



Intégration de connaissances anatomiques a priori dans des modèles géométriques

Sahar Hassan

► To cite this version:

Sahar Hassan. Intégration de connaissances anatomiques a priori dans des modèles géométriques. Mathématiques générales [math.GM]. Université de Grenoble, 2011. Français. NNT : 2011GRENM020 . tel-00607260v2

HAL Id: tel-00607260

<https://theses.hal.science/tel-00607260v2>

Submitted on 6 Feb 2012

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

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

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Mathématiques & Informatique**

Arrêté ministériel : 7 Août 2006

Présentée par

Sahar Hassan

Thèse dirigée par **Georges-Pierre Bonneau**
et codirigée par **Franck Hétroy**

préparée au sein du **Laboratoire Jean Kuntzmann**
et de l'**École Doctorale MSTII**

Intégration de connaissances anatomiques a priori dans des modèles géométriques

Thèse soutenue publiquement le **20 juin, 2011**,
devant le jury composé de :

Mme. Annick Montanvert

Professeur - Université Pierre-Mendès-France, Président

Mme. Bianca Falcidieno

Directrice de recherche, CNR IMATI-Ge, Rapporteur

M. Pierre Alliez

Directeur de recherche - INRIA Sophia-Antipolis, Rapporteur

M. Florent Dupont

Professeur - Université Lyon 1, Examineur

M. Georges-Pierre Bonneau

Professeur - Université de Grenoble, Directeur de thèse

M. Franck Hétroy

Maître de conférences - Université de Grenoble, Co-Directeur de thèse



فَقُمْ بِعِلْمٍ وَلَا تَطْلُبْ بِهِ بَدَلًا

فَالنَّاسُ مَوْتَى وَأَهْلُ الْعِلْمِ أَحْيَاءُ

الإمام علي

Cherches la connaissance et n'attends pas de récompense, les gens meurent et les savants vivront à jamais.

Imam ALI

Remerciements

Une thèse est normalement un premier pas que l'on fait dans le domaine de la recherche. Pour moi, elle était beaucoup plus que ça. Elle était une découverte d'un pays avec sa langue et sa culture. C'était avant tout une expérience humaine à plus que trois dimensions.

Je tiens à remercier mes encadrants Franck Hétroy et Georges-Pierre Bonneau qui m'ont donné de leur temps et leurs expériences sans parcimonie. Un grand merci à Franck pour tout ce qu'il a fait pour me guider, que ça soit dans la thèse ou toute autre démarche liée à mon insertion dans le monde des enseignants chercheurs. Ma gratitude va aussi à François Faure et Olivier Palombi qui ont beaucoup contribué à ces travaux de recherche.

J'aimerais ensuite remercier l'équipe EVASION pour son atmosphère accueillant et la bonne humeur qui règne toujours dans le couloir H. Je remercie en particulier Anne et Elodie pour leur patience, disponibilité et compréhension.

Je souhaite également remercier Bianca Falcidieno et Pierre Alliez d'avoir accepté de rapporter mes travaux de thèse, ainsi que Annick Montanvert d'avoir accepté de présider mon jury.

Un grand grand merci à mes amis syriens et français. Un merci particulier à Farah qui a su rester à mes côtés malgré les milliers de kilomètres qui nous séparent. Puis, merci à mes amis et camarades du bureau : Adeline et Adrien.

Finalement, je remercie ma mère, mes frères, ma belle famille et bien-sûr Raed, pour leur présence permanente, leurs encouragements et leur patience durant ces années.

à ma famille

TABLE DES MATIÈRES

Table des matières	1
Introduction	3
1 Contexte du travail	3
2 Contributions	4
3 Organisation du document	5
1 Représentation de connaissances	7
1 Approche basée images	8
1.1 Atlas classiques	8
1.2 Atlas informatisés	9
1.3 Atlas probabilistes	11
2 Approche basée concepts	12
2.1 Livres de terminologie	12
2.2 <i>Unified Medical Language System (UMLS)</i>	13
2.3 Ontologies	13
3 Liaisons entre l'ontologie et la géométrie	26
4 Conclusion	34
2 Localisation d'anévrismes	37
1 Motivation	37
2 État de l'art	39

2.1	Segmentation des vaisseaux	39
2.2	Détection et quantification d'anévrismes	40
3	Extraction des lignes de centre des vaisseaux	42
3.1	Définitions	42
3.2	État de l'art	43
3.3	Notre méthode	47
4	Détection automatique d'anévrismes	54
5	Quantification d'anévrismes	56
6	Déterminer les vaisseaux porteurs d'anévrismes	58
6.1	Les connaissances <i>a priori</i>	58
6.2	La mise en correspondance	59
7	Résultats	62
8	Conclusion et perspectives	64
3	Segmentation de maillages	67
1	Introduction	67
1.1	Utilisation de maillages en imagerie médicale	68
1.2	Intérêt de l'apport des connaissances <i>a priori</i>	69
2	Segmentation de maillages	69
2.1	Aperçu de l'état de l'art	69
2.2	Algorithme de Simari <i>et al.</i>	73
3	Segmentation de maillage guidée par une ontologie : principe	77
4	Enrichissement de l'ontologie	78
4.1	Forme géométrique approchée	78
4.2	Taille relative	78
5	Utilisation de l'ontologie pour la segmentation de maillages	79
5.1	Initialisation	81
5.2	Étiquetage	84
5.3	Optimisation	85
5.4	Hiérarchie	94

6	Résultats	95
6.1	Comparaison avec d'autres méthodes de segmentation	95
6.2	Segmentation d'autres modèles	96
7	Discussion et perspectives	98
7.1	Résumé	98
7.2	Perspectives	99
8	Conclusion	99
Conclusion		101
	Résumé des contributions	101
	Travaux futurs	103
	Pour conclure	104
Annexe		105
A Techniques d'imagerie médicale		107
1	Imagerie par Résonance Magnétique (IRM)	107
2	La tomodensitométrie (CT-Scan)	108
3	L'angiographie	109
Index		111
Bibliographie		113

Introduction

LES applications informatiques médicales sont très connues et très utilisées aujourd’hui dans les hôpitaux du monde entier. Des applications qui gèrent le stockage et l’accès aux dossiers des patients, en passant par les applications d’analyse et visualisation d’images médicales, et jusqu’aux applications d’aide au diagnostic, à la décision et à la planification du traitement ou des gestes chirurgicaux, celles-ci ont de multiples fonctionnalités.

1 Contexte du travail

L’imagerie médicale est une ressource de données principale pour différents types d’applications. Les images viennent compléter l’expertise des médecins. Les méthodes de segmentation d’images médicales et de visualisation, par exemple, fournissent une meilleure vue de tout ou partie des images, facilitant ainsi la décision sur le traitement à appliquer.

Bien que ces images concrétisent beaucoup d’informations sur le cas étudié, une grande partie des informations reste implicite. Ces informations implicites concernent tout ce que le médecin sait sur le cas étudié. Pour un organe donné par exemple, l’expert connaît son nom, ses fonctions, sa localisation, ses relations avec d’autres organes, etc. Toutes ces connaissances *a priori*, que l’on suppose déjà acquises, jouent un rôle très important dans l’interprétation et l’utilisation des images médicales.

Représenter ces connaissances *a priori* sous une forme standard facilite leur utilisation et leur partage par la communauté médicale. Si cette forme est en plus compréhensible par l’ordinateur, ces connaissances sont intégrables dans des applications informatiques permettant ainsi de répondre à plus d’attentes des utilisateurs des applications sus-citées.

Pour répondre à ces attentes, la représentation de connaissances anatomiques *a priori* est un domaine de recherche très actif. L’intégration de ces connaissances anatomiques produit des applications médicales plus “intelligentes”, dans le sens où elles profitent de plus d’informations.

Dans cette thèse, des connaissances anatomiques *a priori* sont intégrées dans deux applications médicales. La représentation de ces connaissances et leur intégration dépendent des deux contextes.

Pour le premier contexte nous nous plaçons dans le cadre du diagnostic et du choix du traitement

des anévrismes intra-crâniens. Ces poches qui naissent dans les parois des vaisseaux sanguins peuvent être de conséquence mortelle en cas de rupture. Elles sont visibles sur des images en utilisant l'angiographie par résonance magnétique (cf. annexe A, Section 3). Les radiologues, grâce à leur expertise visuelle et leur expérience, voient ces anévrismes sur ces images et estiment leur taille.

En cas de diagnostic d'un anévrisme intra-crânien, deux traitements sont possibles : l'embolisation, et le *clipping* (pincement). Il a été montré par plusieurs études statistiques que la taille, la forme et la localisation d'un anévrisme jouent un rôle très important dans le choix du traitement à faire. Pour cela, la quantification et la localisation de l'anévrisme présentent des informations précieuses aux médecins.

Dans ce contexte, les connaissances anatomiques *a priori* sont utilisées pour déterminer le nom du vaisseau porteur de chaque anévrisme. Ces connaissances concernent les relations entre les vaisseaux qui forment l'arbre vasculaire sain.

L'application que nous proposons est une chaîne de traitement automatique qui détecte, quantifie et localise des anévrismes dans un arbre vasculaire segmenté.

Le deuxième contexte est l'identification des sous-parties d'un organe. Les techniques d'imagerie médicale de nos jours sont de plus en plus puissantes. Les images acquises nous montrent beaucoup de structures anatomiques et la plupart des organes du corps humain.

La segmentation de ces images pour extraire une structure ou un organe d'intérêt consiste à isoler cet organe du reste de l'image. Après la segmentation, nous pouvons fournir un modèle géométrique 3D (sous forme de maillage) représentant la structure (l'organe) en question permettant une meilleure visualisation de cette structure.

Cependant, certaines structures restent peu ou pas du tout visibles par les différentes techniques d'imagerie disponibles (par exemple, les ligaments), alors même que leur présence est connue grâce aux connaissances anatomiques *a priori*. Ces connaissances sont complexes et comprennent des informations sur l'organe lui-même, ainsi que des informations sur ses relations avec les organes voisins. Ces informations peuvent être modélisées par une ontologie anatomique.

Si on voulait avoir une représentation géométrique de ces parties "cachées", une solution intuitive serait d'utiliser l'ontologie (informations *a priori* sur ces organes et leurs relations avec les organes visibles) pour savoir où les placer et comment les insérer, etc.

Pour cela, nous avons besoin d'identifier sur les maillages représentant des organes visibles, les parties et les points de référence. En d'autres mots, nous voulons segmenter en sous-parties les maillages représentant les organes.

Le résultat sera un modèle géométrique complet d'un certain organe avec ses parties visibles et non visibles par imagerie. Un tel modèle peut servir pour l'enseignement de l'anatomie.

Dans ce contexte, les connaissances *a priori*, sous forme d'ontologie, sont utilisées pour guider un algorithme de segmentation de maillages. Le maillage à segmenter correspond à un organe, les segments à trouver correspondent aux parties de cet organe.

2 Contributions

Les contributions proposées dans cette thèse sont relatives aux deux contextes dans lesquels le travail a été réalisé, décrits ci-dessus.

Dans le contexte de la détection d'anévrismes intra-crâniens, nous proposons les quatre contributions suivantes.

Extraction des lignes de centre des vaisseaux. Les lignes de centre des vaisseaux sont le point de départ de la méthode de détection, quantification et localisation d'anévrismes que nous appliquons ensuite. Pour qu'elles puissent être utilisées, elles doivent être : minces, connexes, bien centrées dans les vaisseaux et comporter des connexions nettes et intuitives. Nous proposons un nouvel algorithme pour produire des lignes de centre répondant à ces besoins.

Détection automatique d'anévrismes. L'anévrisme a une forme irrégulière en comparaison avec la forme tubulaire des vaisseaux sanguins. Nous nous basons sur cette idée pour détecter les anévrismes automatiquement.

Quantification d'anévrismes. Pour chaque anévrisme détecté, nous déterminons son collet et donnons des mesures sur sa forme. Ces mesures (ainsi que la localisation) aident le spécialiste à faire le choix du traitement.

Localisation d'anévrismes. L'arbre vasculaire sain, tout comme l'arbre vasculaire du patient ayant des anévrismes, peut être représenté par un graphe. Une mise en correspondance entre un graphe représentant un arbre vasculaire sain et celui du patient est faite afin de déterminer les vaisseaux porteurs des anévrismes.

Dans le contexte de l'identification des sous-parties d'un organe à partir de sa représentation géométrique, nous proposons les deux contributions suivantes.

Ajout des informations nécessaires dans l'ontologie : l'ontologie anatomique contient beaucoup d'informations sur les structures anatomiques, mais ces informations ne sont pas suffisantes pour segmenter un maillage correspondant à un organe en ses sous-parties. Pour répondre à ce problème, nous enrichissons l'ontologie avec toutes les informations nécessaires pour accomplir la tâche de segmentation de maillages.

Utilisation de l'ontologie pour guider la segmentation de maillages : la segmentation de maillages est un vaste domaine de recherche en soi. Nous proposons un nouvel algorithme de segmentation de maillages qui profite de toutes les connaissances *a priori* fournies par l'ontologie. Cela permet de produire une segmentation d'un organe en sous-parties dont la structure correspond à celle décrite par l'ontologie.

3 Organisation du document

Ce document est composé de trois chapitres. Nous commençons dans le premier chapitre par présenter la représentation des connaissances *a priori* en informatique médicale. Nous présentons les différentes catégories de représentations existantes. Nous nous concentrons ensuite sur l'un des types de représentations de connaissances *a priori* les plus prometteurs : l'ontologie. Puisque nous utilisons ensuite l'ontologie dans un contexte géométrique, nous finissons ce chapitre en citant quelques exemples de la littérature qui font une telle utilisation de l'ontologie.

Dans le deuxième chapitre, nous détaillons notre première application dans laquelle des connaissances *a priori* sont utilisées : la localisation d'anévrismes. Nous commençons par expliquer la motivation de cette application, puis nous présentons quelques travaux antérieurs liés à cette application. La méthode de localisation d'anévrismes est ensuite détaillée étape par étape, en commençant par l'extraction des lignes de centre, en passant par la détection automatique et la quantification et en finissant par la localisation des anévrismes. Nous validons la méthode sur un groupe de patients et montrons quelques résultats. Finalement, nous concluons en présentant les travaux futurs envisagés pour couvrir plus de cas cliniques.

Le troisième chapitre détaille la seconde application, qui concerne la segmentation de maillages guidée par ontologie. Après avoir montré l'intérêt de l'utilisation des connaissances *a priori* dans ce domaine, nous donnons un bref état de l'art. Les informations ajoutées à l'ontologie sont ensuite présentées, en expliquant les raisons et les avantages de ces ajouts. Nous proposons ensuite un nouvel algorithme de segmentation de maillages, qui tire parti des informations fournies par l'ontologie. Cet algorithme est appliqué à des modèles anatomiques ainsi qu'à des modèles non anatomiques ; nous en présentons les résultats.

Enfin, en conclusion nous rappelons les contributions de cette thèse, et exposons des pistes pour des travaux futurs.

Représentation des connaissances a priori en anatomie

A priori est une locution latine signifiant “à partir de ce qui vient avant”. Les connaissances *a priori* sont les connaissances logiquement antérieures à l’expérience et aussi “indépendantes de l’expérience” (Kant). Cela s’oppose à *a posteriori*, empirique, factuel, ce qui est “issu de l’expérience” (Kant)¹.

Évidemment ce type de connaissance n’est pas réservé à un domaine spécifique. Selon la définition ci-dessus toutes les suppositions que l’on se fixe avant l’analyse et la résolution d’un problème sont des connaissances *a priori*.

Dans le domaine linguistique, par exemple, pour l’indexation automatique d’un document, les outils d’analyse doivent être paramétrés par les connaissances *a priori* dont on dispose sur le contexte du document.

*“En particulier, la production des documents suit des normes, des contraintes que l’on retrouve dans la structure et le contenu des documents. C’est ainsi que l’on sait qu’un journal télévisé est construit toujours de la même façon. Cette connaissance peut être mobilisée pour reconnaître plus facilement les séquences composant le journal et leur nature.”*² Bruno Bachimont, Institut national de l’audiovisuel (INA).

Un autre exemple sur l’utilisation des connaissances *a priori*, issu du domaine médical, est la segmentation d’images médicales. En effet, pour isoler une certaine structure anatomique des autres structures présentes dans l’image on profite des propriétés de la structure, ou des propriétés de la technique d’imagerie utilisée. Le fait de savoir « *a priori* » que les vaisseaux sanguins ont une structure tubulaire et qu’ils sont plus lumineux que l’arrière-plan dans une image a amené à la création d’une mesure de vaisseau [Frangi98].

1. http://fr.wikipedia.org/wiki/A_priori_et_a_posteriori

2. <http://www.culture.gouv.fr/culture/dgdl/rifal/indexation.htm>

Cet exemple est une utilisation spécifique à un type de connaissance. Si les connaissances changent, la solution est à changer également. Si on veut créer des solutions qui prennent les connaissances *a priori* comme de simples paramètres, on doit disposer d'une représentation formelle de ces connaissances. Les questions que l'on peut se poser sont :
Comment représenter les connaissances ? Comment les rendre partageables ? Quelles sont les utilisations possibles ?

Pour répondre à ces questions, et puisque ce travail s'intéresse à utiliser les connaissances *a priori* dans le domaine médical, nous nous restreindrons aux connaissances anatomiques.

Dans un premier temps, différentes représentations des connaissances anatomiques *a priori* sont données. Puis l'une de ces représentations (l'ontologie) est détaillée et différentes utilisations des ontologies sont présentées dans la Section 2.3. Finalement, on donne des exemples sur des liaisons entre l'ontologie et la géométrie dans la Section 3.

Selon la thèse d'Olivier Dameron [Dameron03] deux approches pour représenter les connaissances anatomiques existent ; l'approche basée images, et l'approche basée concepts. Nous allons détailler chacune de ces approches.

1 Approche basée images

Dans cette approche, les structures anatomiques sont représentées sur une ou plusieurs images. Tout type d'atlas anatomique utilise cette approche. La définition du mot *Atlas* selon le *Dictionnaire de L'Académie française* :

“Recueil de cartes géographiques.

Il se dit également de tout recueil non seulement de cartes géographiques, mais encore de planches, de reproductions de tableaux, de documents graphiques, etc., qu'on joint à un ouvrage pour en faciliter l'intelligence. ”.

En anatomie et selon [Dameron03], nous distinguons trois types d'atlas : classiques, informatisés, et probabilistes.

1.1 Atlas classiques

Ces atlas correspondent aux livres d'anatomie, où on trouve des illustrations sous forme de schémas, photos ou résultats d'examens d'imagerie accompagnés de légendes et souvent de textes descriptifs.

Ces atlas peuvent être spécialisés à une structure anatomique donnée ou génériques au corps humain entier. Le Grand atlas d'anatomie humaine [Orte06] est un exemple d'atlas s'intéressant à l'anatomie du corps humain entier. Nous y trouvons des schémas explicatifs pour les différents systèmes du corps humain, voir Figure 1.1.

Un exemple d'atlas s'intéressant à une structure anatomique spécifique est l'atlas de Duvernoy [Naidich09], qui porte sur l'anatomie du tronc cérébral et du cervelet. La Figure 1.2 montre deux images de cet atlas. Les images sont étiquetées par des chiffres qui sont détaillés en bas de

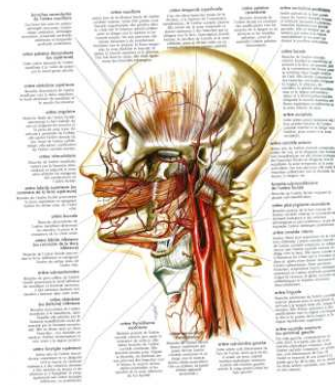
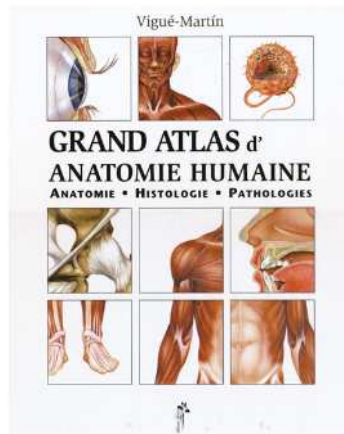


FIGURE 1.1: Exemple d'atlas classique avec une illustration de l'arbre vasculaire cérébral.

chaque image.

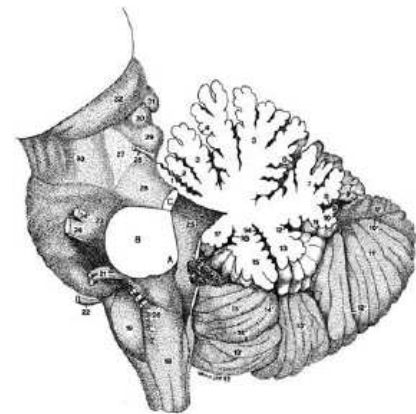
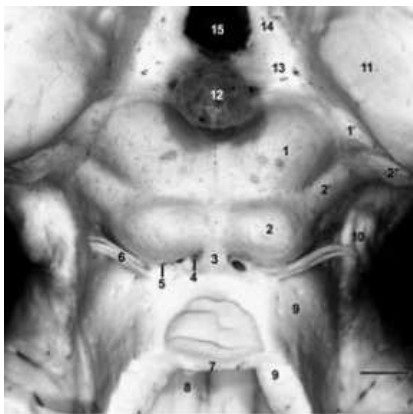


FIGURE 1.2: Des images issues de l'atlas de Duvernoy. A gauche : une vue postérieure agrandie du mésencéphale. A droite : une section médiane du vermis du cervelet.

Les auteurs ont corrélié plusieurs techniques d'imagerie médicale (IRM de diffusion, IRM clinique (3 et 4 T)³). Ces techniques permettent l'illustration de l'anatomie nucléaire et des tracts fibreux. Chaque région du tronc cérébral est ensuite analysée avec l'IRM (9.4 T) pour montrer l'anatomie des différentes parties du tronc et du cervelet avec beaucoup de précision.

Bien que ces atlas restent des références très importantes en anatomie, leur principal inconvénient est leur format papier. Ce format rend ces atlas consultables uniquement manuellement. Ils ne sont pas utilisables par des logiciels et n'offrent pas une représentation 3D de l'anatomie.

