, 2000.
- [3] M. Basseville and I.V. Nikiforov. *Detection of Abrupt Changes : Theory and Application*. Prentice Hall, 1993. Available from <http://www.irisa.fr/sigma2/kniga/kniga.pdf>.
- [4] A. Bernasconi, SB. Antel, D.L. Collins, N. Bernasconi, A. Olivier, F. Dubeau, GB. Pike, F. Andermann, and D.L. Arnold. Texture analysis and morphological processing of MRI assist detection of focal cortical dysplasia in extra-temporal partial epilepsy. *Annals of Neurology*, 49(6) :770–775, 2001.
- [5] M.J. Black, D.J. Fleet, and Y. Yacoob. Robustly estimating changes in image appearance. *Computer Vision and Image Understanding*, 78(1) :8–31, 2000.
- [6] P. Bouthemy and P. Lalande. Recovery of moving object masks in an image sequence using local spatiotemporal contextual information. *Optical Engineering*, 32(6) :1205–1212, June 1993.
- [7] L. Bruzzone and D.F. Prieto. An adaptive semi-parametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images. *IEEE Transactions on Image Processing*, 11(4) :452–466, 2002.
- [8] G. Bueno, O. Musse, F. Heitz, and J.-P. Armspach. 3D segmentations of anatomical structures in MR images on large data bases. *Magnetic Resonance Imaging*, 19/1 :73–88, 2001.
- [9] P. Chatelain. Identifying facsimile duplicates using radial pixel densities. *International Journal on Document Analysis and Recognition*, 4 :219–225, 2002.
- [10] A. Cocosco, V. Kollokian, K.S. Kwan, and A.C. Evans. Brainweb : Online interface to a 3D MRI simulated database. <http://www.bic.mni.mcgill.ca/brainweb>.
- [11] J. Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20 :37–46, 1960.
- [12] M. Cohen, R. DuBois, and M. Zeineh. Rapid and effective correction of RF inhomogeneity for high field magnetic resonance imaging. *Human Brain Mapping*, 10 :204–211, 2000.
- [13] Y. Cointepas. *Modélisation homotopique et segmentation tridimensionnelles du cortex cérébral à partir d'IRM pour la résolution des problèmes directs et inverses en EEG et en MEG*. PhD thesis, École Nationale Supérieure des Télécommunications de Paris, 1999.

- [14] D. Collins and A. Evans. Animal : automatic nonlinear image matching and anatomical labeling. In A. Toga, editor, *Brain Warping*, pages 133–142. Academic Press, 1999.
- [15] D.L. Collins, A.P. Zijdenbos, V. Kollokian, J.G. Sled, C.J. Holmes, and A.C. Evans. Design and construction of a realistic digital brain phantom. *IEEE Transactions on Medical Imaging*, 17 :436–448, June 1998.
- [16] W.L. Curati, W.L. Williams, A. Oatridge, J.V. Hajnal, N. Saeed, and G.M. Bydder. Use of subvoxel registration and subtraction to improve demonstration of contrast enhancement in MRI of the brain. *Neuroradiology*, 38 :717–723, 1996.
- [17] A.M. Dale, B. Fischl, and M. Sereno. Cortical surface-based analysis. i. Segmentation and surface reconstruction. *NeuroImage*, 9 :179–194, 1999.
- [18] T.J. Danaher. An empirical BRDF correction for Landsat TM and ETM+ imagery. In *Proceedings of the 11th Australasian Remote Sensing and Photogrammetry Conference, Brisbane, Australia*, September 2002.
- [19] J. Darbon, T. Géraud, and A. Duret-Lutz. Generic implementation of morphological image operations. In *6th International Symposium on Mathematical Morphology (ISMM'2000), Sydney, Australia, April 2002*, 2002.
- [20] C. Davatzikos and R. N. Bryan. Using a deformable surface model to obtain a shape representation of the cortex. *IEEE Transactions on Medical Imaging*, 15 :785–795, 1996.