1.2 Atlas informatisés

Un atlas informatisé est un ensemble de représentations 2D ou 3D de structures anatomiques identifiées, stockées sur ordinateur. Un atlas informatisé très connu est *Voxel-Man* [Hohne96]. Il s'agit

3. Les chiffres correspondent aux puissances du champ magnétique utilisé, le T est le tesla, l'unité de mesure.

d'un modèle 3D du corps humain basé sur le projet *Visible Human Project*⁴. C'est un projet de la bibliothèque nationale de médecine américaine (National Library of Medicine). Il consiste en une représentation complète, anatomiquement détaillée, des corps sains d'un homme et d'une femme. Cette représentation provient d'acquisitions CT transversales et d'IRM de deux cadavres (homme et femme). Le cadavre masculin a été tranché à des intervalles d'un millimètre et le cadavre féminin à des intervalles de $\frac{1}{3}$ mm.

Prenons par exemple l'atlas des organes internes de l'homme de *Voxel-Man* [Hohne01, Pommert01]. Les données d'entrée viennent du projet *Visible Human* et représentent un volume de $573 \times 330 \times 774$ voxels de 1mm^3 chacun. Ce volume correspond à un ensemble de 774 tranches, chacune contenant trois images, voir Figure 1.3 : une image cryogénique colorée, une image CT, et une image d'étiquettes.

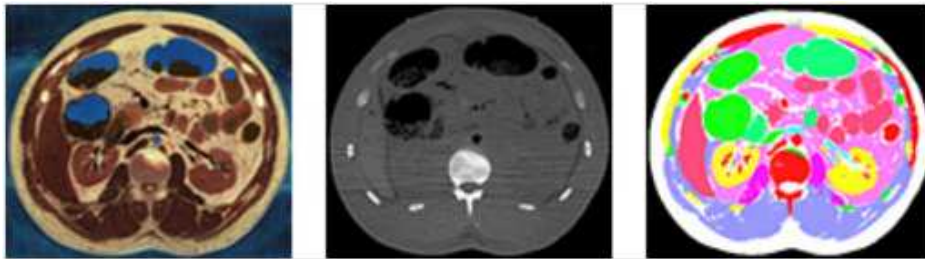


FIGURE 1.3: Un échantillon de tranches du *Voxel-Man*. De gauche à droite : image colorée, image de tomodensitométrie (CT-scan), et l'image d'étiquettes où chaque couleur correspond à une étiquette.

Les étiquettes indiquent à quel objet chaque voxel appartient. Parmi plusieurs objets qui ont été segmentés, citons : l'aorte abdominale, les vertèbres cervicales C5, C6 et C7, le rein droit. La Figure 1.4 montre une des représentations 3D obtenues de ces images.

Malgré le fait que cette approche offre une meilleure visualisation ainsi que la possibilité de navigation, ces atlas sont conçus pour l'utilisation par un humain et ne sont pas utilisables par des logiciels. De plus les atlas sont construits à partir d'un nombre limité de sujets (le projet de *Visible Human* correspond à deux individus, un homme et une femme), ce que ne permet pas de prendre en compte les variations (de forme, taille, etc.) entre les individus.

1.3 Atlas probabilistes

L'idée principale de cette approche est de prendre un ou plusieurs sujets (typiques) et de faire une étude statistique pour obtenir une représentation 3D de la structure d'intérêt. Les atlas ainsi construits sont surtout utiles pour la segmentation d'images médicales.

Prenons comme exemple un atlas de l'arbre vasculaire du cerveau. Dans [Chillet03], les auteurs créent un atlas à partir d'images angiographiques. Les vaisseaux sont d'abord extraits par segmentation, puis une carte de distance inverse est calculée à partir de chaque image binaire obtenue par segmentation. Finalement, l'une de ces cartes est choisie comme référence, et les autres sont recalées de manière affine sur cette image de référence. Les images de moyenne et de variance obtenues forment l'atlas vasculaire.

4. http://www.nlm.nih.gov/research/visible/visible_human.html

6. <http://www.voxel-man.de/3d-navigator/>

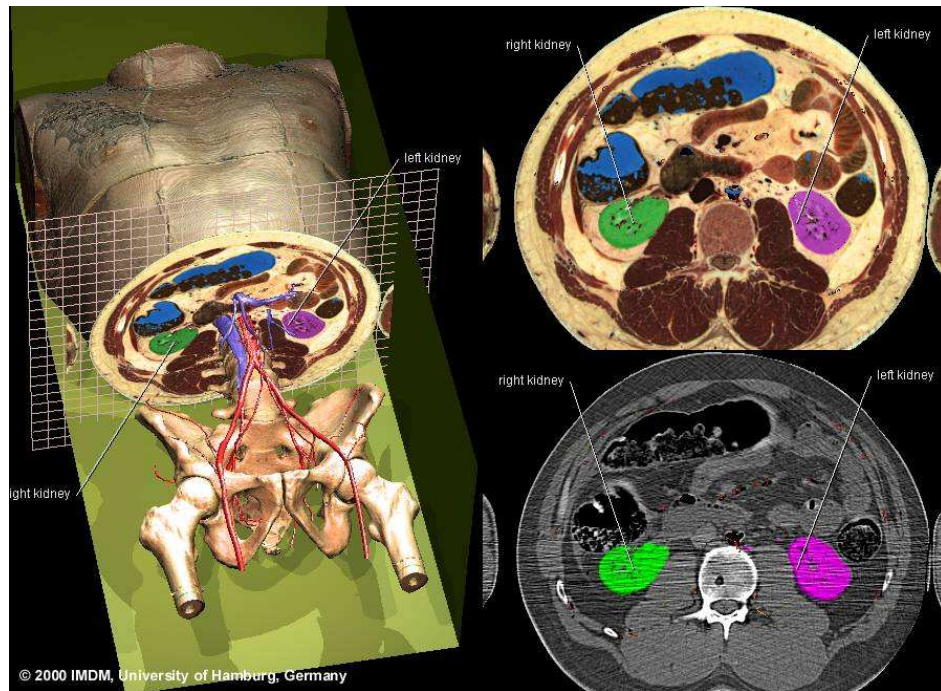


FIGURE 1.4: Voxal-Man⁶ : un exemple d'atlas informatisé. Images illustrant le bassin en 3D. Le rein droit est affiché en vert, le rein gauche en violet, et le colon droit et le colon transverse en bleu.

Dans la thèse de N. Passat [Passat05], une autre méthode pour créer un atlas vasculaire cérébral est proposée. Cette méthode tire bénéfice de deux images : une image de phase et une image de magnitude fournies par la technique d'imagerie (contraste-phase IRM). Comme la méthode précédente, d'abord un ensemble d'images est segmenté puis recalé sur une image de référence. Le plus de cette méthode est l'analyse quantitative. Cette analyse associe à chaque image deux images supplémentaires, l'une représentant le diamètre des vaisseaux (voir Figure 1.5), et l'autre leur orientation. Finalement, l'atlas

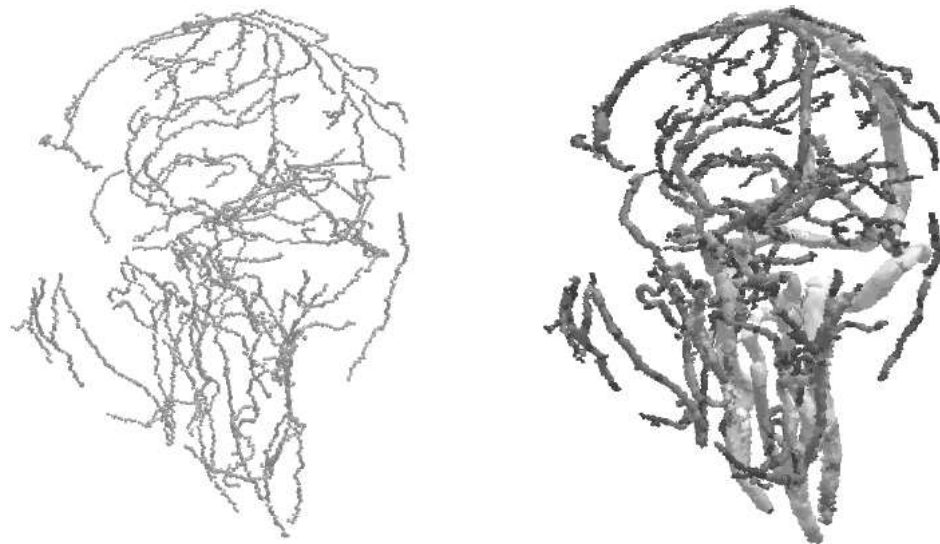


FIGURE 1.5: Informations obtenues lors de l'analyse quantitative. À gauche : squelette du réseau vasculaire cérébral permettant d'évaluer les axes médians des vaisseaux puis leur orientation. À droite : épaisseur des vaisseaux représentée en niveaux de gris. Figure issue de [Passat05].

comprend deux parties, la première (modélisant la densité vasculaire) est définie comme le taux de présence d'un vaisseau en chaque point de l'image (la probabilité que ce point appartienne au vaisseau). La seconde (modélisant le diamètre des vaisseaux) est définie en évaluant, en chaque point, la moyenne et l'écart-type du diamètre des vaisseaux dans les images.

Toutes les représentations basées image partagent une limitation. Cette limitation est qu'une grande partie des connaissances *a priori* (adjacence, forme, taille, ...) n'existe qu'implicitement.

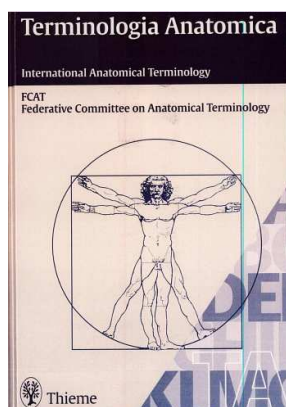
2 Approche basée concepts

Dans cette approche on cherche à modéliser les connaissances anatomiques que l'on ne trouve pas dans l'approche basée image, ou qui sont présentes implicitement. De plus, les connaissances présentées dans cette approche sont générales ; elles ne reposent ni sur un nombre limité de sujets ni sur une technique d'imagerie précise.

Cependant, des réalisations de cette approche peuvent contenir des images, qui restent un moyen d'illustration très important pour montrer certains attributs des structures anatomiques comme leur forme.

2.1 Livres de terminologie

Historiquement, une des premières représentations des connaissances *a priori* basée concepts est *Nomina Anatomica* [nominaAnatmica]. Cette publication, dont la première édition remonte à 1895, avait pour but de fournir un vocabulaire standard et mondial pour les sciences de la santé. Nous y trouvons les termes anatomiques, en anglais et en latin, organisés par région et par système anatomique.



Anatomia systemica	Systemic anatomy
A02.0.00.000 Ossa; Systema skeletale	Bones; skeletal system
Nomina generalia	General terms
A02.0.00.001 Pars ossea	Bony part
A02.0.00.002 Substantia corticalis	Cortical bone
A02.0.00.003 Substantia compacta	Compact bone
A02.0.00.004 Substantia spongiosa; Substantia trabecularis	Spongy bone; Trabecular bone
A02.0.00.005 Pars cartilaginea	Cartilaginous part
A02.0.00.006 Pars membranacea	Membranous part
A02.0.00.007 Periosteum	Periosteum
A02.0.00.008 Perichondrium	Perichondrium
A02.0.00.009 Skeleton axiale	Axial skeleton
A02.0.00.010 Skeleton appendiculare	Appendicular skeleton
A02.0.00.011 Os longum	Long bone
A02.0.00.012 Os breve	Short bone
A02.0.00.013 Os planum	Flat bone
A02.0.00.014 Os irregulare	Irregular bone
A02.0.00.015 Os pneumatizatum	Pneumatized bone
A02.0.00.016 Os sesamoideum	Sesamoid bone
A02.0.00.017 Diaphysis	Diaphysis
A02.0.00.018 Epiphysis	Epiphysis
A02.0.00.019 Cartilago epiphysialis	Epiphysal cartilage
A02.0.00.020 Lamina epiphysialis	Epiphysal plate; Growth plate
A02.0.00.021 Linea epiphysialis	Epiphysal line
A02.0.00.022 Metaphysis	Metaphysis
A02.0.00.023 Apophysis	Apophysis
A02.0.00.024 Tuber	Tuber; Tuberosity
A02.0.00.025 Tuberculum	Tubercle

FIGURE 1.6: *Terminologia Anatomica*. À droite : une page du livre illustrant l'organisation hiérarchique des titres.

Un successeur de *Nomina Anatomica* est *Terminologia Anatomica*, voir Figure 1.6. C'est une publication du *Federative Committee on Anatomical Terminology* et de l'association nationale des

anatomistes qui date de 1998. Les termes sont réorganisés sous forme hiérarchique. En effet, cette terminologie est constituée d'une hiérarchie de titres, de codes alphanumériques et d'indentations. Chaque élément de la hiérarchie comprend le ou les termes associé(s) en latin et en anglais. La position d'un terme dans la hiérarchie et son code indiquent ses relations avec les termes parents et frères [Dameron03].

Ces représentations terminologiques de l'anatomie sont un grand effort vers la formalisation des termes anatomiques. Néanmoins, le domaine anatomique et ses applications sont très vastes. Il est donc très difficile de déterminer le niveau de détail nécessaire à une application. De plus, les informations que l'on retire de la hiérarchie des termes ne suffisent pas à identifier les différentes structures anatomiques représentées par ces termes.

2.2 Unified Medical Language System (UMLS)

Le but de ce système [Lindberg93], développé par la librairie nationale de la médecine (*National Library of Medicine*), est de faciliter le développement des systèmes informatiques qui "comprennent" le langage de la biomédecine et de la santé⁷. L'UMLS contient trois sources d'information :

1. le métathésaurus (*Metathesaurus*) : il s'agit d'une grande base de données multilingue, et multi-usage. Elle contient des informations sur des concepts de la biomédecine et de la santé, leurs différents noms, et les relations entre ces concepts ;
2. le réseau sémantique (*Semantic network*) : il est composé de :
 - (a) types sémantiques : ces types sont une catégorisation de tous les concepts existant dans le métathésaurus. Parmi les types majeurs citons : organismes, structures anatomiques, fonction biologique, événement, objet physique et concept ;
 - (b) relations sémantiques : il s'agit des relations existant entre les types sémantiques. La relation la plus importante est "is a". Cette relation fournit la hiérarchie entre les types sémantiques du réseau. D'autres relations non hiérarchiques existent, elles sont regroupées en cinq catégories : "physically related to", "spatially related to", "temporally related to", "functionally related to" et "conceptually related to" ;
3. le lexique du SPECIALIST (*SPECIALIST lexicon*) : cette partie est développée pour fournir des informations lexicales nécessaires pour le "SPECIALIST Natural Language Processing System"⁸. Il comprend un lexique anglais contenant beaucoup de termes biomédicaux couvrant à la fois les mots anglais utilisés fréquemment et le vocabulaire biomédical.

L'UMLS n'est pas destiné à l'utilisateur final (*end user*), cependant ses ressources sont largement utilisées dans une variété d'applications. Par exemple, Joubert *et al.* [Joubert98], dans le cadre du projet ARIANE, fournit une interface qui permet de construire puis de traiter des requêtes. Pour cela, un modèle conceptuel du métathésaurus et du réseau sémantique de l'UMLS est utilisé, voir Figure 1.7.

7. <http://www.nlm.nih.gov/research/umls/>

8. <http://lexsrv3.nlm.nih.gov/Specialist/Home/index.html>

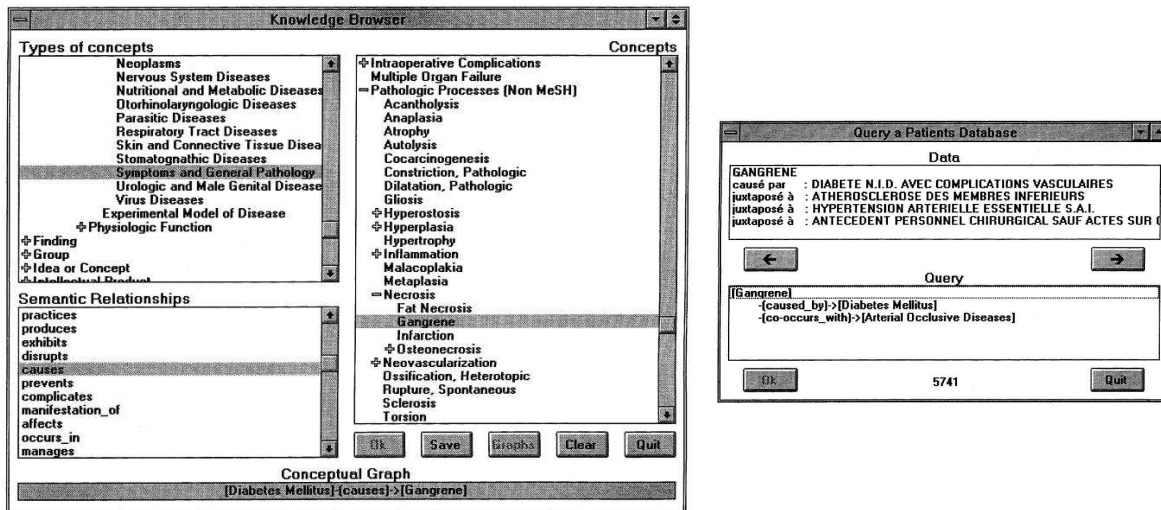


FIGURE 1.7: Figure de [Joubert98]. À gauche : la construction d’une requête conceptuelle à partir des connaissances sémantiques et des contextes. À droite : une requête et le registre qui la satisfait.

2.3 Ontologies

2.3.1 Définitions

L’ontologie ne possède pas une définition unique valide dans tous les domaines d’application. En philosophie, l’ontologie est l’étude de la nature de l’existence ou de la réalité telle quelle. En informatique plusieurs définitions existent dans la littérature.

Définition1 [Neches91] : une ontologie définit les termes de base et les relations constituant le vocabulaire d’un domaine, ainsi que les règles pour regrouper les termes et les relations pour définir des extensions du vocabulaire.

Définition2 [Gruber93] : une ontologie est une spécification explicite d’une conceptualisation. Cette définition est proche de celle utilisée en philosophie, car ce qui est représenté est ce qui “existe”. Plus tard, des adjectifs comme “formelle” et “partagée” ont été ajoutés à cette définition.

Définition3 [Guarino95] : une ontologie est un accord sur une conceptualisation partagée et éventuellement partielle. Des explications sur les mots utilisés dans cette définition et la précédente ont été données par Studer *et al.* [Studer98] :

Conceptualisation est un modèle abstrait d’un certain phénomène du monde (domaine étudié). Ce modèle est obtenu en retrouvant les concepts liés au phénomène en question.

Explicite veut dire que le type des concepts utilisés ainsi que les contraintes de leur utilisation sont explicitement définis.

Formelle veut dire que l’ontologie doit être lisible par une machine (*machine-readable*).

Partagée signifie qu’une ontologie est censée capter de la connaissance consensuelle (*consensual knowledge*), c’est-à-dire la connaissance acceptée par un groupe.

Définition4 [Dameron03] : Une ontologie *formelle* est un développement systématique, formel et axiomatique de la logique de toutes les formes et modes d’existence.

Récemment, Neuhaus *et al.* [Neuhaus08] considèrent l’ontologie comme “un vocabulaire contrôlé explicable par ordinateur”. L’ontologie ne regroupe donc pas une simple liste de termes, mais aussi le contenu sémantique associé à ces termes. Plus spécifiquement, ils distinguent deux types de termes :

ceux désignant les types et ceux désignant les relations. Ainsi, dans une ontologie anatomique le terme “bassin féminin” signifie le **type** *Bassin Féminin*, alors que le terme “est_une_partie_de” signifie la **relation** de *partition*.

Dans le domaine de la biomédecine, les types (connus aussi sous les noms ‘universaux’ ou ‘sortes’) sont les modèles réels que les scientifiques décrivent dans leurs théories. Le contenu sémantique d’une entité est expliqué à travers les relations qui lient cette entité à d’autres. Par exemple, le fait que l’entité “Os coxal” soit liée à l’entité “Os plat” par la relation “est_un” donne à l’ordinateur une information sémantique importante.

Pour comprendre les relations, il faut différencier :

- les types (universaux) : ils peuvent avoir plusieurs instances à un moment donné. Exemple : foie ;
- les individus (instances) : qui existent à chaque moment dans une localisation spatiale unique. Un individu est :
 - matériel : il a une masse. Exemple : mon foie, ton cerveau,
 - immatériel : sans masse. Exemple : cavité de mon estomac.

Normalement, les ontologies s’intéressent aux types. Ainsi la déclaration :

Bassin Féminin est_une_partie_de Corps

signifie que la relation “est_une_partie_de” existe entre le type “Bassin Féminin” et le type “Corps”. Cependant, les anatomistes sont obligés de passer par les instances pour étudier les types. Donc, la façon d’évaluer la justesse d’une déclaration est d’étudier les instances des types concernés.

2.3.2 Relations

La représentation des relations spatiales entre les parties du corps humain est primordiale dans une ontologie anatomique. Ces relations ont été étudiées de manière approfondie par Bittner *et al.* [Bittner08], et ont été classées dans les catégories suivantes : relations méréologiques, relations d’emplacement et relations topologiques.

Relations méréologiques Une méréologie est une théorie formelle traitant la relation entre la partie et le tout. La relation “est_une_partie_de” (symbolisée par P) s’applique entre les individus concrets et ne requiert ni de données quantitatives, ni d’abstraction mathématique. Par exemple, mon cœur est une partie de mon corps, mon doigt est une partie de ma main. La méréologie est une base de raisonnement qualitatif sur les relations spatiales dans une ontologie. Des relations comme “se_chevauchant_avec”, et “distinct_de” sont définies à travers P .

La relation P est une relation primitive [Simons87], c’est-à-dire qu’au lieu de la définir, ses attributs logiques sont décrits à travers des axiomes. Les axiomes suivants sont valables dans presque toute méréologie :

- (A1) Pxx
(chaque objet est une partie de lui – même)
- (A2) $Pxy \wedge Pyx \rightarrow x = y$
(si x est une partie de y et y est une partie de x , alors x et y sont identiques)
- (A3) $Pxy \wedge Pyz \rightarrow Pxz$
(si x est une partie de y et y est une partie de z , alors x est une partie de z)

D'autres relations : "Partie_propre" (PP), "Se_chevauchant_avec" (O) et "Distinct_de" (DS), sont définies à travers P :

Un individu x est une partie propre de l'individu y si x est une partie de y sans être y lui-même.

$$(PP) \quad PPxy \equiv Pxy \wedge x \neq y$$

Un individu x se chevauche avec l'individu y s'il existe un objet z qui est une partie de x et de y .

$$(O) \quad Oxy \equiv (\exists z)(Pzx \wedge Pzy)$$

Un individu x est distinct de l'individu y si x ne se chevauche pas avec y .

$$(DS) \quad DSxy \equiv \neg Oxy$$

Relations d'emplacement L'emplacement relatif entre deux objets peut parfois être déduit de leurs relations méréologiques. Si x est une partie de y , alors x est localisé dans y . En d'autres mots, l'emplacement de x est inclus dans l'emplacement de y . Cependant, une relation d'emplacement peut exister entre deux objets sans que l'un soit une partie de l'autre. Les relations méréologiques ne sont donc pas suffisantes pour décrire les relations d'emplacement. De plus, les entités immatérielles (cavités, passages, etc.) contiennent d'autres organes (entités) ne partageant aucune partie avec elles.

Une fonction de région [Donnelly06] est utilisée pour définir les relations d'emplacement. Cette fonction r associe à chaque individu la région spatiale unique qu'il occupe à un moment donné. Parmi les axiomes de cette fonction notons :

$$(A4) \quad Pxy \rightarrow Pr(x)r(y)$$

(si x est une partie de y alors la région de x est une partie de la région de y)

$$(A5) \quad r(r(x)) = r(x)$$

(la région de la région de x est la région de x)

Avec la fonction de région et les relations méréologiques, plusieurs relations d'emplacement sont définies :

Un individu x est localisé dans l'individu y si la région de x est une partie de la région de y .

$$LocIn(x, y) \equiv Pr(x)r(y)$$

Les individus x et y coïncident si les régions de x et de y se chevauchent.

$$PCoin(x, y) \equiv Or(x)r(y)$$

Relations topologiques (connexité et convexité)

1. Relation de connexité : intuitivement, la relation de connexité C existe entre deux individus x et y si la distance entre x et y est nulle. La distance entre deux objets est la plus grande limite inférieure de la distance minimale entre n'importe quel point du premier objet et n'importe quel point du second. Donc x est connecté à y s'ils se chevauchent, ou s'il existe un contact externe entre eux. Deux régions sont connectées si elles partagent au moins un point. Les relations de connexion externe et de séparation sont définies à travers C :

Deux individus x et y ont une connexion externe si et seulement s'ils sont connectés et s'ils ne coïncident pas partiellement.

$$EC(x, y) \equiv C(x, y) \wedge \neg PCoin(x, y)$$

Deux individus x et y sont séparés si et seulement s'ils ne sont pas connectés.

$$SP(x,y) \equiv \neg C(x,y)$$

Les axiomes de cette relation sont les suivants :

$$(A6) \ Cxx$$

(tout objet est connexe avec lui – même)

$$(A7) \ Cxy \rightarrow Cyx$$

(si x est connecté à y , alors y est connecté à x)

$$(A8) \ LocIn(x,y) \rightarrow (\forall z)(Cxz \rightarrow Cyz)$$

(si x est localisé dans y , alors tout objet connecté à x est connecté à y)

2. Relations dépendant de la convexité : pour pouvoir décrire davantage de relations entre les individus, la fonction d'enveloppe convexe ch est ajoutée. Cette fonction associe à chaque individu son enveloppe convexe. Quelques propriétés importantes de cette fonction sont présentées par les axiomes suivants :

$$(A9) \ Pr(x)ch(x)$$

(la région de x est une partie de son enveloppe convexe)

$$(A10) \ LocIn(x,y) \rightarrow Pch(x)ch(y)$$

(si x est localisé dans y , alors l'enveloppe convexe de x est une partie de celle de y)

$$(A11) \ ch(ch(x)) = ch(x)$$

(l'enveloppe convexe de l'enveloppe convexe de x est l'enveloppe convexe de x)

Cette fonction est ensuite utilisée pour définir certaines relations :

Un individu x est entouré par l'individu y si et seulement si la région de x est une partie de l'enveloppe convexe de y , et les régions de x et de y ne se chevauchent pas, voir Figure 1.8 (a) et (b).

$$SurrBy(x,y) \equiv Pr(x)ch(y) \wedge \neg Or(x)r(y)$$

Un individu x est partiellement entouré par l'individu y si et seulement si la région de x chevauche avec l'enveloppe convexe de y , et les régions de x et de y ne se chevauchent pas, voir Figure 1.8-(c).

$$PSurrBy(x,y) \equiv Or(x)ch(y) \wedge \neg Or(x)r(y)$$

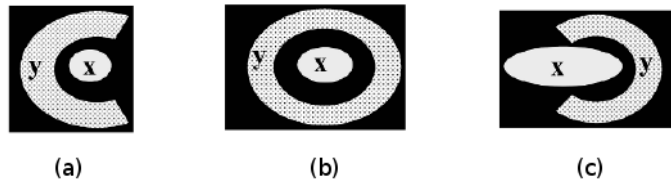


FIGURE 1.8: (a) et (b) : x est entouré par y , (c) : x est partiellement entouré par y . Figure issue de [Bittner08].

Toutes les relations spatiales précédentes sont définies entre des individus, l'extension de ces relations vers les types est faite en prenant en compte la relation d'instantiation. Un individu est

l'instance d'un type, et les relations entre individus ne se propagent pas toujours entre les types correspondants. Dans [Beck03], la logique de description est utilisée pour décrire différentes versions de la relation entre le tout et la partie au niveau des types. Pour une relation R (partie de, chevauchant avec, etc.), Bittner *et al.* distinguent trois relations au niveau des types :

$$\begin{aligned} R_1(A, B) &\equiv (\forall x)(Inst(x, A) \rightarrow (\exists y)(Inst(y, B) \wedge Rxy)) \\ R_2(A, B) &\equiv (\forall y)(Inst(y, B) \rightarrow (\exists x)(Inst(x, A) \wedge Rxy)) \\ R_{12}(A, B) &\equiv R_1(A, B) \wedge R_2(A, B) \end{aligned}$$

Le type de relation R_1 pose des restrictions sur toutes les instances du premier type A , car la relation doit être valable pour **toute** instance de A . Par contre, le type de relation R_2 pose des restrictions sur toutes les instances du second type B . Le type de relation R_{12} est le plus restreint car il pose des restrictions sur les deux types. Prenons comme exemple la relation de propre part PP :

- entre les deux types suivants : le système reproductif féminin humain et le corps humain, on peut seulement mettre le type de relation R_1 , c'est-à-dire $PP_1(système\ reproductif\ féminin\ humain, le\ corps\ humain)$ car tout système reproductif féminin humain est une partie propre d'un corps humain. Le contraire n'est pas vrai ;
- entre les deux types suivants : cavité et cœur, on peut seulement mettre le type de relation R_2 , c'est-à-dire $PP_2(cavité, cœur)$ car tout cœur a comme partie propre une cavité, mais le contraire n'est pas vrai ;
- entre les deux types suivants : système nerveux humain et corps humain, on peut mettre le type de relation R_{12} , c'est-à-dire $PP_{12}(système\ nerveux\ humain, corps\ humain)$ car tout système nerveux humain est une partie propre d'un corps humain et tout corps humain a comme partie propre un système nerveux.

D'autres relations au niveau des types sont proposées dans [Smith05], par exemple la relation “adjacence”, “partie propre tangentielle” et “partie propre non tangentielle”, voir Figure 1.9.



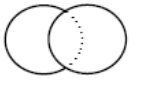
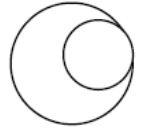
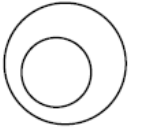
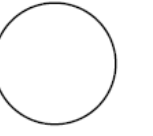
		
séparation	adjacence	chevauchement
		
partie propre tangentielle	partie propre non tangentielle	identité

FIGURE 1.9: Diverses relations spatiales. Figure issue de [Smith05].

Les relations que nous avons présentées dans cette section et les axiomes qui les définissent ou qui présentent leurs attributs sont très importants pour le raisonnement. À partir de ces axiomes, et en utilisant des moteurs d'inférence, plus d'informations peuvent être retirées de l'ontologie. Nous reviendrons sur cette utilisation plus loin (voir Section 3).

Des langages spécifiques sont utilisés pour la création des ontologies. Ces langages sont classés en deux catégories :

- les langages traditionnels : Ontolingua [Farquhar96], *Frame Logic (FLogic)* [Kifer95], LOOM [MacGregor9] etc. ;
- les langages de balisage (*Markup*) : XML [Bray98], RDF [Lassila99], RDFS [Brickly99], SHOE [Luke00], OWL [Dean02], etc. La Figure 1.10 montre un sous-ensemble de ces langages et les relations entre ceux-ci.

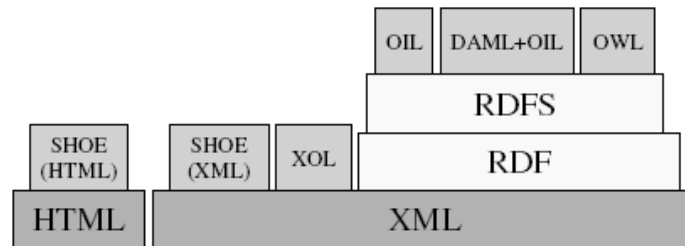


FIGURE 1.10: L'ensemble des langages de balisage d'ontologies, figure issue de [Corcho03].

Détailler ces langages et leurs différences sort du cadre de cette thèse. Cependant, le lecteur intéressé pourra se référer à [Corcho00, Corcho03] pour plus d'informations.

2.3.3 Quelques ontologies de la littérature

Le projet Ontologie Basique Formelle (*Basic Formal Ontology*- BFO)⁹ a commencé en 2002. La théorie derrière BFO a été initialement développée par Smith *et al.* [Smith04, Grenon04]. BFO vient d'une orientation philosophique se chevauchant avec celles de *DOLCE*¹⁰ et de *SUMO*¹¹. Elle fournit une ontologie authentique qui peut servir comme base pour des ontologies spécifiques à certains domaines développées pour la recherche scientifique, comme pour le domaine de la biomédecine. Ainsi, BFO ne contient pas de termes comme : physique, chimique, ou d'autres termes qui sont propres à un certain domaine.

Cette ontologie avec sa "philosophie" de réalité fait des distinctions importantes entre quatre paire d'entités [Rosse07] :

1. la réalité et la connaissance : ce qui existe dans le monde réel et ce qui est décrit dans les livres ;
2. les instances et les universels : les instances existent discrètement comme des individus (ma main), alors que les universels (types) peuvent avoir plusieurs instances à un moment donné ;
3. les continnants et les *processus* : le continuant est une entité qui existe comme un tout (*in toto*) bien qu'elle subisse des changements tout au long de son existence. Le processus ou l'occurent est une entité qui n'endure pas le temps, elle passe par plusieurs phases temporelles successives durant son existence ;
4. les dépendants et les indépendants : le dépendant est une entité qui ne peut pas exister sans d'autres entités. Par exemple, les processus dépendent des entités qui y participent. Des dépendants continnants existent aussi, le lumen de l'œsophage n'existe que si l'œsophage existe.

La fonderie ouverte des ontologies biologiques et biomédicales (*Open Biomedical Ontologies*-OBO)¹² est une expérience collaborative concernant les développeurs d'ontologies. Le but est de

9. <http://www.ifomis.org/bfo>

10. <http://www.loa-cnr.it/DOLCE.html>

11. <http://www.ontologyportal.org/>

12. <http://www.obofoundry.org>

créer un ensemble de principes de développement d'ontologies, et de fournir des ontologies de référence dans le domaine biomédical. Beaucoup d'ontologies anatomiques se trouvent dans cette fonderie, nous prenons quelques exemples, certains proviennent d'OBO (FMA, CARO).

Generalized Architecture for Languages Encyclopedias and Nomenclature in Medicine (GALEN) [Rector94, Rogers01] : GALEN est un projet financé par la communauté européenne, son but est de fournir une représentation des concepts médicaux que les applications cliniques puissent stocker, manipuler et partager [Dameron03]. GALEN est basé sur un modèle de référence commun (CORE) pour les termes médicaux, et le langage GRAIL (*GALEN Representation And Integration Language*) [Rector93, Rector94]. Ce langage est un langage de subsumption¹³. Un modèle de GRAIL consiste en une hiérarchie de subsumptions des entités de base, et un ensemble de clauses (*statement*) liant ces entités.

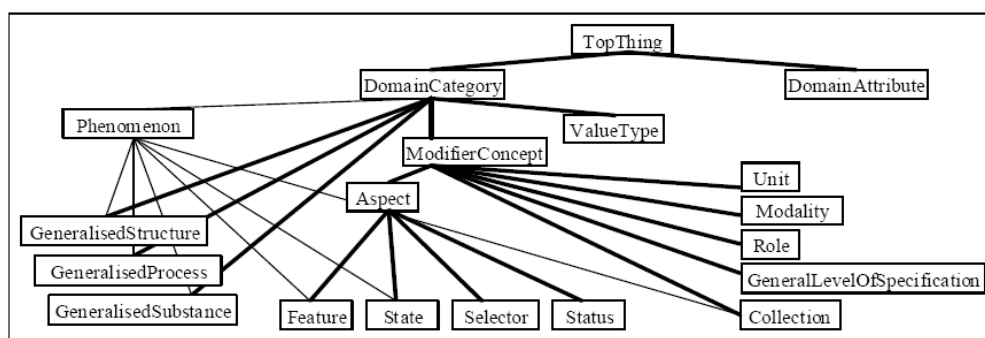


FIGURE 1.11: GALEN : l'espace de catégories de haut niveau du modèle CORE [Rector96].

L'ontologie GALEN comprend plusieurs catégories, voir Figure 1.11. Ces catégories sont [Rector96] :

- les structures généralisées *GeneralizedStructures* : ce sont des choses abstraites ou physiques ayant des parties et indépendantes du temps ;
- les substances généralisées *GeneralizedSubstances* : ce sont des choses continues, abstraites ou physiques et indépendantes du temps ;
- les processus généralisés *GeneralizedProcesses* : les changements qui arrivent dans le temps ;
- les modificateurs *Modifiers* : les principaux sont les aspects qui raffinent le sens d'une catégorie (localisation, gravité, forme, ...) et les modalités et collections (historique familial, risque) dont la signification vient de la catégorie, mais qui sont en même temps très différents. Par exemple, l'"historique familial de diabète" n'est pas une sorte de "diabète". D'autres types de modificateurs sont illustrés sur la Figure 1.11.

Une structure secondaire est mise au-dessus des quatre structures principales, il s'agit du phénomène *Phenomenon*. Le but de cette structure est d'exprimer les maladies ou les désordres. Ainsi une maladie dans GALEN est définie comme (notation en GRAIL) :

Phenomenon which hasPathologicalStatus Pathological

La Figure 1.12 montre un tableau de [Rector96] illustrant la taxonomie des catégories majeures avec quelques exemples.

L'ontologie GALEN est disponible en ligne¹⁴ en plusieurs langages : GRAIL, OWL, RDF.

Foundational Model of Anatomy (FMA) [Rosse95, Brinkly99, Rosse07] : c'est une ontologie

13. La subsumption désigne une relation hiérarchique entre des concepts, dans les logiques de description. Cette notion est proche de la relation « est impliqué par » en logique classique, ou encore « contient » en logique ensembliste (<http://fr.wikipedia.org/wiki/Subsumption>).

14. <http://www.opengalen.org/sources/sources.html>

Entité	Exemple	Entité	Exemple
<u>GeneralisedProcess</u>		<u>GeneralisedSubstance</u>	
SpecificProcess		Energy	Radiation, SoundEnergy
BiologicalProcess		Substance	
BodyProcess	Peristalsis, Breathing, Clotting	BodySubstance	
Behaviour	VolitionalAct, ClinicalAct	Tissue	MuscleTissue, BoneTissue
NonBiologicalProcess		NAMEDBodySubstance	Urine, Bile, Sputum etc.
PhysicalProcess	Irradiation	ChemicalSubstance	Drugs, Sodium etc.
ChemicalProcess	Histological Staining	NAMEDSubstance	Air, Wood
GenericProcess	Transport, Opening, Closing		
<u>GeneralisedStructure</u>		<u>ModifierConcept</u>	
AbstractStructure		Aspect	
PsychosocialConstruct	Clinic, Hospital	Feature	Sex, Chronicity, Shape, Malignancy
LogicalStructure	Protocol, Plan	State	male/female, acute/chronic, round/square
<u>PhysicalStructure</u>		Selector	leftSelector/rightSelector
LinearStructure	Displacement	Status	pathological/physiological normal/nonNormal
PlanarStructure	Triangle, Square	<u>Collection</u>	(Multiple which is\MultipleOf Spot).=Rash
SolidStructure		<u>Modality</u>	FamilyHistory, presence/absence
MicroscopicStructure	Cell, Microorganism	<u>Role</u>	DrugRole, HormoneRole, PatientRole
InertSolidStructure	Building, Device	<u>Unit</u>	second, metre, kilogram
OrganicSolidStructure		<u>LevelOfSpecification</u>	uniquelySpecified n
BodyStructure			
BodyPart	Heart, Leg, Head, Femur, AdrenalGland, Sacrum, PisiformBone, etc etc		
GenericBodyStructure	Cusp, Horn, Promontory, Artery, Lump, Bursa, Orifice, Rim, Ridge etc		

FIGURE 1.12: GALEN : exemples de [Rector96].

mature, elle a été créée et développée par le groupe *Structural Informatics Group (SIG)* de l'université de Washington.

Le but a d'abord été l'enrichissement de types sémantiques, concepts et relations d'UMLS, afin de représenter les concepts anatomiques sous un format compréhensible par un ordinateur [Rosse95].

L'ontologie FMA est une source de connaissances basée sur l'ordinateur pour l'informatique biomédicale. Elle s'intéresse à la représentation des classes ou types et des relations nécessaires pour représenter symboliquement la structure phénotypique du corps humain. Cette représentation est compréhensible par les humains et navigable, *parsable*, et interprétable par les systèmes basés-machine¹⁵.

L'ontologie FMA est fondamentale (*foundational*) pour deux raisons : (1) les nœuds de haut niveau de la taxonomie de FMA sont généralisables à tout vertébré, et selon plusieurs aspects aux métazoaires¹⁶ (2) les entités incluses dans FMA sont celles qui participent de manière saillante à tous les processus biologiques qui finissent par se manifester, comme la maladie ou la santé [Rosse07].

FMA adopte les distinctions faites par BFO, les continuants dépendants et indépendants doivent être donc organisés afin de représenter l'anatomie réelle.

De plus, FMA applique la règle d'héritage singulier (*single inheritance*), c'est-à-dire que si une classe (entité) hérite d'une seule classe (entité), elle prend alors toutes ses propriétés et diffère par un certain nombre de caractéristiques supplémentaires. Cette règle est nécessaire pour avoir une taxo-

15. <http://sig.biustr.washington.edu/projects/fm/AboutFM.html#whatIsTheFM>

16. Les métazoaires (Metazoa, du grec meta 'après' et zoön 'animal', par opposition aux protozoaires) sont des organismes eucaryotes pluricellulaires mobiles et hétérotrophes (<http://fr.wikipedia.org/wiki/Metazoa>).

nomie nette et précise mais elle est très problématique [Rosse07]. Par exemple, selon cette règle, le rein (entité “kidney”) doit être lié à une seule entité par la relation “is_a”, mais il a des fonctions multiples : excréter l’urine, et sécréter une hormone. Doit-il être classé comme organe excréteur ou comme organe endocrinien ? Pour cela, le choix du contexte selon lequel les entités seront classées est primordial. Ce choix dépend largement de ce que l’on veut dire par “anatomie” ou théorie de l’anatomie.

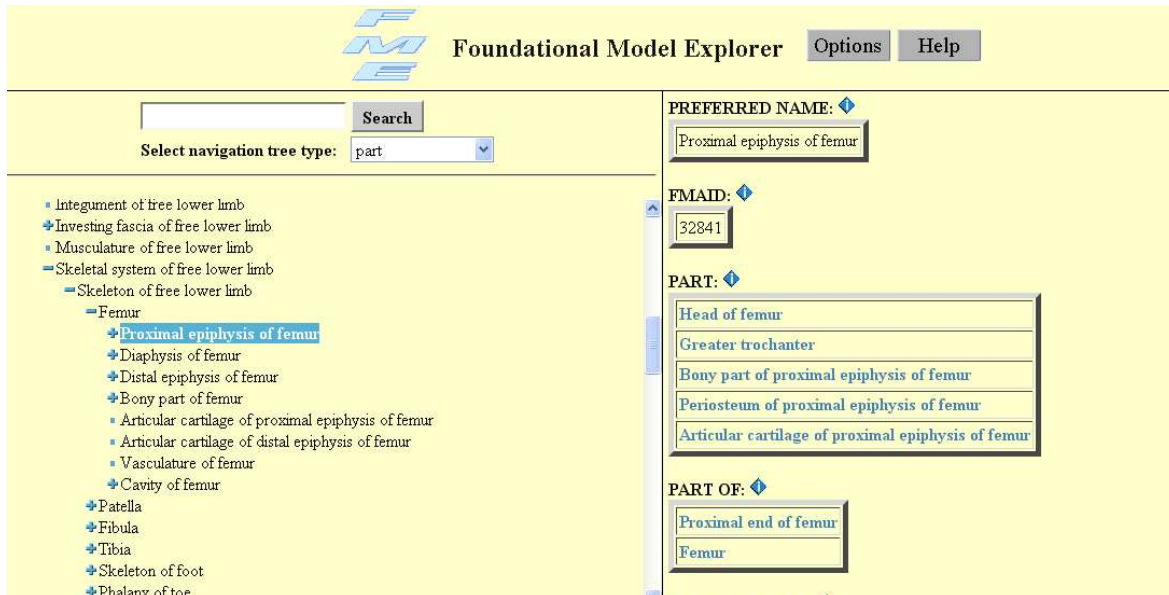


FIGURE 1.13: FMA.

En l’absence d’une théorie de l’anatomie connue, FMA adopte sa propre théorie de l’anatomie. Dans cette théorie, toute instance d’une structure anatomique est un continuant anatomique indépendant. La définition d’une structure anatomique est fondamentale dans cette théorie :

Une entité anatomique matérielle, qui est générée par une expression coordonnée des gènes (propres de l’organisme) guidant sa morphogenèse, a une forme 3D héritée, ses parties sont connectées et spatialement liées entre elles par des modèles déterminés par l’expression de gènes [Rosse07].