- [21] C. Davatzikos, A. Genc, D. Xu, and S.M. Resnick. Voxel based morphometry using the ravens maps : Methods and validation using simulated longitudinal atrophy. *NeuroImage*, 14 :1361–1369, 2001.
- [22] C. DeCarli, J. Moisog, D.G. Murphy, D. Teichberg, S.I. Rapoport, and B. Horwitz. Method of quantification of brain, ventricular and sub-arachnoid CSF volumes for MRI images. *Journal of Computerized Tomography*, 16(2) :274–284, 1992.
- [23] P.J. Deer. Digital change detection techniques : civilian and military applications. In *International Symposium on Spectral Science Research ISSSR-95 Melbourne, Australia, November 26 - December 1*, 1995.
- [24] S. Derrode, G. Mercier, and W. Pieczynski. Unsupervised change detection in SAR images using a multicomponent HMC model. In *Multi-Temp'03, Ispra, Italy, July 16-18*, 2003.
- [25] A.C. Downie. SRS-X : The scottish radiological society educational resource. <http://www.radiology.co.uk/srs-x/index.htm>, 2002.
- [26] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern classification*. Wiley, 2001.
- [27] B. Fischl, M. Sereno, and A.M. Dale. Cortical surface-based analysis. II. Inflation, flattening and a surface-based coordinate system. *NeuroImage*, 9 :195–207, 1999.
- [28] P. Freeborough, R. Woods, and N. Fox. Accurate registration of serial 3D MR brain images and its application to visualizing change in neurodegenerative disorders. *Journal of Computed Assisted Tomography*, 20 :1012–1022, 1996.
- [29] P.A. Freeborough and N.C. Fox. Modeling brain deformations in Alzheimer disease by fluid registration of serial 3D MR images. *Journal of Computed Assisted Tomography*, 22(5) :838–843, 1998.
- [30] K. J. Friston, A. P. Holmes, K. J. Worsley, J. P. Poline, C. D. Frith, and R. S. J. Frackowiak. Statistical parametric maps in functional imaging : A general linear approach. *Human Brain Mapping*, 2 :189–210, 1995.

- [31] S. Geman and D. Geman. Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6 :721–741, 1984.
- [32] G. Gerig, O. Kubler, R. Kikinis, and F. Jolesz. Nonlinear anisotropic filtering of MRI data. *IEEE Transactions on Medical Imaging*, 11(2) :221–232, 12 1992.
- [33] G. Gerig, D. Welte, C. Guttmann, A. Colchester, and G. Székely. Exploring the discrimination power of the time domain for segmentation and characterization of lesions in serial MR data. In *Lecture Notes in Computer Science Proceedings of Medical Image Computing and Computer Assisted Intervention-MICCAI'98*, volume 1496, pages 469–480, 1999.
- [34] R.L. Gilmore, M.D. Childress, C.M. Leonard, R. Quisling, S. Roper, and S. Eiehschenk. Hippocampal volumetrics in patients with temporal lobe and extemporal lobe epilepsy. *Neurology*, 44(A350), Dec. 1994.
- [35] D. Goldberg-Zimring, A. Achiron, S. Miron, M. Faibel, and H. Azhari. Automated detection and characterization of multiple sclerosis lesions in brain MR images. *Magnetic Resonance Imaging*, 16 :311–316, 1998.
- [36] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky. Cortex segmentation : A fast variational geometric approach. *IEEE Transactions on Medical Imaging*, 21(2) :1544–51, December 2002.
- [37] J. Gomes and O. Faugeras. Reconciling Distance Functions and Level Sets. In Mads Nielsen, P. Johansen, O.F. Olsen, and J. Weickert, editors, *Scale-Space Theories in Computer Vision*, volume 1682 of *Lecture Notes in Computer Science*, pages 70–81. Springer Verlag, 1999.
- [38] J. Gomes and O.D. Faugeras. Reconciling Distance Functions and Level Sets. *Journal of Visual Communication and Image Representation*, 11 :209–223, 2000.
- [39] T.J. Grabowski, R.J. Frank, N.R. Szumski, C.K. Brown, and H. Damasio. Validation of partial tissue segmentation of single-channel magnetic resonance images of the brain. *NeuroImage*, 12 :640–656, 2000.
- [40] J. Grimaud, M. Lai, J. Thorpe, P. Adeleine, L. Wang, G. Barker, D. Plummer, P.S. Tofts, W.I. McDonald, and D. Miller. Quantification of MRI lesion load in multiple sclerosis : a comparison of three computer-assisted techniques. *Magnetic Resonance Imaging*, 14(5) :495–505, 1996.
- [41] J. Hajnal, N. Saeed, A. Oatridge, E. Williams, I. Young, and G. Bydder. Detection of subtle brain changes using subvoxel registration and subtraction of serial MR images. *Journal of Computed Assisted Tomography*, 19(5) :677–691, September 1995.
- [42] K.P. Hinshaw, A.V. Poliakov, E.B. Moore, R.F. Martin, L.G. Shapiro, and J.F. Brinkley. Shape-based cortical surface segmentation for visualization and brain mapping. *NeuroImage*, 16 :295–316, 2002.
- [43] J. Hofmann. Free software, big business? *Deutsche Bank Research Publications*, 32, November 2002. <http://www.dbresearch.com>.
- [44] J.P. Hornak. *The Basics of MRI, a hypertext book on magnetic resonance imaging*. Hornak, J.P., 1996. Available from <http://www.cis.rit.edu/htbooks/mri>.
- [45] Y. Hsu, H. Nagel, and G. Rekkers. New likelihood test methods for change detection in image sequences. *Computer Vision, Graphics, and Image Processing*, 26 :73–106, 1984.



- [46] C. Ichoku, Y.J. Kaufman, Giglio L., Z. Li, R.H. Fraser, Jin J.Z., and W.M. Park. Comparative analysis of daytime fire detection algorithms using AVHRR data for the 1995 fire season in Canada : perspective for MODIS. *International Journal of Remote Sensing*, 24(8) :1669–1690, 2003.
- [47] Michal Irani and P. Anandan. A unified approach to moving object detection in 2D and 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(6) :577, June 1997.
- [48] J.R. Jenson. *Introductory Digital Image Processing : A Remote Sensing Perspective 2nd Edition*. Prentice Hall, New Jersey, 1996.
- [49] K.A. Johnson and J.A. Becker. The whole brain atlas. <http://www.med.harvard.edu/AANLIB/home.html>, 1999.
- [50] Armspach J.P., Gounot D., Rumbach L., and Chambron J. In vivo determination of multiexponential t2 relaxation in the brain of patients with multiple sclerosis. *Magnetic Resonance Imaging*, 9 :107–113, 1991.
- [51] M. Kamber, R. Shinghal, DL. Collins, S. Francis, and A. Evans. Model-based 3D segmentation of multiple sclerosis lesions in magnetic resonance brain images. *IEEE Transactions on Medical Imaging*, 14 :442–453, 1995.
- [52] M. Kass, A. Witkin, and D. Terzopoulos. Snakes : Active contour models. *International Journal of Computer Vision*, 1(4) :321–331, 1987.
- [53] S. Kay. *Fundamentals of statistical signal processing : detection theory*. Prentice Hall, 1998.
- [54] C.M. Kelty. Free software/free science. *First-Monday - peer reviewed journal on the internet*, December 2001.
- [55] R. Kikinis, CRG. Guttman, D. Metcalf, W. Wells, G. Ettinger, H. Weiner, and F. Jolesz. Quantitative follow-up of patients with multiple sclerosis using MRI : Technical aspects. *Magnetic Resonance Imaging*, 9 :519–530, 1999.
- [56] E.E. Kim. An introduction to open source communities. *Blue Oxen Associates Reports*, 2003. <http://www.blueoxen.org/research/00007>.
- [57] E.C. Klatt. The internet pathology laboratory for medical education. <http://medlib.med.utah.edu/WebPath/webpath.html>, 2002.