La propriété essentielle qui distingue les entités anatomiques d’objets matériels est la participation des gènes dans la génération des entités anatomiques. Les deux autres propriétés sont la forme 3D et l’arrangement des parties de la structure qui dépend lui aussi de gènes. Cette théorie conduit à une taxonomie de l’anatomie.

La taxonomie dans FMA a une racine unique et respecte la règle d’héritage singulier. Les types et les instances sont liés par la relation “is_a”. Selon la théorie de FMA, la structure est le contexte prédominant puisqu’elle est l’essence de l’anatomie. La Figure 1.14 montre les nœuds de haut niveau de la taxonomie de FMA.

- Anatomical entity
 - Physical anatomical entity
 - Material anatomical entity
 - + Anatomical structure
 - + Portion of body substance
 - + Anatomical set
 - + Immaterial anatomical entity
 - + Non-physical anatomical entity

FIGURE 1.14: Les nœuds de haut niveau de la taxonomie de FMA.

Les relations anatomiques sont très importantes pour la description des structures anatomiques. FMA adopte quelques relations proposées dans UMLS [Lindberg93] et GALEN. Cependant, la liste des relations utilisées continue à évoluer, non seulement en nombre de relations mais aussi en expres-

sivité de ces relations. Deux types de relations existent : les relations taxonomiques, et les relations structurelles. La Figure 1.15 montre une arborescence de ces relations.

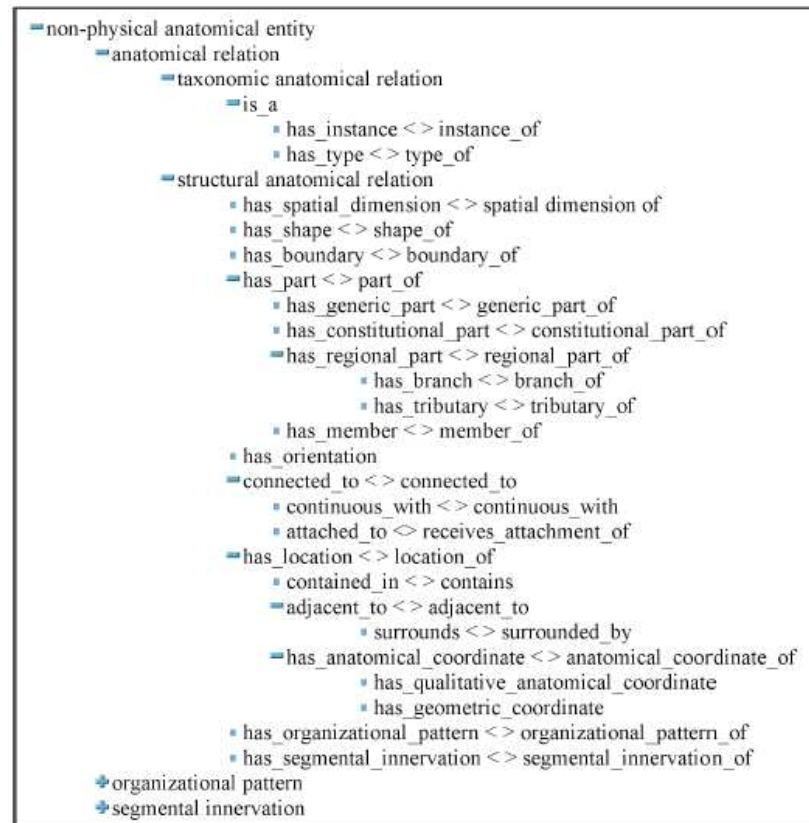


FIGURE 1.15: Relations dans FMA [Rosse07].

FMA a été implantée avec *Protégé* (un outil pour l'édition d'ontologie¹⁷). Les auteurs de FMA utilisent *Protégé* pour éditer les données stockées dans une base de données MySQL, cependant un accès moins complexe *via* un navigateur web est fourni (Figure 1.13). FMA est aussi disponible sous format OWL¹⁸ [Golbreich06].

L'ontologie FMA n'est pas développée pour une application spécifique, elle est une source pour développer des applications pour l'éducation, la recherche biomédicale et d'autres sous-domaines de la santé.

Common Anatomy Reference Ontology (CARO) [Haendel07] : le but de CARO est de faciliter l'interopérabilité entre les ontologies existantes de différentes espèces. Elle sera utilisée comme patron (*template*) pour construire de nouvelles ontologies anatomiques.

CARO est une ontologie d'anatomie commune, son cœur est constitué d'un unique schéma de classification structurelle. Elle a repris la politique d'héritage unique de FMA. La définition des termes de l'ontologie en respectant le principe "*genus-differentia*" peut garantir cette propriété. Selon ce principe, tout type est un raffinement (selon un *differentia*) d'un type existant plus général (*genus*, le parent dans la hiérarchie de la relation *is_a*). Si ce principe est suivi, nous obtenons des hiérarchies de

17. <http://protege.stanford.edu/>

18. <http://www.bioontology.org/wiki/index.php/FMAInOwl>

classification nettes où tout type a un parent (*is_a*) unique et des fils disjoints.

Les définitions et relations fournies par CARO concernent les anatomies canoniques. Une anatomie canonique donne la composition “prototypique” des membres d’une espèce. La définition d’anatomie canonique est assez problématique, car elle doit tolérer les variations naturelles dans l’espèce. En médecine, elle est la généralisation déduite des observations qualitatives qui sont admises et utilisées par les anatomistes.

CARO contient la définition de beaucoup de types anatomiques génériques (*cellule, organe complexe, système anatomique,...*) organisés suivant une hiérarchie (*is_a*). Cependant d’autres relations (comme *part_of*) seront ajoutées.

Une classification structurale ne suffit pas pour obtenir une représentation complète de l’anatomie. Par exemple, les composants du système immunitaire ne sont pas une partie d’un groupe ou d’une structure, donc ils ne peuvent pas être définis dans CARO.

Les types et les définitions dans CARO sont basés sur FMA. Les classifications structurales de FMA ont été considérées comme un bon point de départ, cependant il a fallu généraliser des types qui sont propres à l’anatomie humaine ou celle de mammifères pour qu’ils deviennent valables pour toute espèce. La Figure 1.16 montre une partie de la taxonomie de CARO.

CARO est disponible en ligne sur la fonderie OBO et sous format OWL¹⁹.

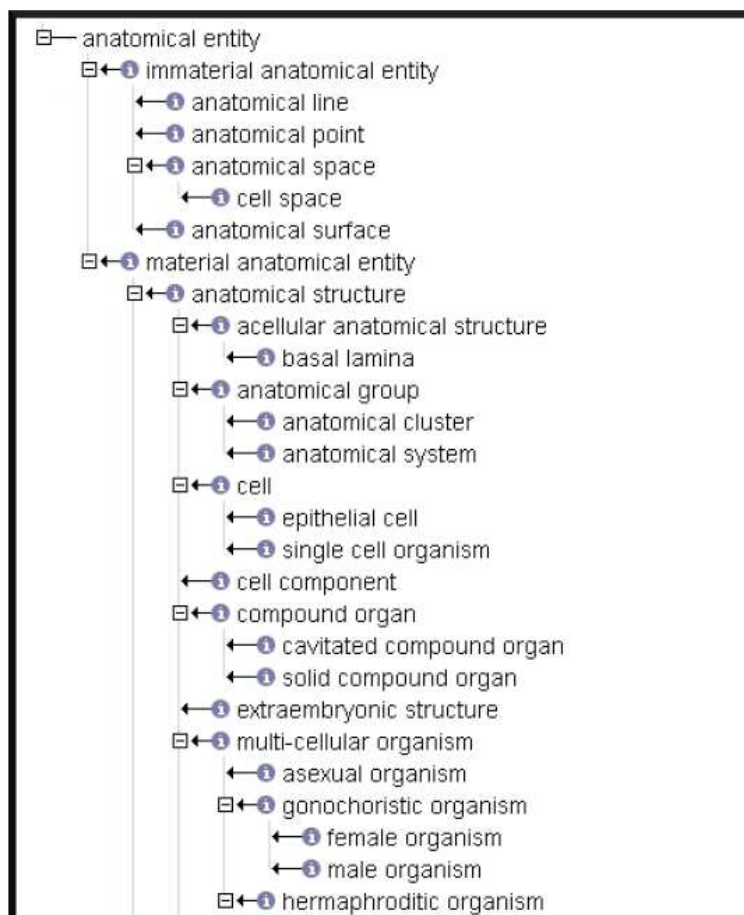


FIGURE 1.16: Une partie de la taxonomie de CARO [Haendel07].

MyCorporisFabrica [Palombi09] :

MyCorporisFabrica est une ontologie récente et toujours en cours de développement. C’est une base de données anatomique qui est basée sur FMA. La particularité de cette ontologie est le lien qu’elle crée entre une description canonique des entités anatomiques et des instances de ces entités qui viennent du monde réel, voir Figure 1.17.

En effet, MyCorporisFabrica étend l’ontologie FMA pour ajouter des données supplémentaires permettant la création des modèles géométriques et mécaniques. Le schéma de la base de données (Figure 1.17) montre que MyCorporisFabrica comprend deux parties.

19. <http://obofoundry.org/cgi-bin/detail.cgi?caro>

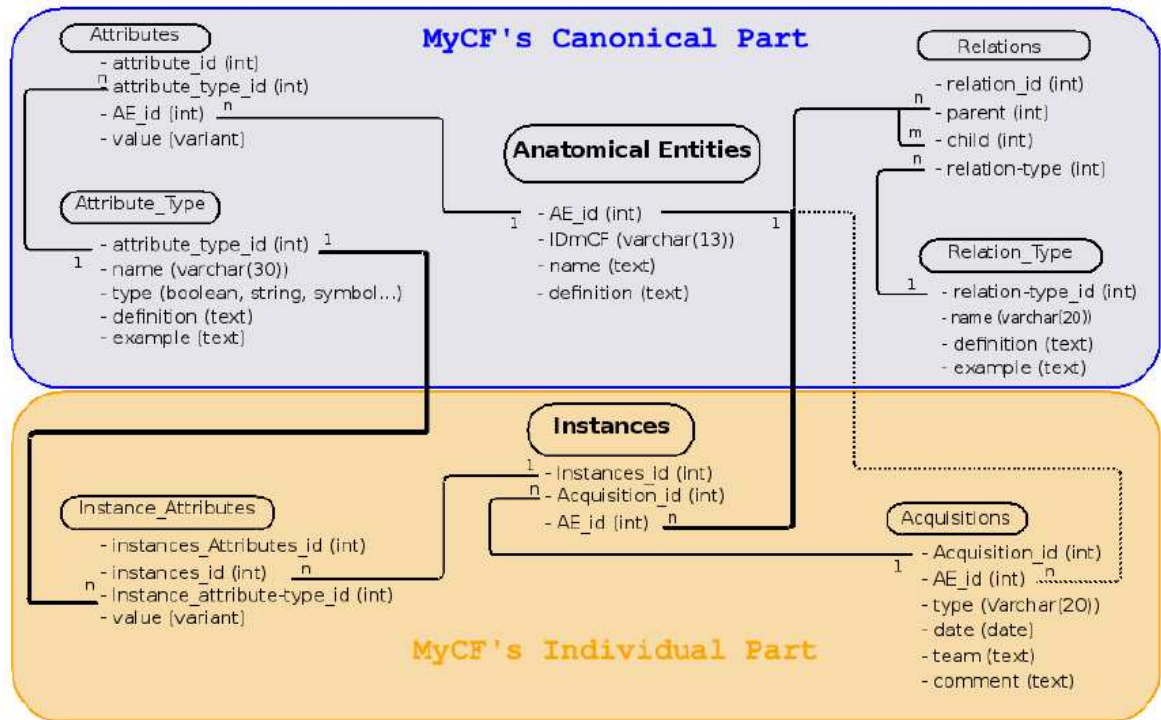


FIGURE 1.17: Le schéma MySQL de MyCorporisFabrica [Palombi09].

- Partie canonique : toutes les classes canoniques sont stockées dans le tableau *Anatomical Entities* avec un identifiant (*AE_id*) comme clé primaire. Les différents types de relations sont

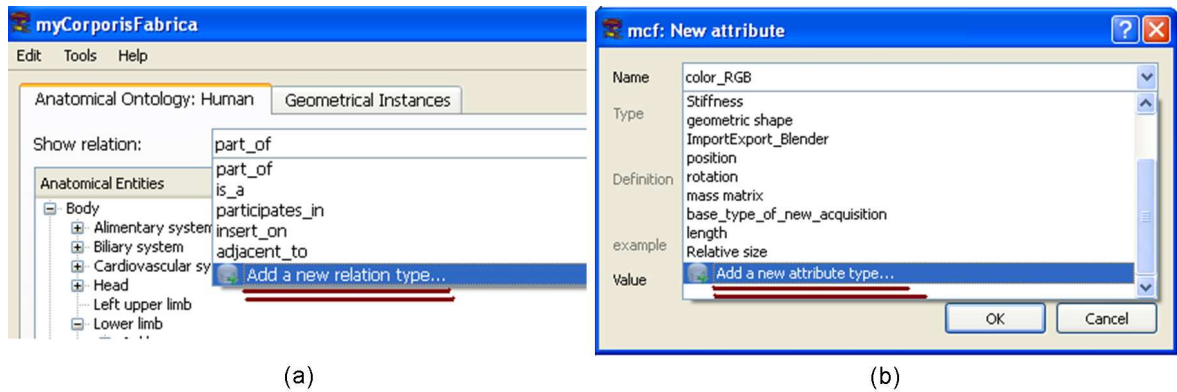


FIGURE 1.18: (a) Les relations actuelles de MyCorporisFabrica et la possibilité d'ajout de nouvelles relations. (b) Les attributs actuels de MyCorporisFabrica et la possibilité d'ajout de nouveaux attributs.

stockés dans *Relation_Type*, chacun pouvant produire une taxonomie ou plus généralement un graphe orienté. Les nœuds sont des entités du tableau *Anatomical Entities* et les arêtes du tableau *Relation*.

Le tableau *Relation_Type* donne les définitions des relations avec un exemple, alors que le tableau *Relation* stocke les relations entre un parent et un fils (des entités anatomiques).

De la même façon, le tableau *Attribut_type* contient les types d'attributs qu'une entité anatomique peut avoir, et le tableau *Attributes* contient les attributs des entités anatomiques. Chaque

ligne de ce tableau indique la valeur de l'attribut (identifié par *attribute_type_id*) que l'entité (identifiée par *AE_id*) possède.

Avec cette organisation, de nouvelles relations et de nouveaux attributs peuvent être ajoutés à l'infini à la base de MyCorporisFabrica, sans toucher à sa structure, voir Figure 1.18.

- **Partie individu** : cette partie contient les instances stockées dans le tableau *Instances* et liées aux entités de la partie canonique. La relation 1 : *n* montre qu'une entité anatomique peut avoir plusieurs instances. Les instances ont des attributs spécifiques, les types de ces attributs viennent du tableau *Attribut.Type* de la partie canonique, mais leurs valeurs sont stockées dans le tableau *Instance_Attributes* de la partie individu. Les valeurs de certains attributs peuvent être prises de la partie canonique si elles ne sont pas précisées pour l'instance. Cela permet de compléter par des informations générales un modèle spécifique au patient. Par exemple, dans une acquisition du genou, l'attribut qui est la couleur de l'os n'est pas fourni par l'utilisateur mais extrait de la partie canonique. Les données spécifiques au patient, des images médicales principalement, sont gérées dans le tableau *Acquisitions*.

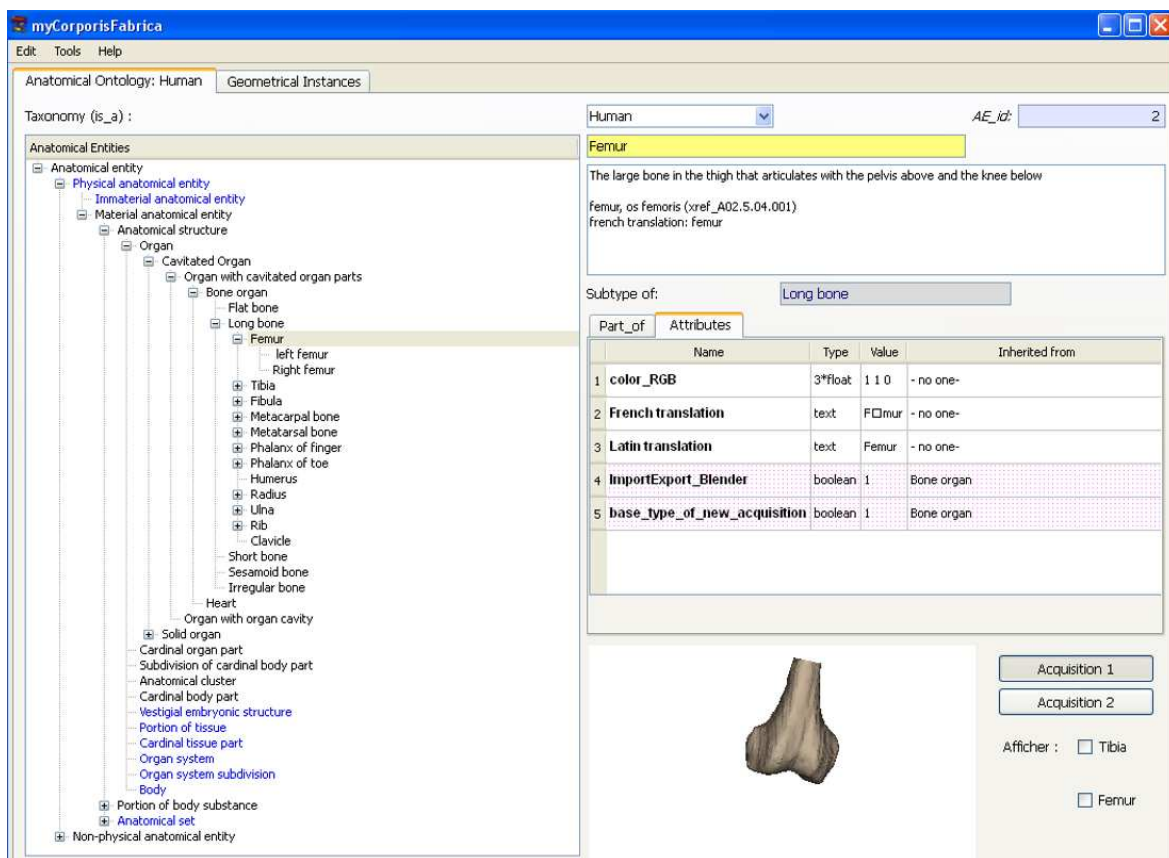


FIGURE 1.19: Le navigateur de MyCorporisFabrica. À gauche : la taxonomie (arbre) selon la relation 'is_a', à droite : en haut : des informations sur l'entité, au milieu : les attributs et les relations de l'entité, en bas : une visualisation en VTK des acquisitions contenant l'entité.

MyCorporisFabrica est disponible en ligne ²⁰. L'utilisateur peut installer la base de données écrite en MySQL, ou un navigateur pour le système Windows, voir Figure 1.19.

Ayant les droits de lecture/écriture, l'utilisateur peut faire un clic droit et choisir de voir/modifier une entité, c'est-à-dire ses attributs et ses relations, voir Figure 1.20.

20. <http://www.mycorporisfabrica.org/node/17>

The screenshot shows a software window titled "myCF - Edit AE". It contains fields for "Name" (Femur), "AE_id" (2), and a "Definition" box with text: "The large bone in the thigh that articulates with the pelvis above and the knee below", "femur; os femoris (xref_A02.5.04.001)", and "french translation: femur". Below these are "Source" and "Curator" fields. A tabbed interface shows "Attributes" and "Relations". The "Relations" tab contains a table with 7 rows of data. At the bottom are "Add Child", "Add Parent", and "Delete" buttons, along with "OK" and "Cancel" buttons.

	Child	Relation type	Parent
1	Femur	part_of	Thigh
2	Distal epiphysis of the femur	part_of	Femur
3	Femur	is_a	Long bone
4	left femur	is_a	Femur
5	Right femur	is_a	Femur
6	Diaphysis of the femur	part_of	Femur
7	Proximal epiphysis of the femur	part_of	Femur

FIGURE 1.20: L'interface permettant (avec les droits d'écriture) la modification des attributs et des relations de l'entité anatomique choisie.

3 Liaisons entre l'ontologie et la géométrie

En tant qu'informaticiens, nous nous intéressons aux représentations de connaissances *a priori* compréhensibles par la machine, et travaillant dans le domaine graphique ; les applications possibles de ces représentations dans ce domaine sont notre priorité.

En effet, différentes ontologies (celles que nous avons citées entre autres) ont servi de source d'information pour plusieurs applications. Les ontologies peuvent ajouter de l'information aux modèles géométriques issus de différentes sources et pour des buts différents. Nous avons choisi dans la littérature quelques exemples d'applications (pas toutes médicales) qui font un lien entre la géométrie sous forme de modèles 3D d'une part, et des connaissances *a priori* organisées sous forme d'ontologie d'autre part.

Parmi les applications médicales, certaines s'intéressent à l'anatomie du corps humain entier, alors que d'autres sont concentrées sur une partie précise du corps. Le but le plus fréquent de ces applications, en général, est d'identifier les organes visibles sur les images ou les sources numériques. Cette identification permet d'afficher les connaissances symboliques liées à ces organes et stockées dans les sources d'informations symboliques utilisées (des ontologies).

Cerveria *et al.* [Cerveria00] proposent une application concernant la totalité du corps humain. Ils utilisent trois bases de données pour intégrer d'une part les connaissances symboliques d'UMLS,

et d'autre part les images du VHD (*Visual Human Dataset*). La première base de données (*Anatomical Knowledge Database*) est relationnelle et contient les concepts issus d'UMLS et liés selon différentes relations : “*is-part-of*”, “*acts-as*”, “*has-function-of*”, etc. La deuxième base de données *Image-Database* contient des images colorées issues du VHD, où le contour de chaque structure anatomique est affiché. Ce contour est calculé grâce aux masques qui sont pris avec les images. Selon ces masques, chaque pixel contient un entier qui pointe vers une table de correspondance (*Look-Up Table*). Dans cette table, les codes de presque 1600 structures anatomiques ont été stockés. La mise en correspondance entre ces codes (en utilisant l'*Image-Database*) et la base de connaissances anatomiques (*Anatomical Knowledge Database*) produit les contours de différentes structures anatomiques. Le système a été implanté en utilisant la technologie client/serveur en Java, voir Figure 1.21.

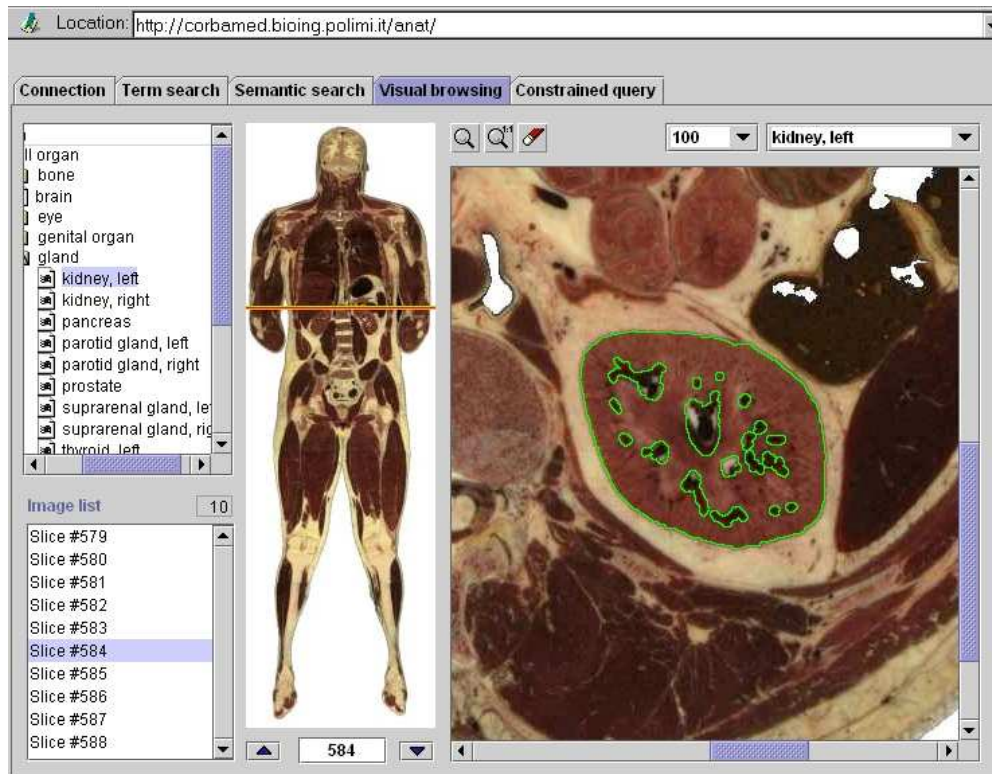


FIGURE 1.21: Le panneau du navigateur visuel se compose de la structure choisie (le rein gauche) dans la taxonomie en haut à gauche, de l'image choisie (le choix se fait en bas à gauche), et de la structure entourée dans l'image choisie.

(<http://www.nlm.nih.gov/research/visible/vhpconf2000/AUTHORS/CERVERI/IMAGINDX.HTM#FIG8>)

Les “Voxels Intelligents” [Pommert94] offrent plus d'informations symboliques (les fonctions des organes) et permettent de visualiser interactivement ces informations. Les données numériques consistent en des images médicales provenant de différentes techniques d'imagerie (IRM, CT-Scan, ...), les voxels de ces images ont des propriétés indiquant à quelle structure anatomique ils appartiennent, ces informations sont stockées dans un ou plusieurs volumes étiquetés (*Label volumes*). Les données symboliques sont des représentations des objets et de leurs relations. Les objets sont représentés par des nœuds et les relations par des arêtes donnant ainsi un réseau sémantique *semantic network*. Les objets ont des attributs comme : un nom, des paramètres d'affichage (couleur, texture, etc.), des pointeurs vers des informations supplémentaires (textes, images, etc.). Les arêtes (relations) ont également des propriétés comme : la symétrie, l'asymétrie et la transitivité. Cela réduit l'ambiguïté et favorise certains inférences [Dameron03].

La relation entre les informations symboliques “génériques” (*semantic network*) et les informations numériques “spécifiques” (*Label volumes*) est créée en liant les objets génériques *classes* aux *instances* de ces objets représentés par les images. Ceci se fait en donnant à chaque instance la référence de la classe correspondante. Grâce à ce lien, les informations symboliques sont accessibles à partir des représentations 3D en cliquant sur l'image, voir Figure 1.22.

Le même groupe de scientifiques [Pommert01] a proposé un atlas informatisé pour les organes internes, voir Section 1.2).

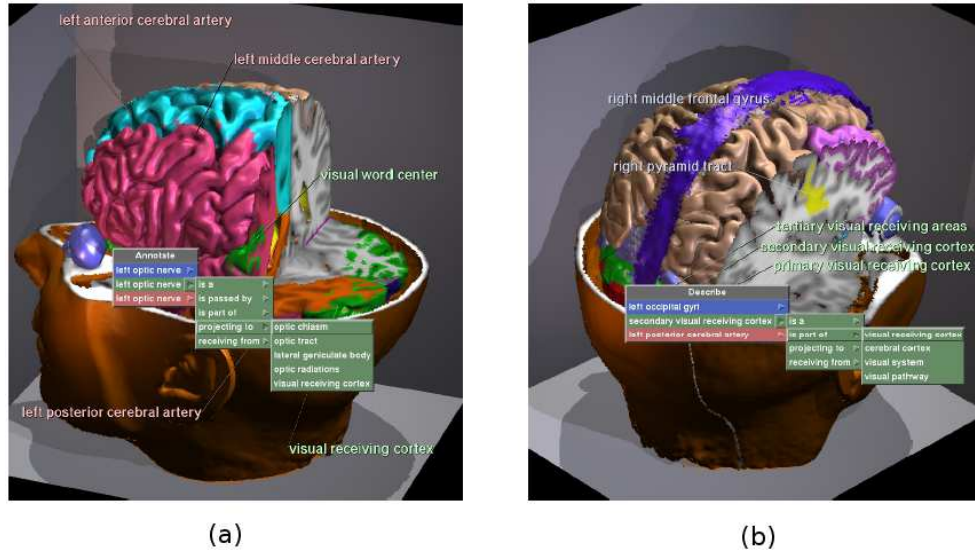


FIGURE 1.22: “Intelligent volumes”. (a) Menu qui apparaît suite à un clic : le nerf optique droit y figure trois fois correspondant aux domaines symboliques dans lesquels il est défini. (b) Autre menu qui apparaît suite à un clic : dans ce cas, trois objets différents appartenant à des domaines différents correspondent au point cliqué. Images issues de [Pommert94].

Une application concernant une structure anatomique précise (le cortex) a été proposée par Gibaud *et al.* [Gibaud03]. Un lien entre des données symboliques et des données numériques a été fait à des fins pédagogiques. Le but de l’application est de fournir un système d’enseignement de la structure du cortex basé web. Les connaissances symboliques sont écrites en XML, un fichier HTML par concept anatomique et un fichier présentant les concepts anatomiques organisés par compositions sont générés. Le fichier propre à chaque concept contient les propriétés de la structure anatomique, ainsi que les relations (taxonomiques, méréologiques et topologiques). Les relations entre les concepts sont réalisées par des liens hypertextes [Dameron03].

Les connaissances numériques comprennent : des coupes IRM, des scènes 3D montrant les sillons corticaux, des scènes 3D pour les gyri, et des vues de dissections. Chacune de ces données possède un fichier XML contenant les métadonnées la décrivant.

Lors du lancement de l’application, l’utilisateur choisit le sujet sur les données duquel il veut travailler, ainsi que les examens souhaités. Plusieurs fenêtres sont ensuite affichées, chacune présentant un type d’information, voir Figure 1.23.

Les fenêtres des coupes IRM et scènes 3D avec les sillons et les gyri sont en correspondance spatiale. Le déplacement dans l’une engendre des déplacements dans les autres. Toutes les fenêtres sont aussi en correspondance sémantique. La sélection d’une structure provoque sa mise en valeur dans toutes les scènes numériques en changeant sa couleur, et vice versa.

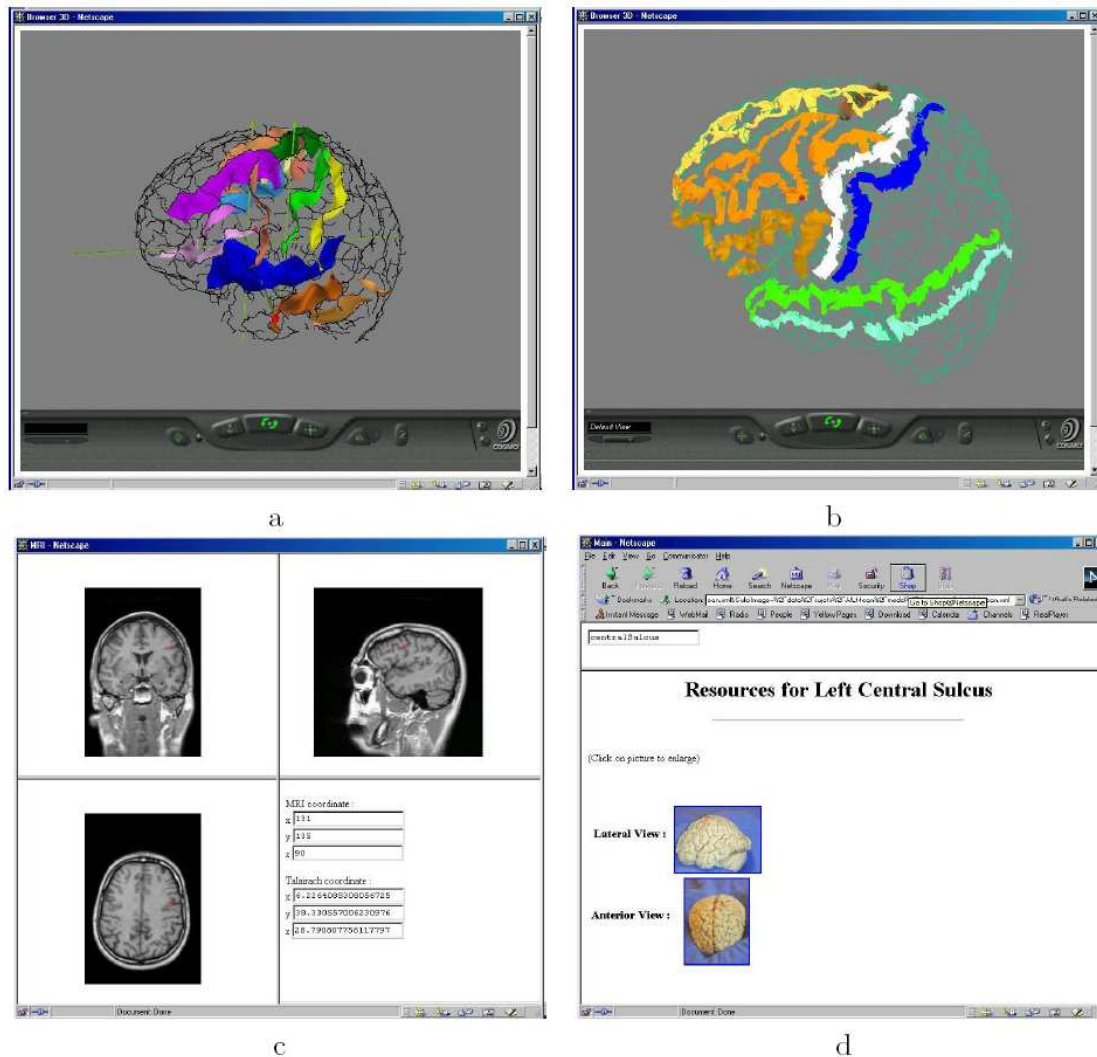


FIGURE 1.23: (a) Fenêtres de la scène 3D avec les sillons corticaux, (b) avec les gyri, (c) les coupes coronales, sagittales, et axiales, (d) la fenêtre principale avec la liste de dissections pour le sillon central. Figure issue de [Dameron03].

Mechouche *et al.* [Mechouche07] s'intéressent aussi au cortex, mais pour faire de l'étiquetage. Pour cela ils ajoutent aux connaissances symboliques et numériques le raisonnement afin d'identifier les structures anatomiques du cortex cérébral humain. La Figure 1.24 montre le schéma de la méthode utilisée. Après la segmentation des images IRM et l'extraction du cerveau puis d'un graphe des sillons corticaux, l'utilisateur choisit une zone d'intérêt. Le sous-graphe correspondant à cette zone est extrait automatiquement. Les relations entre les traces externes et les patches sont décrites dans un fichier OWL DL.

Puis les SPAMs (*Statistical Probability Anatomy Map*) sont utilisées. Une SPAM d'une structure anatomique donnée est une image volumique 3D, l'information en chaque point de ce volume est la probabilité d'appartenance de ce point à la structure en question. Pour chaque trace externe, sa probabilité d'appartenance à chaque SPAM est calculée, et selon certaines heuristiques les positions des traces par rapport aux SPAMs sont déterminées et stockées dans un fichier OWL DL.

L'ontologie écrite en OWL DL est ensuite étendue avec des règles de Horn, afin d'enrichir l'ontologie et propager les relations ce qui permet d'inférer de nouveaux faits. La Figure 1.25 montre l'ontolo-

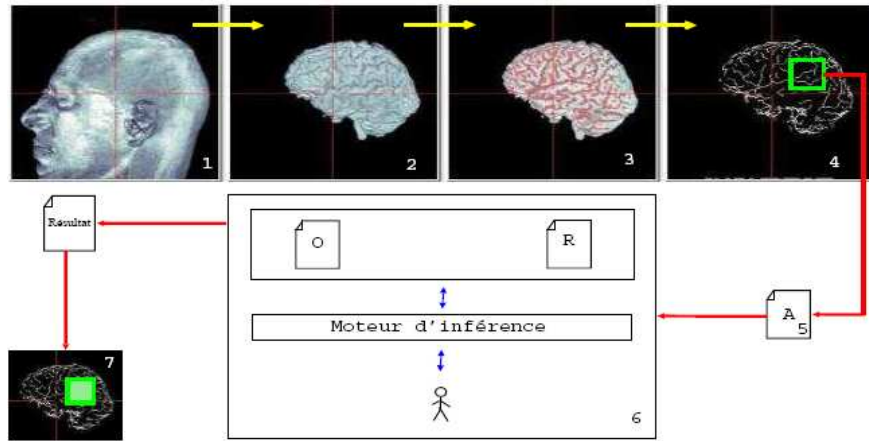


FIGURE 1.24: Le schéma de la méthode d'étiquetage de [Mechouche07].

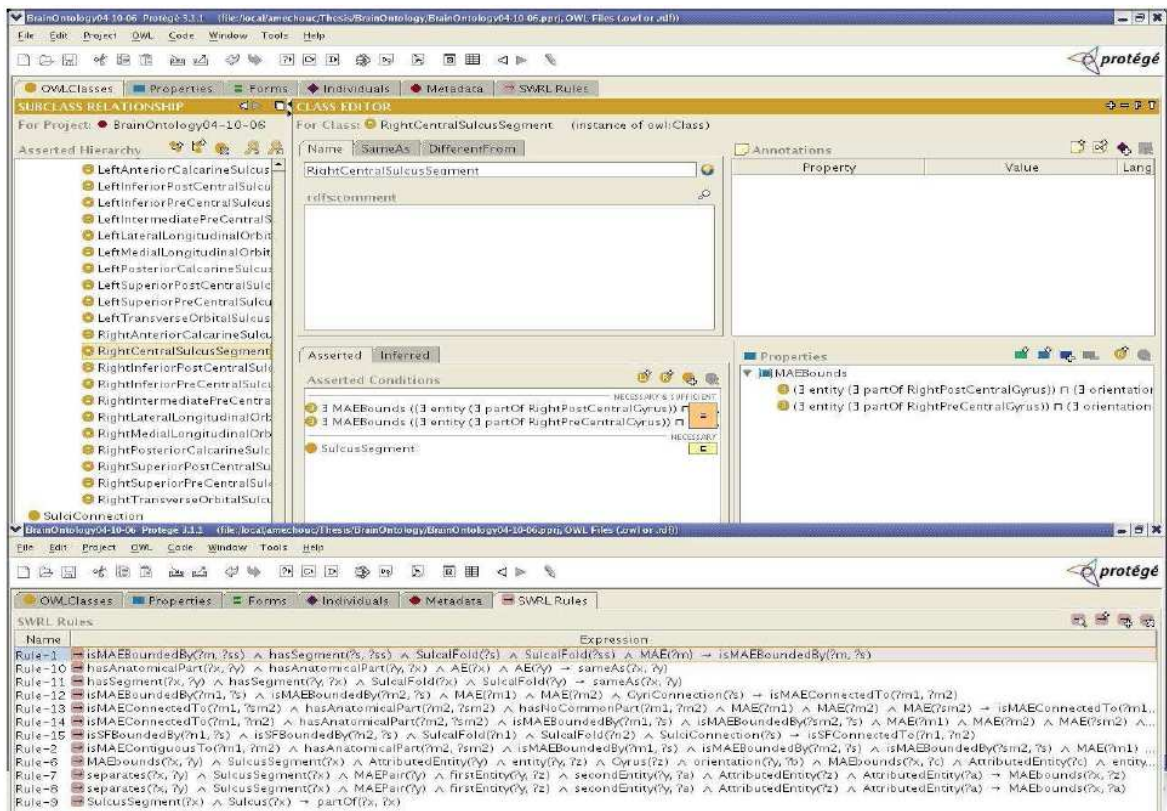


FIGURE 1.25: L'ontologie étendue avec les règles [Mechouche07].

gie étendue. Finalement, l'étiquetage est vérifié en respectant certaines règles de raisonnement et en utilisant un moteur d'inférence (KAON2²¹).

Les applications présentées jusqu'à présent font le lien entre les images et l'ontologie pour enrichir les représentations géométriques par des informations symboliques ou pour identifier des structures anatomiques sur les images. Rubin *et al.* font ce lien pour "raisonner" sur les organes atteints par une balle [Rubin04, Rubin06]. Les connaissances anatomiques sont génériques et proviennent de

21. <http://kaon2.semanticweb.org/>

deux sources : (1) l'ontologie FMA représentant l'anatomie canonique, (2) une ontologie décrivant l'anatomie coronaire et la perfusion régionale. Ce type d'informations n'est pas disponible dans FMA et il est indispensable pour raisonner sur les dommages (*injuries*) secondaires. Pour cela, les auteurs ont créé une ontologie en OWL contenant ces informations.

Les connaissances géométriques sont normalement spécifiques au patient. Des images du *Visible Human* sont prises, manuellement segmentées, et étiquetées par les noms de structures anatomiques. Un modèle géométrique 3D (un maillage) est ensuite créé. Ce maillage est lié aux classes des ontologies (FMA, et coronaires) grâce aux étiquettes.

Finalement, deux outils de raisonnement sont fournis, un permettant de dire quels organes sont atteints, et l'autre pour dire quelles structures vitales sont affectées par conséquence. La Figure 1.26 montre les deux sources de connaissances anatomiques ainsi qu'un modèle géométrique.

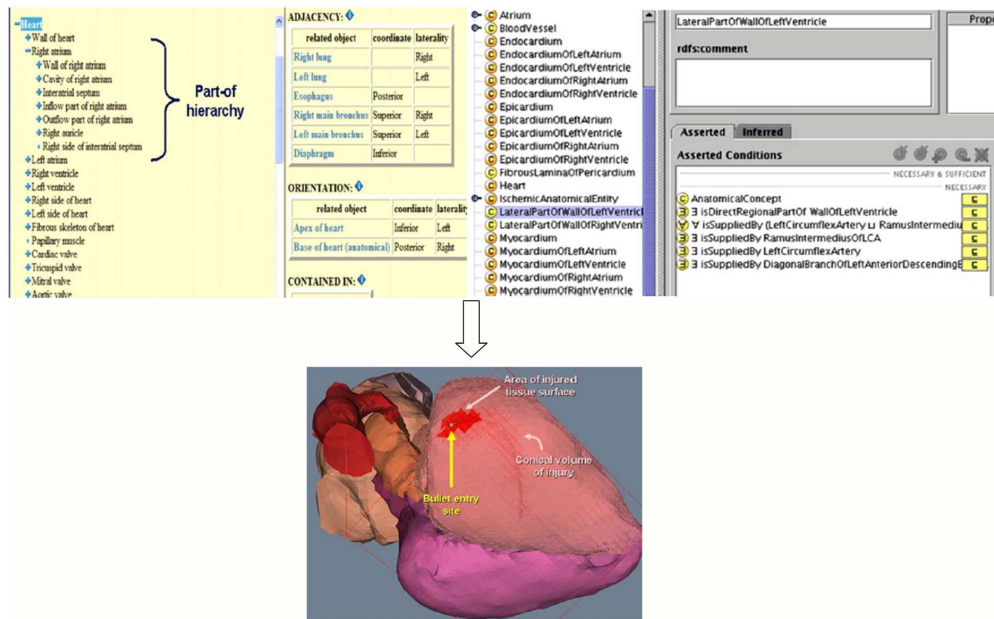


FIGURE 1.26: En haut à gauche : une vue de l'ontologie FMA, en haut à droite : vue de l'ontologie coronaire, et en bas le modèle géométrique 3D [Rubin04, Rubin06].

L'utilisation de l'ontologie et son lien avec la géométrie existent dans d'autres domaines que l'informatique médicale. Garcia-Rojas et al. [Garcia-Rojas05] proposent une ontologie pour représenter les humains virtuels (*Virtual Humans* - VH). Pour construire cette ontologie, ils commencent par définir les questions qui donnent les informations concernant le VH à mettre dans l'ontologie. Ces questions sont classées dans plusieurs catégories : l'histoire du modèle (quels outils ont été utilisés pour la synthèse/modification du VH ? Qui a effectué la tâche *T* sur le modèle *X* ? etc.), la liste de caractéristiques (taille, sexe, etc.), les séquences d'animation (quels sont les modèles utilisés par cette animation ? Est-ce que l'animation est applicable sur le VH *X* ?, etc.), les algorithmes d'animation, et l'interaction avec des objets. Le but de l'ontologie de VH est d'organiser les connaissances de trois domaines de recherche concernant les représentations graphiques des humains : (1) la modélisation et l'analyse du corps humain, (2) l'animation des humains virtuels, et (3) l'interaction entre les humains virtuels et les objets virtuels. Les composants de l'ontologie proposée sont illustrés sur la Figure 1.27. Parmi les applications potentielles de cette ontologie des humains virtuels, on cite : un dépôt pour des données de personnages virtuels, un dépôt de données de modélisation, et la reconnaissance, l'analyse et l'extraction de forme.