- [58] MI. Kohn, NK. Tanna, and GT. Herman. Analysis of brain and cerebrospinal fluid volumes with MR imaging. I. Methods, reliability and validation. *Radiology*, 178 :115–122, 1991.
- [59] N. Kriegeskorte and R. Goebel. An efficient algorithm for topologically correct segmentation of the cortical sheet in anatomical MR volumes. *NeuroImage*, 14 :329–346, 2001.
- [60] S. Lee, N. Moisa, and M. Weiss. Open source as a signaling device - an economic analysis. *Working Paper Series : Finance and Accounting*, 2003. (paper 102) Goethe University Frankfurt am Main <http://ideas.repec.org/p/fra/franaf/102.html>.
- [61] T.M. Lehmann, C. Gönner, and K. Spitzer. Survey : interpolation methods in medical image processing. *IEEE Transactions on Medical Imaging*, 18(11) :1049–1075, 1999.
- [62] L. Lemieux, U.C. Wiesmann, N.F. Moran, and D.R. Fish. The detection and significance of subtle changes in mixed-signal brain lesions by serial MRI scan matching and spatial normalization. *Medical Image Analysis*, 2(3) :227–242, 1998.

- [63] C. Li, D.B. Goldgof, and L.O. Hall. Knowledge-based classification and tissue labeling of MR images of the human brain. *IEEE Transactions on Medical Imaging*, 12(4) :740, 1993.
- [64] L. Li and M.K.H. Leung. Integrating intensity and texture differences for robust change detection. *IEEE Transactions on Image Processing*, 11(2) :105–112, February 2002.
- [65] KO. Lim and A. Pfefferbaum. Segmentation of MR brain images into cerebrospinal fluid spaces, white and gray matter. *Journal of Computed Assisted Tomography*, 13 :558–593, 1989.
- [66] S. Lohr. An alternative to Microsoft gains support in high places. *New York Times*, page 1, September-5 2002.
- [67] F.D. Lublin and S.C. Reingold. Defining the clinical course of multiple sclerosis : results of an international survey. National multiple sclerosis society (USA) advisory committee on clinical trials of new agents in multiple sclerosis. *Neurology*, 46(4) :907–911, April 1996.
- [68] D. MacDonald, N. Kabani, D. Avis, and A.C. Evans. Automated 3-D extraction of inner and outer surfaces of cerebral cortex from MRI. *NeuroImage*, 12 :340–356, 2000.
- [69] E.N. Marieb. *Essentials of human anatomy and physiology*. Redwood City, Calif., Benjamin/Cummings Pub. Co., 1991.
- [70] J.L. Marroquin, B.C. Vemuri, S. Botello, F. Calderon, and A. Fernandez-Bouzas. An accurate and efficient Bayesian method for automatic segmentation of brain MRI. *IEEE Transactions on Medical Imaging*, 21(8) :934–945, August 2002.
- [71] R. Marteau. *La sclérose en plaques*. Editions Odile Jacob, 1998.
- [72] W.I. McDonald, A. Compston, G. Edan, D. Goodkin, H.P. Hartung, F.D. Lublin, H.F. McFarland, D.W. Paty, C.H. Polman, S.C. Reingold, M. Sandberg-Wollheim, W. Sibley, A. Thompson, S. van den Noort, B.Y. Weinschenker, and Wolinsky J.S. Recommended diagnostic criteria for multiple sclerosis : guidelines from the international panel on the diagnosis of multiple sclerosis. *Annals of Neurology*, 50(1) :121–127, July 2001.
- [73] T. McInerney and D. Terzopoulos. Deformable models in medical image analysis : A survey. *Medical Image Analysis*, 1(2) :91–108, 1996.
- [74] JR. Mitchell, SJ. Karlik, DH. Lee, M. Eliasziw, GP. Rice, and A. Fenster. The variability of manual and computer assisted quantification of multiple sclerosis lesion volumes. *Medical Physics*, 23(1) :85–97, 1996.