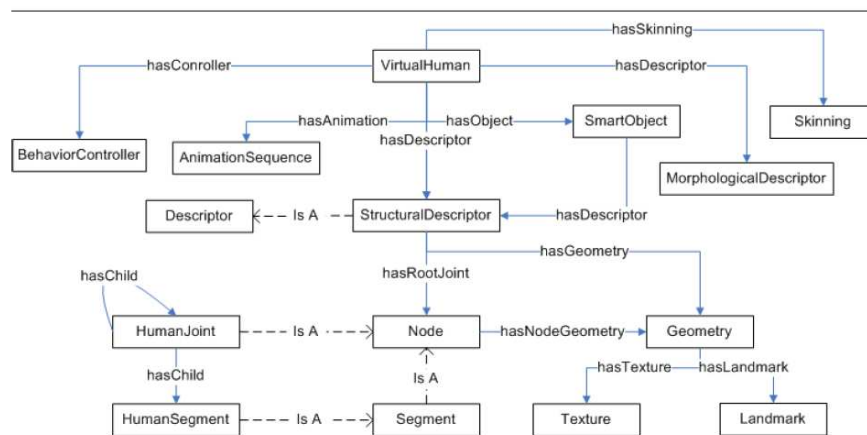


FIGURE 1.27: Les composants de l'ontologie des humains virtuels [Garcia-Rojas05].

Un autre cas non anatomique où l'ontologie est utilisée est proposée dans [Catalano09]. Une ontologie pour la création de produits (*Product Design Ontology - PDO*) est créée. L'ontologie a été modélisée selon le principe de *On-To-Knowledge*, c'est-à-dire en quatre phases. Durant la première phase les concepts clés sont déterminés, dans la deuxième ces concepts sont modélisés en utilisant Protégé et OWL comme langage. La troisième phase consiste à peupler l'ontologie (instancier les

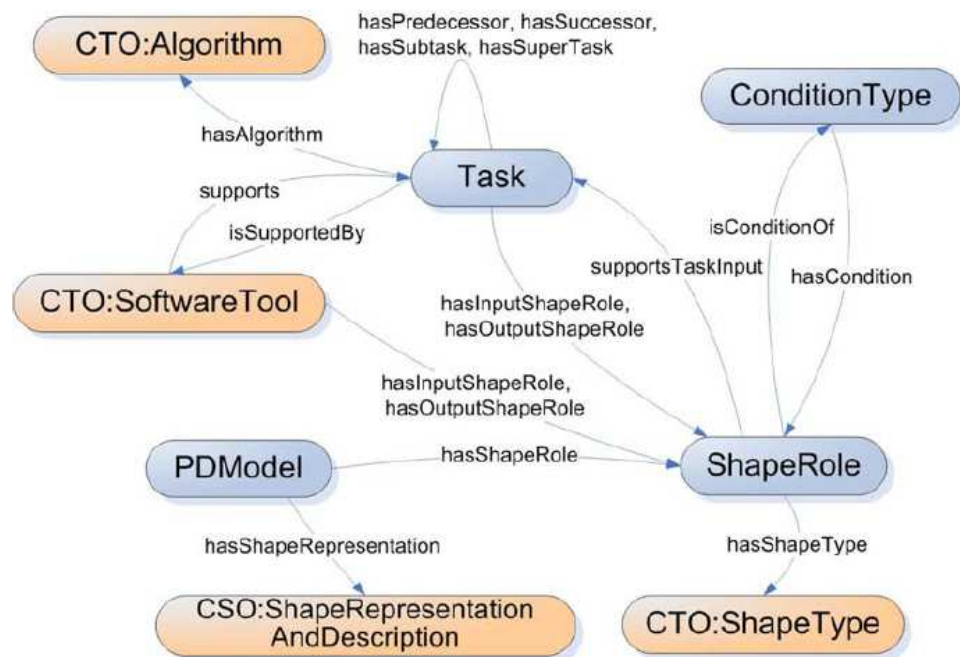


FIGURE 1.28: Les concepts de base de l'ontologie PDO [Catalano09].

classes) et vérifier qu'elle correspond aux buts. La dernière phase est la maintenance et l'évolution de l'ontologie selon les cas d'utilisation. Les composants principaux de l'ontologie sont montrés sur la Figure 1.28.

La partie géométrique est représentée à travers le concept *ShapeRole*. Ce concept montre que la représentation numérique d'une forme est toujours accompagnée par des informations supplémentaires dépendantes du contexte. Cela permet aux utilisateurs non experts de savoir pour chaque tâche quelle

est la forme nécessaire, les conditions à vérifier pour compléter la tâche, et le type de résultat retourné par la tâche. Cette ontologie a été validée comme une partie de *Digital Shape Workbench* du projet AIM@SHAPE²².

Pour conclure cette section, et en restant dans le domaine anatomique qui est le domaine de cette thèse, nous constatons que pour la plupart des applications citées, le lien est fait entre des structures anatomiques “*visibles*” et les connaissances *a priori*. L’ontologie MyCorporisFabrica a comme but d’aider à la reconstruction des structures anatomiques “*non visibles*”. Dans le chapitre 3, nous présentons une utilisation de l’ontologie MyCorporisFabrica pour identifier les parties d’un organe, première étape vers la reconstruction de ces structures non visibles.

4 Conclusion

Au sein de ce chapitre nous avons présenté l’organisation et l’utilisation de connaissances *a priori* dans le domaine de l’informatique médicale.

Nous avons commencé par montrer des cas où ces connaissances sont très simples et ne requièrent pas de représentation spéciale. Ces cas sont principalement ceux où l’on profite d’une information concernant la tâche précise que l’on cherche à faire, comme par exemple profiter de la caractéristique tubulaire des vaisseaux pour les segmenter.

Ensuite nous avons présenté la catégorisation de différentes représentations de connaissances *a priori* anatomiques proposée par Dameron [Dameron03]. En effet, les représentations sont soit basées image, soit basées concepts. Les représentations basées image sont principalement les atlas dans leur diversité (classiques, informatisés, ou probabilistes).

Les représentations basées concepts varient de simples termes de références que l’on trouve dans les livres de terminologie aux ontologies. Nous avons vu que plusieurs définitions du terme ontologie existent. Peu importe la définition adoptée et le domaine pour lequel l’ontologie est construite, elle comprend toujours des entités (concepts, classes, etc.) et des relations qui les lient. Les relations utilisées dans les ontologies sont généralement des relations méréologiques (“fait_partie_de”, “est_distinct_de”, etc.), des relations d’emplacement (“localisé_dans”, “coïncide_avec”, etc.) et des relations topologiques (“connecté_à”, “entouré_par”, etc.). Cependant, d’autres relations peuvent exister suivant le domaine, les connaissances à représenter et l’application envisagée.

Les deux approches (basée images et basée concepts) sont toujours utilisées dans le domaine médical. Évidemment le choix de l’approche et de la représentation dépend largement de l’application envisagée et des connaissances *a priori* dont elle a besoin. Nous nous sommes intéressés aux ontologies car elle permettent de représenter des connaissances *a priori* plus complexes que celles que les images ou de simples termes peuvent représenter.

Ensuite, quelques ontologies du domaine anatomique ont été présentées. Certaines de ces ontologies, comme FMA, sont génériques et ne sont pas destinées à une catégorie spécifique d’utilisateurs. Elle essaient donc de capter les structures anatomiques en général et servent surtout comme base pour faire des ontologies spécifiques à une certaine application. D’autres ontologies se focalisent sur un aspect précis pour répondre aux besoins de l’application envisagée. L’ontologie MyCorporisFabrica,

22. Un projet de réseau européen d’excellence (*European Network of Excellence*). <http://dsw.aimatshape.net/tools/>

par exemple, introduit une base de données avec une partie générique et une partie spécifique au patient. Un lien est fourni entre les deux parties, et des informations géométriques (des acquisitions) sont également stockées dans la base.

Finalement, quelques applications utilisant des ontologies ont été présentées. Nous nous sommes concentrés sur les applications qui font un lien entre les connaissances symboliques (sous forme d'ontologie) et celles numériques (sous forme d'images). Dans la plupart des cas, l'intérêt de ce lien est d'enrichir les images avec les connaissances *a priori*, comme par exemple dans le cas de l'étiquetage des images du cerveau. Dans d'autres cas, un raisonnement est fait afin de retirer des informations qui ne sont pas dans les images, comme dans le cas de la prédiction des organes atteints par la trajectoire d'une balle.

Dans les chapitres suivants, nous présentons deux applications dans lesquelles les connaissances *a priori* sont utilisées avec les connaissances numériques pour accomplir une certaine tâche. Pour la première application, nous utilisons un graphe pour représenter une partie de l'arbre vasculaire humain, voir Chapitre 2-Section 6. Cette représentation est spécifique à l'application qui est la localisation d'anévrismes et peut être considérée basée concepts. Dans le Chapitre 3, l'ontologie MyCorporisFabrica est utilisée pour identifier les parties d'un organe représenté par un maillage.

Application1 : Localisation d'anévrismes

1 Motivation

Un anévrisme est une dilatation dans la paroi d'une artère aboutissant à la formation d'une poche de taille variable¹. La génération d'un anévrisme est souvent liée à plusieurs facteurs [Wier02] :

- une anomalie de la paroi du vaisseau, c'est-à-dire l'affaiblissement de son élasticité interne,
- les effets hémodynamiques des pulsations du liquide,
- l'hypertension artérielle peut, par les à-coups de pression engendrés au niveau de la zone fragilisée, entraîner une augmentation de la taille de l'anévrisme.

Selon l'endroit où se trouve l'anévrisme, on peut distinguer :

- les anévrismes extra-crâniens : du cœur, de l'aorte, etc.
- les anévrismes intra-crâniens : du cerveau. Nous allons nous restreindre à ce type d'anévrismes.

Selon la morphologie de l'anévrisme, on peut distinguer les anévrismes sacciformes, fusiformes ou disséquants, voir Figure 2.1.

La rupture d'anévrisme intra-crânien représente 10% des accidents vasculaires cérébraux (AVC). Lorsqu'elle survient subitement, la mort est souvent inévitable. Des signes précurseurs peuvent cependant attirer l'attention dans les heures et même les jours qui précèdent l'AVC, et un traitement médical est envisageable.

Principalement deux traitements sont possibles :

- Embolisation : c'est un traitement minimalement invasif qui consiste à insérer un cathéter dans l'artère fémorale et à naviguer dans le système vasculaire jusqu'au cerveau et l'anévrisme. Puis, des petits ressorts en platine (*coil*) sont insérés dans l'anévrisme *via* le cathéter. Ces ressorts bloquent la circulation dans l'anévrisme pour éviter sa rupture, voir Figure 2.2-(a).

1. http://fr.wikipedia.org/wiki/An%C3%A9vrisme#An.C3.A9vrismes_intra-cr.C3.A2niens

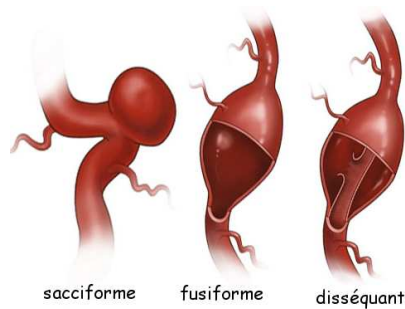


FIGURE 2.1: Types d'anévrismes (<http://nyp.org/health/neuro-cerbaneu.html>).

- Clipping : l'ouverture du crâne est nécessaire pour introduire une pince au niveau du collet de l'anévrisme, voir Figure 2.2-(b). Le collet de l'anévrisme est la zone rétrécie par laquelle il communique avec l'artère porteuse.

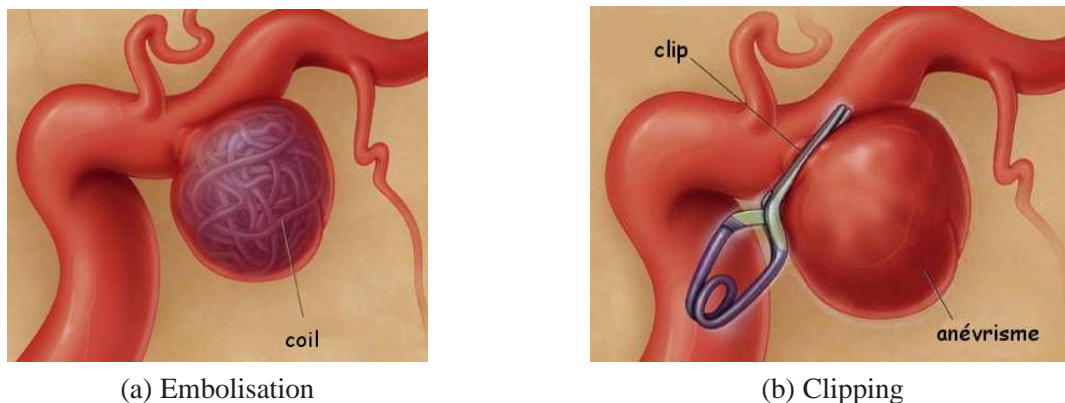


FIGURE 2.2: Traitements possibles (<http://www.brainaneurysm.com/aneurysm-treatment.html>).

Le choix entre ces deux traitements n'est pas toujours évident. Beaucoup d'études ont été faites en essayant de trouver des critères qui aident à décider du meilleur traitement. Par exemple Ecker et Hopkins [Ecker04] essaient de trouver des informations pratiques pour aider les médecins à décider s'il faut traiter l'anévrisme ou s'il suffit de l'observer. Ils profitent surtout d'études statistiques comme *International Study on Unruptured Intracranial Aneurysms (ISUIA)*. Ces études ont montré que le risque de rupture d'un anévrisme de taille inférieure à 10mm est de 0.05% si le patient n'a jamais eu d'hémorragie méningée. De plus les taux de morbidité² et de mortalité³ associés à l'intervention chirurgicale sont de 17.5% et 13.6% pour les premiers 30 jours, et de 15.7% et 13.1% pour la première année après l'intervention. Il faudrait donc être sûr des bénéfices d'une intervention avant de prendre la décision.

Parmi les facteurs qui peuvent être corrélés au risque de rupture, les auteurs ont étudié les suivants :

1. La taille de l'anévrisme : il n'y a pas de taille critique au dessus de laquelle l'anévrisme se rompt. Cependant, le risque de rupture semble être lié d'une façon continue à la taille de l'anévrisme.

2. Le taux de morbidité : le nombre de personnes atteintes par une certaine maladie.

3. Taux de mortalité : le nombre de décès rapportés.

2. Le rapport d'aspect (*aspect ratio*) : ce taux (Figure 2.3) est proposé [Ujiie99] comme indice de rupture. Les statistiques montrent que même si ce taux n'est pas un indice parfait, il est un outil qui aide à trouver les petits anévrismes à grand risque de rupture.

3. L'emplacement de l'anévrisme : l'artère porteuse de l'anévrisme joue un rôle dans le risque de sa rupture. Par exemple, les anévrismes de l'artère communicante postérieure présentent un risque de rupture élevé.

4. Les symptômes autres que la rupture : un anévrisme peut provoquer chez les patients des symptômes comme : manque de vision, ischémie, etc. Dans ce cas, les anévrismes sont traités pour essayer de supprimer ces symptômes. De plus, les anévrismes provoquant de tels symptômes présentent un risque de rupture plus élevé que les anévrismes ne provoquant aucun symptôme.

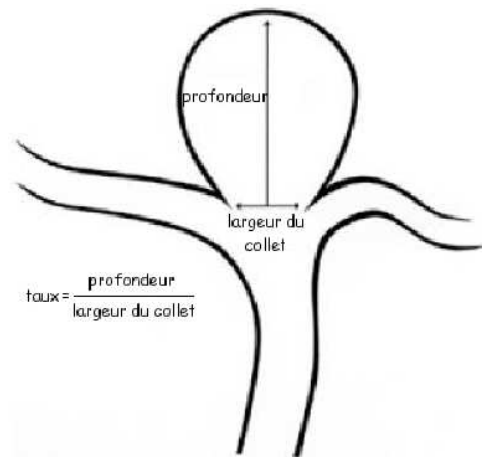


FIGURE 2.3: Rapport d'aspect proposé dans [Ujiie99]

La conclusion que l'on peut retirer est qu'en plus des facteurs spécifiques au patient (âge, symptômes, . . .), la taille, la morphologie et la localisation de l'anévrisme jouent un rôle très important dans le choix du traitement. Cela a motivé beaucoup de travail dans le domaine de la détection et de la quantification d'anévrismes.

2 État de l'art

Le point de départ pour découvrir, surveiller ou observer un anévrisme intra-crânien est l'imagerie médicale. L'angiographie (voir Annexe A, Section 3) est souvent présente dans les modalités d'imagerie utilisées. Elle consiste en l'injection d'un produit de contraste pour rendre les vaisseaux plus visibles dans les images qui sont créées ensuite par des techniques comme la résonance magnétique, ou la tomodensitométrie (voir Annexe A, Section 2 et 1). A partir des images du cerveau acquises par différentes techniques d'imagerie, on essaie d'isoler l'arbre vasculaire des autres tissus du cerveau. Cette opération est appelée la segmentation des images. Les méthodes de segmentation existantes dans la littérature sont très diverses et peuvent être classées selon différents critères : la modalité des images, le type d'organes traités, l'interactivité, etc.

Bien que le problème de segmentation des images ne soit pas l'objet de cette thèse, nous présentons brièvement des méthodes utilisées dans le cas de la segmentation des images cérébrales. Pour un état de l'art plus complet, se référer à [Passat05].

2.1 Segmentation des vaisseaux

Dans le cas de la segmentation des vaisseaux sanguins, certaines méthodes profitent des propriétés offertes par la technique d'imagerie utilisée. Pour la segmentation des vaisseaux sanguins à partir des images prises par la technique d'angiographie par résonance magnétique en temps de vol (TOF

MRA, voir Annexe A, Section 3), les propriétés de circulation du sang sont utiles [Wilson97]. La segmentation est faite en deux étapes : la première consiste à appliquer l'algorithme de maximisation d'espérance pour obtenir deux seuils pour un seuillage d'hystérésis en deuxième étape.

D'autres méthodes utilisent le fait que les vaisseaux sont de forme tubulaire. La mesure de vaisseau (*vesselness measure*) [Frangi98] est un filtre que l'on applique sur l'image médicale pour renforcer les vaisseaux (où les structures tubulaires en général). Pour cela, la Hessienne est calculée et analysée. Les valeurs propres λ_1 , λ_2 , et λ_3 de cette matrice donnent une idée sur les orientations privilégiées de l'objet. Pour une structure tubulaire idéale en 3D nous avons :

$$\begin{aligned} |\lambda_1| &\approx 0 \\ |\lambda_1| &\ll |\lambda_2| \\ \lambda_2 &\approx \lambda_3 \end{aligned}$$

Cela permet de créer un filtre pour renforcer les structures tubulaires. Ce filtrage est appliqué sur des images 2D d'angiographie numérique par soustraction (voir Annexe A, Section 3), et sur des images 3D d'angiographie par résonance magnétique. Dans les deux cas la méthode réussit à rendre les vaisseaux plus visibles.

Des atlas de l'arbre vasculaire sont construits et peuvent être ensuite intégrés pour améliorer la segmentation des images cérébrales [Passat05].

Certains logiciels permettent d'alléger la tâche de segmentation manuelle comme le logiciel Amira⁴, et le logiciel récent *TurtleSeg*⁵. Ce dernier fournit automatiquement une segmentation 3D en se basant sur les contours faits par l'utilisateur dans quelques images 2D.

2.2 Détection et quantification d'anévrismes

Plusieurs méthodes ont été proposées pour la segmentation et la visualisation de l'arbre vasculaire [Wilson97, Tang07]. Cependant, [Millan07] propose une méthode pour donner une caractérisation morphologique d'anévrismes afin de prédire leur risque de rupture. La Figure 2.4 montre les différentes étapes proposées.

La segmentation : Les auteurs utilisent deux méthodes de segmentation selon la modalité d'imagerie. Si la modalité utilisée est l'angiographie rotative 3D alors la méthode de segmentation utilisée est celle des modèles déformables. Alors que si la modalité est l'angiographie par tomodensitométrie, la méthode utilisée est celle des surfaces implicites déformables basées région. L'étape de segmentation est suivie par un *marching cubes* [Lorensen87]. Le maillage reconstruit est ensuite lissé et optimisé afin d'obtenir un maillage triangulaire qui représente le modèle des vaisseaux.

La squelettisation : Pour cela les auteurs appliquent la méthode proposée par Bouix et al. [Bouix05]. Le squelette est nécessaire pour l'étape de normalisation. Nous donnons plus de détails sur les squelettes et les algorithmes de squelettisation dans la section 3.

L'isolement de l'anévrisme : Les auteurs proposent des plans de coupe pour isoler l'anévrisme. Ces plans de coupe sont positionnés à une distance équivalente au diamètre du vaisseau porteur

4. <http://www.amira.com/>

5. Un outil de segmentation interactif développé dans le laboratoire *Medical Image Analysis Lab* de l'université Simon Fraser et le laboratoire *Biomedical Signal and Image Computing Laboratory* de l'université de British Columbia (<http://www.turtleseg.org/>)

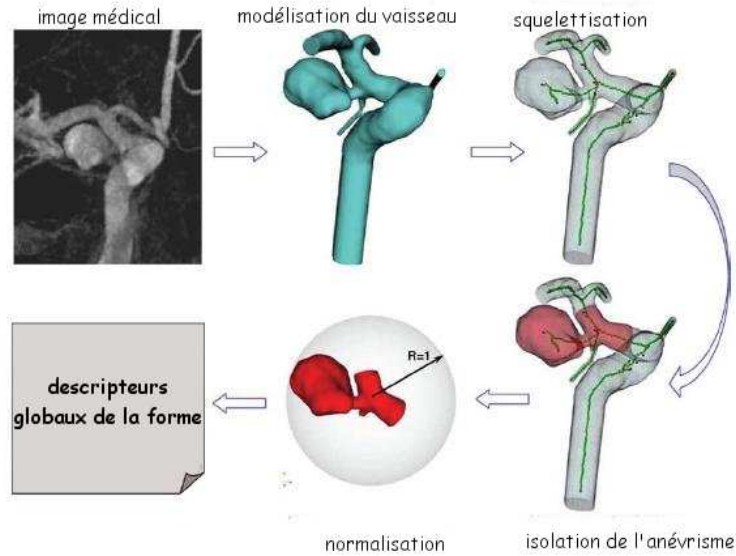


FIGURE 2.4: Le pipeline de la méthode de [Millan07]

de l'anévrisme. Ainsi les anévrismes qui ont le même dôme (sac anévrisimal) et qui sont liés différemment aux vaisseaux auront des caractéristiques morphologiques différentes, ce qui est exigé car ils ont des risques de rupture différents.