- [75] A. Mitiche and P. Bouthemy. Computation and analysis of image motion : A synopsis of current problems and methods. *International Journal of Computer Vision*, 19(1) :29–55, 1996.
- [76] G.E. Moore. Cramming more components onto integrated circuits. *Electronics*, 38 :114 – 117, 1965.
- [77] O. Musse, F. Heitz, and J.-P. Armspach. Topology preserving deformable image matching using constrained hierarchical parametric models. *IEEE Transactions on Image Processing*, 10(7) :1081–1093, 2001.
- [78] O. Musse, F. Heitz, and J.-P. Armspach. Fast deformable matching of 3D images over multiscale nested subspaces. application to atlas-based MRI segmentation. *Pattern Recognition*, 36(8) :1881–1899, Aug 2003.

- [79] D. Musser and A. Saini. *STL Tutorial and Reference Guide. Professional Computing Series*. Addison-Wesley, 1996.
- [80] J.A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7 :308–315, 1965.
- [81] F.H. Netter. *The Ciba collection of medical illustrations, nervous system (Vol. 1) part I and II*. Summit, NJ : Hoechstetter Printing, 1995.
- [82] W.J. Niessen, K.L. Vincken, J. Weickert, B.M. ter Haar Romeny, and M.A. Viergever. Multiscale segmentation of three-dimensional MR brain images. *International Journal of Computer Vision*, 31(2/3) :185–202, 1999.
- [83] C. Nikou, G. Bueno, F. Heitz, and J.-P. Armspach. A joint physics-based statistical deformable model for multimodal brain image analysis. *IEEE Transactions on Medical Imaging*, 20 :1026–1037, Oct 2001.
- [84] J. Nolte. *The Human Brain : An Introduction to Its Functional Anatomy 5th edition, revised*. Elsevier Science, 2001.
- [85] J.M. Odobez and P. Bouthemy. Separation of moving regions from background in an image sequence acquired with a mobile camera. In H.H. Li, S. Sun, and H. Derin, editors, *Video Data Compression for Multimedia Computing*, chapter 8, pages 283–311. Kluwer Academic Publisher, 1997.
- [86] J.I. O’Riordan, C.M. Gawne, A. Coles, L. Wang, D.A. Compston, and P. Tofts. T1 hypointense lesion load in secondary progressive multiple sclerosis : a comparison of pre versus post contrast loads and of manual versus semi automated threshold techniques for lesion segmentation. *Multiple Sclerosis*, 4 :408–412, 1998.
- [87] N. Otsu. A threshold selection method from grey-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9 :62–66, 1979.
- [88] N. Paragios and R. Deriche. Geodesic active contours and level sets for the detection and tracking of moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(3), March 2000.
- [89] D. Pham and J. Prince. Adaptive fuzzy segmentation of Magnetic Resonance Images. *IEEE Transactions on Medical Imaging*, 18(9) :737–752, 1999.
- [90] D.L. Pham, C. Xu, and J.L. Prince. A survey of current methods in medical image segmentation. *Annual Review of Biomedical Engineering*, 2, 2000.
- [91] E.S. Raymond. *The Cathedral and the Bazaar*. O’Reilly, 2001. <http://catb.org/~esr/writings/cathedral-bazaar>.
- [92] A.L. Reiss, F. Faruque, S. Abrams, T. Beaty, R.N. Bryan, and H. Moser. Neuroanatomy or Rett syndrome : a volumetric imaging study. *Annals of Neurology*, 34(2) :227–234, 1993.
- [93] D. Rey, G. Subsol, H. Delingette, and N. Ayache. Automatic detection and segmentation of evolving processes in 3D medical images : Application to multiple sclerosis. *Medical Image Analysis*, 6(2) :163–179, 2002.
- [94] H. Rouvière and A. Delmas. *Anatomie humaine : descriptive, topographique et fonctionnelle*. Masson, 2002.
- [95] S. Ruan, C. Jaggi, J. Xue, J. Fadilij, and D. Bloyet. Brain tissue classification of Magnetic Resonance Images using partial volume modeling. *IEEE Transactions on Medical Imaging*, 19(12) :1179–1187, December 2000.

- [96] J.A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, 1999.
- [97] D.W. Shattuck and R.M. Leahy. Brainsuite : An automated cortical surface identification tool. *Medical Image Analysis*, 6(12) :129–142, 12 2002.
- [98] D. Shen, E.H. Herskovits, and C. Davatzikos. An adaptive-focus statistical shape model for segmentation and shape modeling of 3D brain structures. *IEEE Transactions on Medical Imaging*, 20(4) :257–270, April 2001.
- [99] J.D. Shepherd03 and J.R. Dymond. Correcting satellite imagery for the variance of reflectance and illumination with topography. *International Journal of Remote Sensing*, 24(17) :3503–3514, 2003.
- [100] J. Sijbers, A.J. den Dekker, J. Van Audekerke, Verhoye M., and D. Van Dyck. Estimation of the noise in magnitude mr images. *Magnetic Resonance Imaging*, 16(1) :87–90, 1998.
- [101] J. Simon, R. Schiffer, R. Rudick, and R. Herndon. Quantitative determination of MS-induced corpus callosum atrophy in vivo using MR imaging. *American Journal of Neuroradiology*, 8 :599–604, 1987.
- [102] S.M. Smith, N. De Stefano, M. Jenkinson, and P.M. Matthews. Measurement of brain change over time. Technical Report TR00SMS1, FMRIB John Radcliffe Hospital, Headington, Oxford OX3 9DU, UK, 2000.
- [103] H. Soltanian-Zadeh, J.P. Windham, and A.E. Yagle. A multidimensional nonlinear edge-preserving filter for magnetic resonance image restoration. *IEEE Transactions on Medical Imaging*, 4 :147–161, 1995.
- [104] A. Stevens and J. Lowe. *Pathology*. Mosby - Times mirror international publishers, 1995.
- [105] B. Stroustrup. *The C++ Programming Language*. Addison-Wesley, 2002.
- [106] J. Suri, S. Singh, and L. Reden. Computer vision and pattern recognition techniques for 2-D and 3-D cerebral cortical segmentation (part-I) : A state of the art review. *Pattern Analysis and Applications*, 5(1) :46–47, 2002.
- [107] P. Thevenaz, T. Blu, and M. Unser. Interpolation revisited. *IEEE Transactions on Medical Imaging*, 19(7) :739–758, July 2000.
- [108] J-P. Thirion. Image matching as a diffusion process : an analogy with Maxwell’s demons. *Medical Image Analysis*, 2(3) :243–260, 1998.
- [109] J-P. Thirion and G. Calmon. Deformation analysis to detect and quantify active lesions in three-dimensional medical image sequences. *IEEE Transactions on Medical Imaging*, 18(5) :429–441, 1999.
- [110] P. Thompson and A.W. Toga. Warping strategies for intersubject registration. In I.N. Bankman, editor, *Handbook of Medical Imaging*, chapter 36, pages 569–601. Academic press, 1999.
- [111] A.W. Toga and Thompson P.M. Maps of the brain. *The Anatomical Record (New Anatomist)*, 265 :37–53, 2001.
- [112] A.W. Toga and P. Thompson. An introduction to brain warping. In A.W. Toga, editor, *Brain Warping*, chapter 1, pages 1–26. Academic Press, 1999.
- [113] M. Torregrossa, C.V. Zint, and P. Poulet. The use of MRI information to initialize the first stage of the image reconstruction algorithm in optical tomography. In *Proc SPIE’03*, volume 4255, pages 264–272, 2003.

- [114] D. Toth, T. Aach, and V. Metzler. Illumination-invariant change detection. In *IEEE Southwest Symposium on Image Analysis and Interpretation 2000 Austin, TX on April 2-4, 2000*.
- [115] A. Tourbah. IRM et sclérose en plaques. *Neurologies*, 33(4) :270–274, September 2001.