La normalisation : Pour obtenir des descripteurs invariants en translation, les auteurs les normalisent dans une sphère unité dont le centre est le point de bifurcation du squelette.

Le calcul des descripteurs : Les moments sont définis généralement comme les projections d'une fonction représentant l'objet sur un ensemble de fonctions caractérisant le type de moments utilisé [Novotni04] :

$$m_i = \langle f, \psi_i \rangle = \int_{\Omega} f(x) \cdot \overline{\psi_i(x)} d(x)$$

où f est la fonction de l'objet, $\psi = \psi_i$ ($i \in N$) est la base des fonctions.

Une propriété des moments la plus recherchée est l'invariance aux transformations (principalement : translation, rotation, réflexion et dilatation uniforme). Les auteurs étudient deux types de moments invariants 3D : géométriques et de Zernike. Les résultats montrent que les moments invariants de Zernike sont meilleurs.

Les résultats trouvés montrent que l'utilisation des moments invariants comme indices de forme de l'anévrisme pour prédire son risque de rupture est très prometteuse.

Firouzian *et al.* [Firouzian10] ont proposé récemment une méthode pour segmenter les anévrismes intra-crâniens à partir d'images d'angiographie par tomodensitométrie. L'utilisateur choisit un point germe à l'intérieur de l'anévrisme puis l'anévrisme est détecté en minimisant une fonction d'énergie. La fonction dépend de trois facteurs : l'intensité, la variance de l'intensité et la magnitude du gradient. Le volume de l'anévrisme est ensuite calculé. La méthode est validée sur un groupe de 11 patients avec 15 anévrismes, et en utilisant trois mesures :

- L'indice de ressemblance (*Similarity Index- SI*) : c'est le montant de chevauchement (*overlap*) entre la segmentation manuelle S_{man} faite par un radiologue et la segmentation automatique S_{auto} de la méthode.

$$SI = 2 \times \frac{\|S_{man} \cap S_{auto}\|}{\|S_{man} + S_{auto}\|}$$

- La moyenne de la distance de surface (*Average Surface Distance- ASD*) est la moyenne de distance entre deux surfaces de segmentation :

$$ASD = \frac{2h(V_{MS} - V_{AS})}{(D_{MS} - V_{MS}) + (V_{MS} - E_{MS})},$$

où : V_{MS} , V_{AS} sont les volumes de la segmentation manuelle et automatique respectivement,
 D_{MS} est le volume de la segmentation manuelle dilatée par un voxel,
 E_{MS} est le volume de la segmentation manuelle érodée par un voxel,
 h est la taille d'un voxel.

- La différence de volume (*Volume Difference- VD*) : c'est la différence entre le volume de la segmentation manuelle et automatique.

Ces mesures de validation sont généralisables sur n'importe quelle méthode de quantification d'anévrismes.

Le but du travail que l'on propose est de détecter, quantifier et localiser les anévrismes sacciformes intra-crâniens automatiquement afin d'aider les médecins à prendre une décision sur le traitement. Nous supposons que la segmentation de l'arbre vasculaire est déjà faite, et sur cet arbre vasculaire nous effectuons les étapes suivantes :

1. **Extraction des lignes de centre des vaisseaux** : Nous proposons un nouvel algorithme pour extraire les lignes de centre des vaisseaux. Cet algorithme est une extension de l'algorithme de Dijkstra. Cette étape est détaillée dans la section 3.
2. **Détection automatique d'anévrismes** : Nous proposons un critère pour trouver les anévrismes sacciformes automatiquement sans aucune intervention de l'utilisateur. Ce critère est basé sur l'évolution du diamètre des vaisseaux. Cette étape est détaillée dans la section 4.
3. **Quantification d'anévrismes** : Une fois l'anévrisme détecté, différentes mesures sont données. Ces mesures ont été choisies pour aider les médecins à caractériser l'anévrisme, puis à faire le choix sur le meilleur traitement. Cette étape est détaillée dans la section 5.
4. **Localisation d'anévrismes** : puisque le site de l'anévrisme est l'un des facteurs qui affectent son risque de rupture, le nom de l'artère porteuse de l'anévrisme est donné automatiquement. Pour cela nous utilisons une technique de mise en correspondance entre graphes. Cette étape est détaillée dans la section 6.

3 Extraction des lignes de centre des vaisseaux

3.1 Définitions

Le voxel est un pixel en 3D. Les voisinages entre les voxels peuvent être définis selon l'incidence entre les voxels [Boltcheva07] :

- Si deux voxels sont incidents à une même face, alors ils sont 6-voisins,
- Si deux voxels sont incidents à une même arête mais pas à une même face, alors ils sont 18-voisins,
- Si deux voxels sont incidents à un même sommet mais pas à une même arête, alors ils sont 26-voisins

Selon le voisinage choisi on peut définir un objet 6-, 18- ou 26-connexe. Un objet est i -connexe, si toute paire de voxels de cet objet est i -connexe, c'est-à-dire : entre ces voxels il existe une suite de voxels $(V_k, 1 \leq k \leq n)$, telle que V_k et V_{k+1} sont i -voisins.

Le squelette est une représentation compacte de l'objet. En dimension 2, le squelette est l'ensemble des lignes inter connectées au centre de l'objet [Fouard04]. En dimension 3, ce squelette peut être également un ensemble de lignes (comme en dimension 2), ou il peut contenir des surfaces centrales. Les lignes qui composent le squelette sont appelées les lignes de centre par la suite. La plus longue ligne est appelée la ligne principale, alors que les autres lignes sont appelées les branches.

Nous sommes intéressés par le premier type de squelettes, ceux qui sont composés de lignes centrales. Nous cherchons un squelette 26-connexe et mince, où tout voxel du squelette a exactement deux de ses 26-voisins dans le squelette, sauf les extrémités (ils n'en ont qu'un voisin) et les jonctions qui en ont trois.






illustration	nom	caractérisation
	4-voisinage	$d = 1$
	8-voisinage	$d \leq 2$
	6-voisinage	$d = 1$
	18-voisinage	$d \leq 2$
	26-voisinage	$d \leq 3$

FIGURE 2.5: Types de voisinages et dimensions, figure prise de [Boltcheva07].

3.2 État de l'art

Dans le domaine de l'imagerie médicale, cette extraction du squelette peut être faite :

Pendant la segmentation Certaines méthodes sont basées sur l'intensité [Aylward96]. Dans ces méthodes, une image en N dimensions est considérée comme une surface en $N+1$ dimensions, où la hauteur correspond à l'intensité. Pour les objets qui ont une coupe circulaire, l'intensité est maximale au centre de la coupe. Les lignes de centre sont donc des crêtes sur la surface d'intensité. Le rayon est estimé avec l'aide de l'utilisateur qui donne une estimation initiale. La méthode est validée sur deux techniques différentes d'imagerie : angiographie par résonance magnétique (pour extraire les vaisseaux), et la tomodensitométrie (pour extraire les os). Les effets de l'initialisation, du bruit, et des singularités sur cette méthode sont étudiés plus tard, et plusieurs heuristiques pour minimiser ces effets sont proposées [Aylward02].

Le caractère tubulaire est également utilisé dans [Wink04]. La mesure de vaisseaux [Frangi98] est calculée sur plusieurs échelles pour segmenter les vaisseaux. Un algorithme de propagation est ensuite utilisé pour trouver les lignes de centre des vaisseaux en 2D et 3D.

Deschamps et *al.* proposent une méthode pour extraire une ligne centrée qui va d'une extrémité à l'autre [Deschamps01]. Cette méthode qui requiert un minimum d'interaction par l'utilisateur (pour déterminer les deux extrémités), est basée sur la technique de propagation des ondes. Cette technique est utilisée deux fois : la première est une propagation vers l'extérieur (*outward*) pour trouver le chemin le moins coûteux entre les deux points, l'autre est une propagation vers l'intérieur (*inward*) pour centrer ce chemin. L'étape de centrage est faite après la segmentation des structures tubulaires. La méthode est appliquée sur la colonoscopie virtuelle.

La mesure de vaisseaux [Frangi98] est utilisée d'abord pour trouver les lignes de centre des

objets tubulaires et estimer leur rayon. Cette mesure est ensuite distribuée pour créer un champ de vecteurs [Descoteaux08]. Cela permet l'application d'un algorithme de maximisation de circulation du flux (*flux maximization flow algorithm*) pour récupérer les bords des vaisseaux.

Après la segmentation. On peut distinguer :

1. Les approches basées distance : ces approches construisent une carte de distance qui sera utilisée pour trouver la/les lignes de centre. La plupart de ces approches utilisent à un moment donné l'algorithme de Dijkstra [Dijkstra59] pour construire un arbre des plus courts chemins à un voxel/point source S . L'utilisation de l'algorithme de Dijkstra, avec la distance euclidienne et en prenant une extrémité de l'objet comme source, donne une ligne de centre qui coupe les virages, voir Figure 2.6. Cela est attendu, puisque la ligne qui coupe les virages est plus courte qu'une ligne qui passe au centre.

CEASAR [Bitter00] est une méthode pour extraire une ligne de centre (la ligne principale). Cette méthode utilise une distance pénalisée par la distance au bord de l'objet DFB (*Distance From Boundary*) pour éloigner cette ligne de la surface de l'objet. Cela répond à l'inconvénient de l'algorithme de Dijkstra où la ligne coupe les virages.

TEASAR [Sato00] est une extension de la méthode *CEASAR* où tout le squelette est extrait. D'abord, *CEASAR* est appliquée pour extraire la ligne de centre principale, puis les branches sont extraites pour avoir un squelette complet. Les auteurs de ces deux méthodes essaient plus tard ([Bitter01]) de résoudre les deux inconvénients de leurs premières méthodes. La Figure 2.7 montre ces deux inconvénients.

Le premier inconvénient est le phénomène de pointe (*spike*). On peut constater sur la Figure 2.7-(a) que si l'objet a des pointes (*spikes*), la ligne de centre va passer dans ces pointes au lieu de parcourir l'objet. En effet, la méthode *CEASAR* trouve la ligne de centre comme le chemin qui a le coût le plus élevé parmi tous les chemins les moins coûteux. Puisque ce coût augmente en s'approchant du bord de l'objet, la ligne qui passe dans les pointes est la plus coûteuse.

Le deuxième inconvénient est la jonction entre les branches (ou une branche et la ligne principale). On peut constater sur la Figure 2.7-(b) que la branche rejoint la ligne principale dans un endroit très éloigné (presque à la source) de l'endroit souhaité Figure 2.7-(c).

Pour résoudre le problème de pointe, au lieu d'utiliser la distance pénalisée pour trouver la deuxième extrémité de la ligne de centre, les auteurs calculent la distance euclidienne à la source et l'utilisent pour trouver les extrémités de la ligne de centre. Le problème de jonction est résolu en re-calculant la distance pénalisée mais cette fois à tous les voxels de la ligne de centre (au lieu de la source S). La branche est extraite ensuite comme le chemin entre le voxel le plus loin (en utilisant la distance euclidienne) et le premier voxel de la ligne de centre.

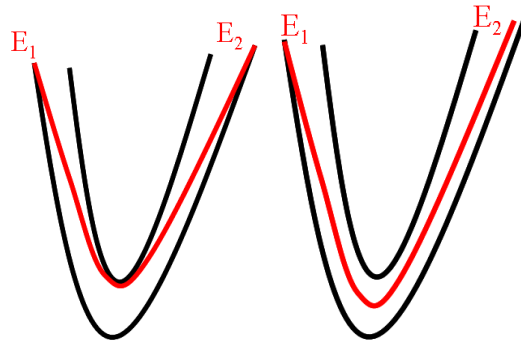


FIGURE 2.6: La ligne de centre que l'on obtient en utilisant l'algorithme de Dijkstra avec la distance euclidienne (à gauche), et la ligne centrée que l'on souhaite obtenir (à droite)

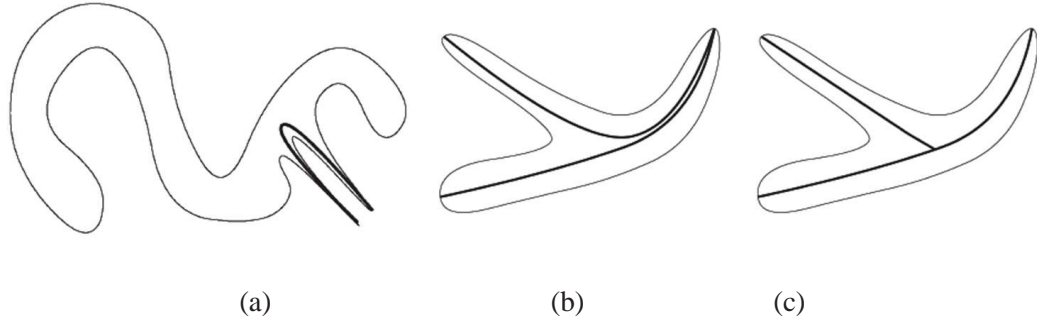


FIGURE 2.7: Les désavantages de CEASAR [Bitter00] (a), et de TEASAR [Sato00] (b) selon [Bitter01].

L'algorithme de Dijkstra est utilisé par Wan et *al.* pour construire un arbre des chemins les plus courts mais selon la distance des voxels au bord de l'objet DFB [Wan02]. Un voxel source S est choisi par l'utilisateur pour commencer la construction de l'arbre. Pendant la construction de l'arbre, les voxels sont stockés dans un tas ordonné selon $\frac{1}{DFB}$, et la distance euclidienne (DFS) de chaque voxel à la source S est parallèlement calculée et stockée. La ligne de centre est ensuite le chemin dans l'arbre entre S et l'autre extrémité E (soit indiquée par l'utilisateur, soit prise comme le voxel avec la distance DFS maximale). Le fait de traiter les voxels selon l'inverse de leur distance au bord privilégie les voxels les plus loin du bord, et donc donne une ligne centrée.

Ensuite, pour extraire des branches directement connectées à cette ligne, on parcourt les voxels de la ligne principale de E vers S . Pour chaque voxel C de la ligne de centre, et pour chacun de ces voisins B (sans compter ses voisins de la ligne de centre), on trouve les sous-arbres liés à B . Puis on identifie dans chacun de ces sous-arbres le voxel T avec la distance DFS maximale. Si la longueur du chemin entre B et T (longueur = $DFS[T] - DFS[B]$) est au-dessus d'un certain seuil, T est considéré comme une extrémité d'une branche. La branche est extraite en remontant l'arbre : de l'extrémité T et jusqu'à B . Cet algorithme est utilisé pour extraire la ligne de centre du côlon afin de faire de la navigation virtuelle. L'algorithme est efficace au niveau de temps de calcul, et la ligne de centre obtenue est centrée, connexe et mince. Mais la méthode pour extraire les branches devient de plus en plus coûteuse si on veut trouver les branches qui ne sont pas connectées directement à la ligne principale mais aux autres branches déjà extraites.

Jiang et *al.* [Jiang05] utilisent une méthode très proche de la précédente, avec cependant quelques différences principales. D'abord, et pour garantir la centralité, au lieu de trier les voxels selon $\frac{1}{DFB}$ ils utilisent $MAX - DFB$ où MAX est une constante suffisamment grande pour que $MAX - DFB$ soit toujours positif. Puis, au lieu de travailler sur la totalité des voxels de l'objet, ils enlèvent les voxels du bord (*Boundary Voxels Cutting*), ce qui réduit le temps de calcul. Finalement, ils proposent de trouver la source S automatiquement en exécutant l'algorithme de Dijkstra avec un voxel arbitraire comme source, puis ils prennent le voxel le plus éloigné de ce voxel arbitraire comme la source S pour la suite. Leur méthode est également validée sur la navigation virtuelle dans le côlon, ce qui pourrait expliquer pourquoi ils ne traitent pas l'extraction des branches.

Le point fort de ces approches est leur efficacité au niveau du temps de calcul (leur complexité est de $O(N \log N)$, où N est le nombre de voxels) et la centralité des lignes de centre obtenues. Mais le squelette obtenu avec ces méthodes a toujours la structure d'un arbre, et n'est donc pas toujours homotope à l'objet.

2. Les approches d'amincissement topologique : L'idée principale de ces approches est de peler

itérativement l'objet couche par couche jusqu'à arriver au squelette.

Les points de l'objet sont testés pour suppression à chaque itération, seulement les points dits "simples" sont supprimés [Palagyi01]. Un point est simple si sa suppression ne modifie pas la topologie de l'objet. La boucle s'arrête quand aucune suppression n'est plus possible. Cette méthode d'amincissement est séquentielle. Elle est appliquée pour extraire le squelette du côlon et aussi celui des vaisseaux pour détecter la présence d'anévrismes infraréniaux, ou de sténoses.

Puisque ces méthodes doivent vérifier à chaque suppression la conservation de la topologie, elles sont très coûteuses. Pour accélérer ces méthodes, la suppression des points peut être faite d'une façon parallèle [Lohou02]. Pour supprimer les points parallèlement il faut faire attention à ne pas changer la topologie de l'objet. Pour cela, une itération d'un algorithme d'amincissement est divisée en 6 sous-itérations. Les sous-itérations peuvent correspondre aux directions ou aux sous-grilles. En plus de la parallélisation Lohou et Bertrand utilisent la notion de points P-Simple (*P-Simple points*), qui sont des groupes de points dont la suppression ne change pas la topologie de l'objet. Les mêmes auteurs proposent plus tard un algorithme avec 12 sous-itérations [Lohou04].

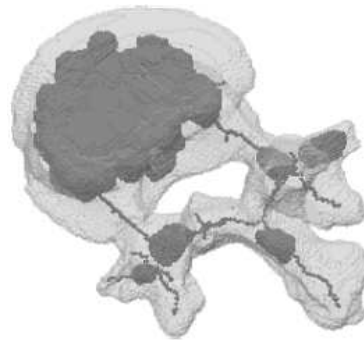


FIGURE 2.8: Le résultat obtenu par la méthode proposée dans [Lohou02].

La notion de noyau critique a été présentée par Bertrand [Bertrand07], et elle a été utilisée pour faire l'amincissement topologique en 3D [Bertrand06], et en 2D [Bertrand08]. Il est montré après (jusqu'à la dimension 4) l'équivalence entre les points P-Simple et la notion de noyau critique [Bertrand09].

Le point fort des approches d'amincissement topologique est d'obtenir directement un squelette (la ligne principale + les branches) qui est homotope à l'objet. Cependant, le résultat peut comprendre des surfaces médianes, voir Figure 2.8. Cela demande plus de traitement pour avoir des lignes et non des surfaces, ce qui augmente le temps de calcul.

3. Les approches hybrides : Ces approches essaient de garder les avantages des deux approches précédentes. Dans [Pudney98], par exemple, l'amincissement topologique (suppression séquentielle des points dont la suppression ne change pas la topologie) est fait dans un ordre croissant de la distance de ces points au bord de l'objet. Le résultat est un squelette centré, homotope à l'objet et mince.

Fouard et *al.* adaptent la méthode hybride ci-dessus pour extraire le squelette dans le cas de données très grandes, comme pour la microcirculation du cerveau [Fouard04]. Le squelette est extrait par blocs, tout en conservant les propriétés nécessaires du squelette : l'homotopie, la centricité, et le caractère mince.

L'axe médian est d'abord extrait en se basant sur le gradient d'une fonction de distance euclidienne au bord [Bouix05], puis cet axe est raffiné (par amincissement) pour avoir des courbes 3D. La combinaison entre l'utilisation d'une fonction de distance et l'amincissement donne un squelette centré et homotopique à l'objet.

Finalement, [Cornea07] donne une étude très détaillée sur les squelettes, leur propriétés, les différentes classes de méthodes de squelettisation, et une comparaison entre ces méthodes.

3.3 Notre méthode

Nous supposons que la segmentation de l'arbre vasculaire est déjà faite et que nous avons une représentation 3D de cet arbre (sous forme de voxels). Nous cherchons un squelette qui soit un ensemble de lignes centrées, minces, et 26-connexes. La méthode que nous proposons pour l'extraction des lignes de centre est une approche basée distance. Nous avons choisi cette classe de méthodes car elles donnent directement des lignes de centre (pas de surfaces), et pour leur efficacité en terme de temps de calcul. Nous veillons à ce que les lignes de centre soient extraites rapidement, car leur extraction est une étape parmi plusieurs pour détecter et localiser les anévrismes. Notre ensemble de lignes de centre ne sera pas forcément homotope à l'arbre vasculaire, mais ceci n'arrive que rarement, et n'est pas une problème critique pour la détection et la quantification de l'anévrisme. Cette méthode est en fait une extension de la méthode proposée dans [Bitter01]. Nous détaillons dans la suite les deux étapes principales de notre méthode pour trouver les lignes de centre.

3.3.1 Extraction d'une ligne de centre principale

La plupart des algorithmes basés distance sont des variations de l'algorithme de Dijkstra. L'algorithme de Dijkstra a été proposé pour les graphes. Dans notre cas, les voxels de l'objet seront les nœuds du graphe et ils seront liés entre eux de façon à minimiser leur distance (DFS) à un voxel source S , la racine du graphe. Le résultat de l'algorithme de Dijkstra est un arbre des plus courts chemins de S à tous les autres voxels. La Figure 2.9 rappelle le principe de l'algorithme de Dijkstra.

Comme dans les autres algorithmes de la littérature, la ligne de centre "principale" est le chemin le plus long dans l'arbre, c'est à dire le chemin que l'on obtient en remontant l'arbre du voxel le plus loin de la source, jusqu'à la source S . Si la distance ($Dist$ dans l'algorithme de la Figure 2.9) utilisée pendant la construction de l'arbre est la distance euclidienne, la ligne de centre obtenue coupe les virages, voir Figure 2.6. Ce phénomène est attendu, puisque le chemin qui coupe le virage est plus court qu'un chemin qui reste centré dans l'objet.