- [116] J.R.G. Townshend, C.O. Justice, and C. Gurney. The impact of misregistration on change detection. *IEEE Transactions on Geoscience and Remote Sensing*, 30(5) :1054–1060, 1992.
- [117] A. Tsai, A. Yezzi, W. Wells, C. Tempany, D. Tucker, A. Fa, W.E. Grimson, and A. Willsky. A shape-based approach to the segmentation of medical imagery using level sets. *IEEE Transactions on Medical Imaging*, 22(2) :137–154, February 2003.
- [118] J. Udupa, L. Wei, S. Samarasekera, M. Van Buchem, and R. Grossman. Multiple sclerosis lesion quantification using fuzzy-connectedness principles. *IEEE Transactions on Medical Imaging*, 17 :598–609, 1997.
- [119] M. Vaillant and C. Davatzikos. Finding parametric representations of cortical sulci using an active contour model. *Medical Image Analysis*, 1(4) :295–315, Sep 1997.
- [120] K. Van Leemput, F. Maes, D. Vandermeulen, A. Colchester, and P. Suetens. Automated segmentation of multiple sclerosis lesions by model outlier detection. *IEEE Transactions on Medical Imaging*, 20(8) :677–688, August 2001.
- [121] K. Van Leemput, F. Maes, D. Vandermeulen, and P. Suetens. Automated model-based tissue classification of MR images of the brain. *IEEE Transactions on Medical Imaging*, 18(10) :897–908, October 1999.
- [122] MW. Vannier, RL. Butterfield, and D. Jordan. Multispectral analysis of magnetic resonance images. *Radiology*, 154 :221–224, 1985.
- [123] S.J. Vaughan-Nichols. Building better software with better tools. *IEEE Computer*, 36(9) :12–14, September 2003.
- [124] K. Warfield, S., M. Kaus, F.A. Jolesz, and R. Kikinis. Adaptive, template moderated, spatially varying statistical classification. *Medical Image Analysis*, 4(1) :43–55, 2000.
- [125] S.K. Warfield, F. Talos, A. Tei, A. Bharatha, A. Nabavi, M. Ferrant, P.M. Black, F.A. Jolesz, and R. Kikinis. Real-time registration of volumetric brain MRI by biomechanical simulation of deformation during image guided neurosurgery. *Computing and Visualization in Science*, 5(1) :3–11, 2002.
- [126] A.J. Worth, N. Makris, V.S.Jr. Caviness, and D.N. Kennedy. Neuroanatomical segmentation in MRI : Technological objectives. *International Journal of Pattern Recognition and Artificial Intelligence*, 11(8) :1161–1187, 1997.
- [127] C. Xu, D.L. Pham, M.E. Rettmann, D.N. Yu, and J.L. Prince. Reconstruction of the human cerebral cortex from magnetic resonance images. *IEEE Transactions on Medical Imaging*, 18(6) :467–480, June 1999.
- [128] J.H. Xue, S. Ruan, B. Moretti, M. Revenu, and D. Bloyet. Knowledge-based segmentation and labeling of brain structures from MRI images. *Pattern Recognition Letters*, 22 :395–405, 2001.
- [129] X. Zeng. *Volumetric Layer Segmentation Using a Generic Shape Constraint with Applications to Cortical Shape Analysis*. PhD thesis, Yale University, 2000.
- [130] X. Zeng, L. H. Staib, R. T. Schultz, and J. S. Duncan. Segmentation and measurement of the cortex from 3D MR images using coupled surfaces propagation. *IEEE Transactions on Medical Imaging*, 18(10) :100–111, October 1999.

- 
- [131] 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, January 2001.
- [132] K. Zilles. Automatic detection and segmentation of evolving processes in 3D medical images : Application to multiple sclerosis. In G. Paxinos, editor, *The Human Nervous System*. Academic Press, Inc., San Diego, CA, 1990.
- [133] B. Zitova and J. Flusser. Image registration methods : a survey. *Image and Vision Computing*, 21 :977–1000, 2003.