Nous avons vu dans la section 3.2 différentes solutions pour résoudre ce problème et forcer l'algorithme à privilégier les voxels loin du bord de l'objet. Ce que nous proposons pour résoudre le problème est d'utiliser la fonction de distance suivante :

$$Dist(V, N) = \frac{Euc(V, N)}{1 + DFB[V] + DFB[N]} \quad (2.1)$$

où :

- $Euc(V, N)$ est la distance euclidienne entre les deux voxels V et N ,
- $DFB[V]$ est la distance du voxel V au bord de l'objet,
- $DFB[N]$ est la distance du voxel N au bord de l'objet,
- le bord de l'objet est défini comme l'ensemble des voxels de l'objet qui ont moins de 26 voisins dans l'objet.

Le fait de diviser la distance euclidienne par les distances au bord privilégie les voxels qui ont une distance au bord élevée, et donc les voxels qui, pour une distance euclidienne constante, sont centrés dans l'objet.

Pour que l'extraction de la ligne de centre soit automatique, le choix des deux extrémités doit se faire sans l'intervention de l'utilisateur. Ces extrémités doivent correspondre aux deux extrémités de l'objet. Pour les trouver d'une façon totalement automatique, nous procédons ainsi :

- La première extrémité (qui est également la source S) [Lazarus99] : pour trouver la source, nous exécutons l'algorithme de Dijkstra en utilisant la distance euclidienne, et en prenant n'importe quel voxel comme une source. Le voxel le plus loin de cette source arbitraire sera une extrémité de l'objet. Il sera donc la première extrémité de la ligne de centre que l'on cherche, ou notre voxel source S .

- La deuxième extrémité : une fois la source trouvée, nous pouvons appliquer l'algorithme de Dijkstra pour construire l'arbre des plus courts chemins à cette source. Pendant la construction de l'arbre, nous calculons à la fois la distance à la source DFS selon la fonction $Dist$ (cf. Eq. 2.1) et la distance euclidienne à la source ($EucDFS(V,N) = Euc(V,N) + EucDFS[V]$). La deuxième extrémité de la ligne de centre est choisie comme le voxel (E) le plus éloigné (selon la distance euclidienne) du voxel source ; c'est-à-dire qui a la valeur maximale pour $EucDFS$.

La ligne de centre est l'ensemble des voxels obtenus en remontant l'arbre du voxel (E) jusqu'à la source (S). Nous appelons cet ensemble de voxels la ligne de centre principale (B_0).

La Figure 2.10 montre le résultat de notre algorithme sur une représentation volumique d'une vache. Nous pouvons constater que la ligne de centre obtenue est bien située entre deux extrémités de l'objet, est connexe, mince et bien centrée même dans les endroits étroits comme la patte de la vache.

La Figure 2.11 montre la ligne de centre obtenue dans le cas d'un arbre vasculaire. Les deux zooms montrent que la ligne de centre ne coupe pas les virages (b) et qu'elle reste valide dans les endroits les plus étroits (c).

Nous procédons maintenant à l'extraction des branches pour obtenir le squelette de l'arbre vasculaire.

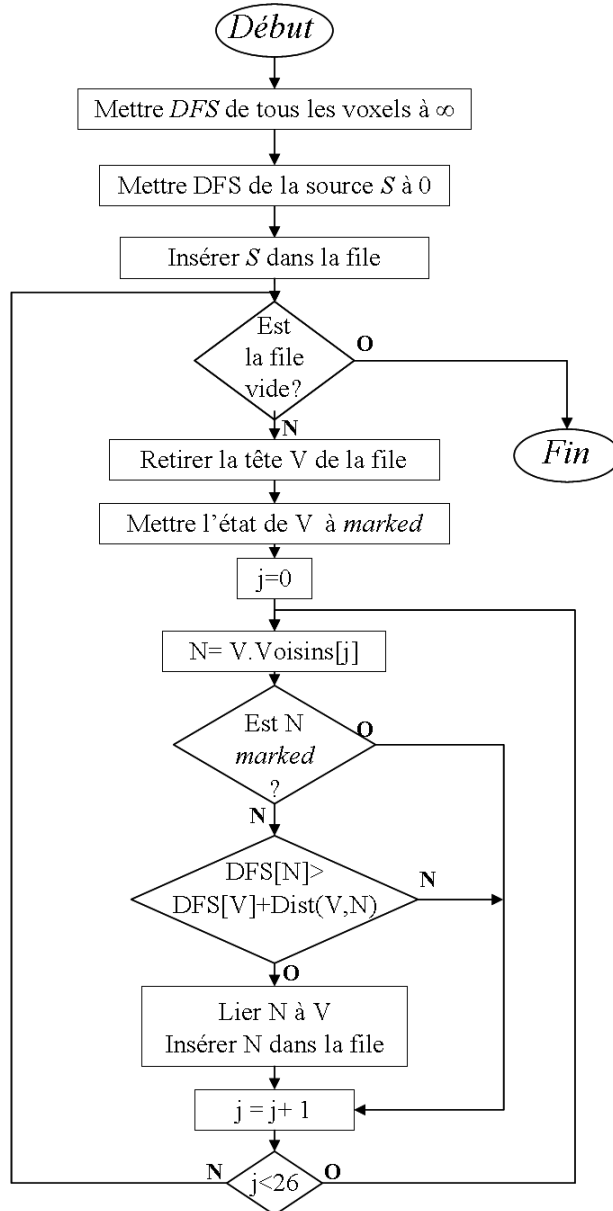


FIGURE 2.9: Algorithme pour construire l'arbre des plus courts chemins (*Minimum Spanning Tree*).

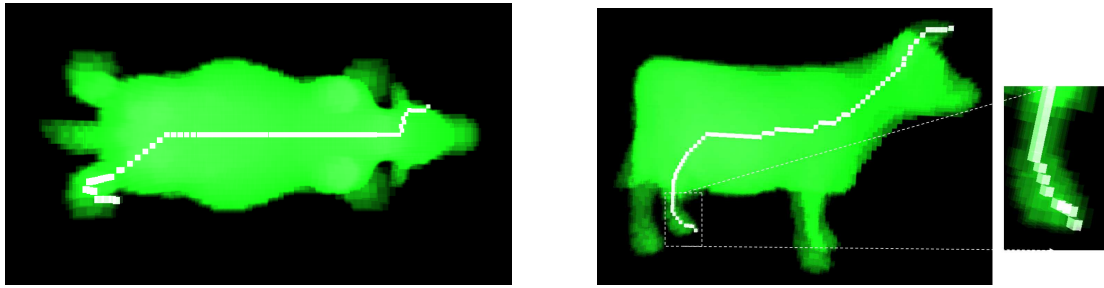


FIGURE 2.10: Trois vues de la ligne de centre principale extraite dans le cas d'une vache.

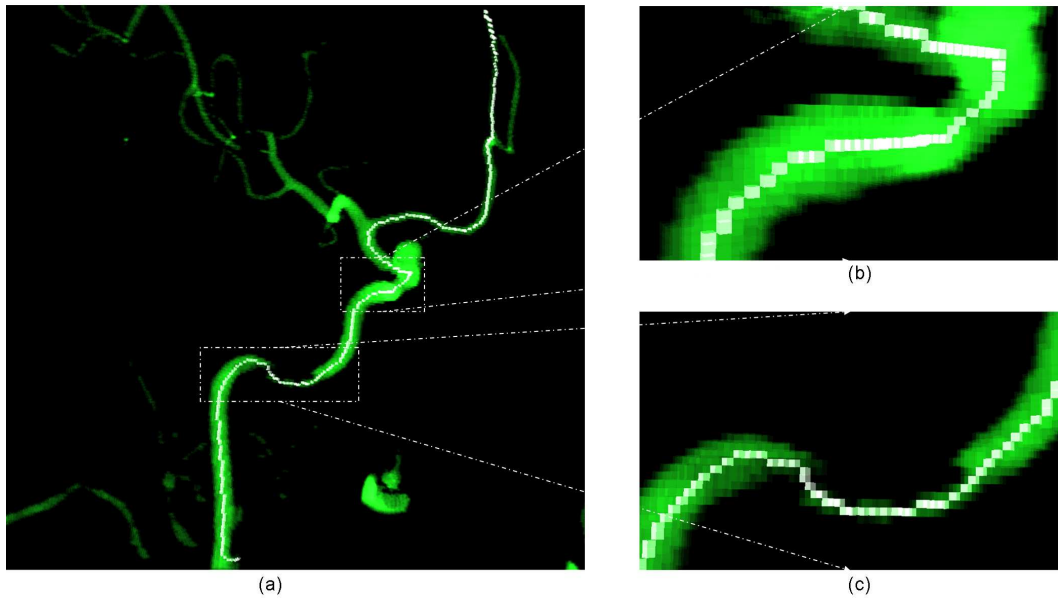


FIGURE 2.11: Trois vues de la ligne de centre principale extraite dans le cas d'un arbre vasculaire.

3.3.2 Extraction des branches

Nous cherchons à obtenir des branches centrées (comme la ligne principale) et qui se rejoignent de façon à éviter le problème mis en avant par [Bitter01], voir la Figure 2.7-(b).

Nous accordons beaucoup d'importance aux connexions entre les branches et souhaitons que ces connexions soient centrées et perpendiculaires. Cela nous permet de faire des calculs plus précis sur les diamètres des vaisseaux. Ces diamètres jouent un rôle important dans la détection et la quantification d'anévrismes, voir sections 4 et 5.

La Figure 2.12 montre l'algorithme de notre méthode pour extraire les branches du squelette. En entrée, nous avons l'ensemble des voxels de la ligne de centre principale (B_0). Nous lançons une boucle qui permet d'extraire jusqu'à 50 branches. Nous avons choisi cette limite car elle est largement suffisante dans notre cas (arbre vasculaire), mais elle peut être facilement changée si besoin est.

Pendant l'itération i de la boucle :

- Mettre la distance à la source (DFS) de tous les voxels de la branche B_{i-1} à 0.
- Exécuter l'algorithme (décrit par l'algorithme de la Figure 2.9) pour construire l'arbre des plus courts chemins.

- Le voxel le plus loin de la source, qui a la valeur maximale pour $EucDFS$, est l'extrémité (T_i) d'une branche possible.

- Extraire la branche B_i en remontant l'arbre de T_i jusqu'au premier voxel de centre.

- Si B_i est assez longue (au-dessus d'un certain seuil), on l'ajoute à la liste des branches. Sinon, on sort de la boucle. Le seuil que nous avons utilisé est lié au diamètre D_j de la branche B_j à laquelle B_i est connectée. Il est de $4 * D_j$ si $i < 10$ (les dix premières branches), $5 * D_j$ si $i < 20$ et $6 * D_j$ pour les branches suivantes. Cette valeur progressive du seuil permet de prendre en compte le fait que les branches extraites sont de plus en plus minces et donc le seuil augmente pour ne pas avoir des branches trop courtes.

- Pour calculer D_i le diamètre associé à la branche, on utilise la formule suivante :

$$D_i = 2 \times \frac{1}{M} \sum_{k=1}^M DFB[v_k], \quad \text{où } M \text{ est}$$

le nombre de voxels de la branche B_i . Une branche peut parcourir plusieurs vaisseaux (surtout les premières branches extraites, voir Figure 2.14), pour cela le diamètre est associé à la branche et non pas au vaisseau. Le diamètre est utilisé comme une approximation du diamètre des vaisseaux parcourus par la branche.

Le fait de mettre à zéro la valeur de DFS pour tous les voxels du centre (la ligne de centre principale ainsi que les branches déjà extraites) donne un arbre des plus courts chemins à ces voxels de centre et non seulement à la source, voir Figure 2.13. Cela permet de résoudre le problème des branches qui rejoignent la ligne de centre principale (ou une branche précédente) à un endroit près de la source, voir Figure 2.7-(b).

La Figure 2.14 montre la première branche (B_1) extraite en utilisant notre méthode. Nous pouvons voir un zoom sur la connexion entre B_1 et B_0 qui montre que cette connexion répond aux objectifs que nous nous sommes fixés.

La Figure 2.15 montre le squelette (la ligne principale + toutes les branches) calculé sur plu-

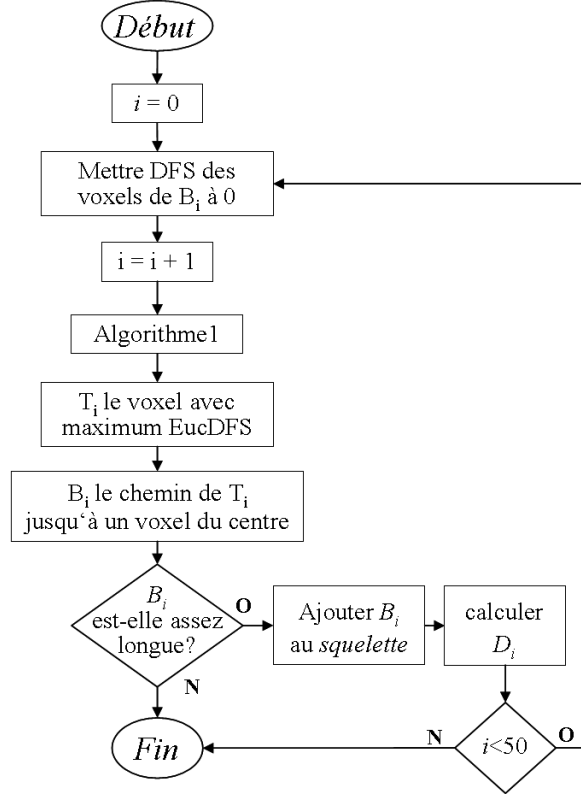


FIGURE 2.12: Algorithme pour extraire les branches.

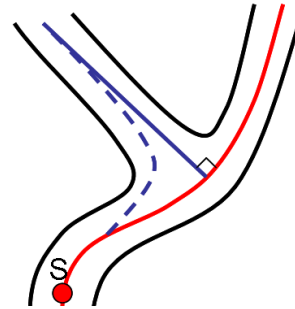


FIGURE 2.13: Ligne pointillée : branche quand la source n'est que le voxel S , ligne continue : la branche quand tous les voxels du centre (ligne rouge) sont considérés comme source.

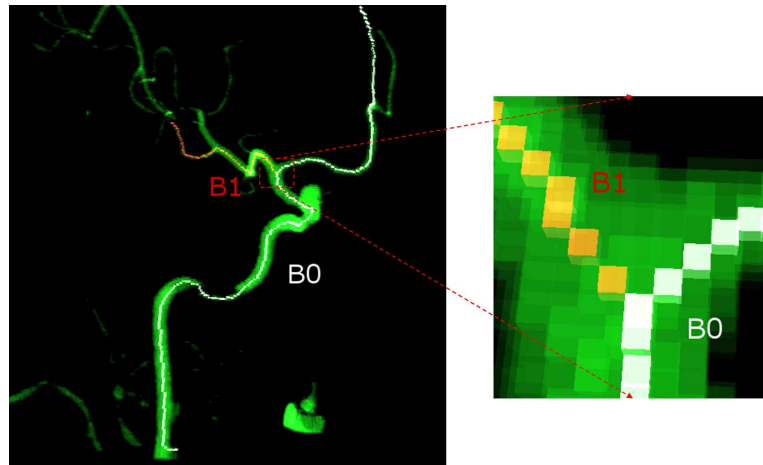


FIGURE 2.14: La première branche extraite.

sieurs exemples. Sur l'exemple (a), nous voyons l'ordre d'extraction des branches. Notre algorithme trouve les branches dans l'ordre décroissant de leur longueur même si ces branches sont connectées à une branche précédente (et non à la ligne de centre principale). À l'inverse, les algorithmes qui utilisent l'idée proposée dans [Wan02] (voir section 3.2) vont trouver les branches selon l'ordre de leur connexion : d'abord, on trouve toutes les branches connectées à la ligne de centre principale, puis, celles attachées à une des branches trouvées. Nous pouvons donc répéter l'algorithme d'extraction des branches plusieurs fois avant d'extraire une branche importante (longue) si elle n'est pas attachée directement à la ligne principale. De plus, les connexions entre les branches peuvent avoir le problème montré sur la Figure 2.7-(b).

La Figure 2.16-(a) montre un résultat d'application de cette idée pour extraire le squelette de l'arbre vasculaire de l'exemple de la Figure 2.15-(a). Pour pouvoir comparer avec notre algorithme nous avons utilisé les mêmes noms pour les branches. Il faut constater que, contrairement à notre méthode, les noms ici ne signifient pas l'ordre d'extraction. Toutes les branches sont extraites en appliquant l'algorithme proposé dans [Wan02] une seule fois. Cela explique le fait que toutes ces branches sont connectées à la ligne principale. En comparaison avec notre méthode, nous pouvons constater que la branche B3 ne correspond pas à l'extrémité (montrée sur la figure par une croix). De plus, les connexions entre les branches et la ligne principale (Figure 2.16-(b)) sont plus loin de l'emplacement intuitif que les connexions de nos branches.

Un autre point que nous pouvons constater concerne les oscillations parasites de la ligne de centre. Sur le même exemple nous obtenons une ligne de centre principale de 359 voxels, cependant l'algorithme de [Wan02] donne une ligne de centre de 373 voxels. Cela signifie que notre ligne de centre est forcément plus mince. La Figure 2.17 confirme notre constat.

En effet, quand deux voxels ont la même distance au bord (DFB) notre l'algorithme privilégie celui qui minimise la distance euclidienne à la source S , car la fonction de distance définie (cf. Eq. 2.1) prend cette distance en compte. En revanche pour l'algorithme de [Wan02], ce choix ne dépend que de (DFB), et sera arbitraire en cas d'égalité. Sur la Figure 2.17 tous les voxels ont la même valeur nulle pour DFB. Notre algorithme est capable de fournir le chemin le plus court entre deux extrémités, cependant l'algorithme de [Wan02] donne un chemin arbitraire. Cette égalité de distance au bord peut

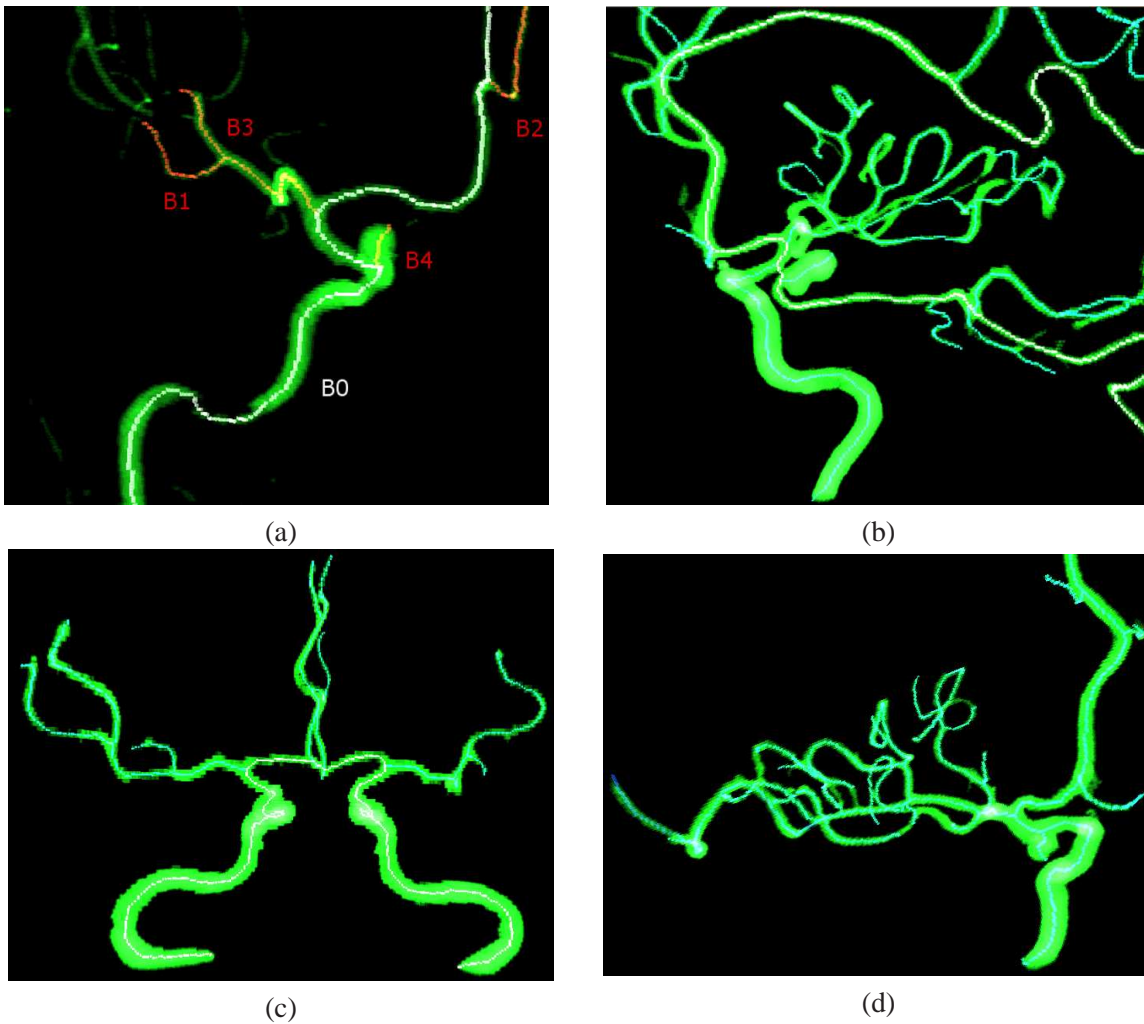


FIGURE 2.15: Les squelettes extraits pour plusieurs arbres vasculaires.

facilement arriver dans les arbres vasculaires (vaisseaux étroits).

Finalement, concernant le temps de calcul, les deux algorithmes ont presque la même performance pour l'extraction de la ligne de centre principale. Par exemple, pour l'arbre vasculaire de la Figure 2.16 et sur la même machine, notre algorithme prend 453 *ms* et celui de Wan et *al.* 594 *ms*. En revanche, pour l'extraction du squelette (la ligne principale + les branches), notre algorithme prend plus de temps (1,094 *s*) que celui de [Wan02] (16 *ms*). Cela est expliqué par le fait que nous reconstruisons l'arbre des plus courts chemins pour chaque branche, et que l'algorithme de [Wan02] a été exécuté une seule fois pour extraire les branches attachées à la ligne principale.

Notre méthode construit un squelette avec une structure d'arbre, centré mais qui n'a pas forcément la topologie de l'objet. Comme les autres méthodes basées distance, elle est raisonnable en terme de temps de calcul.

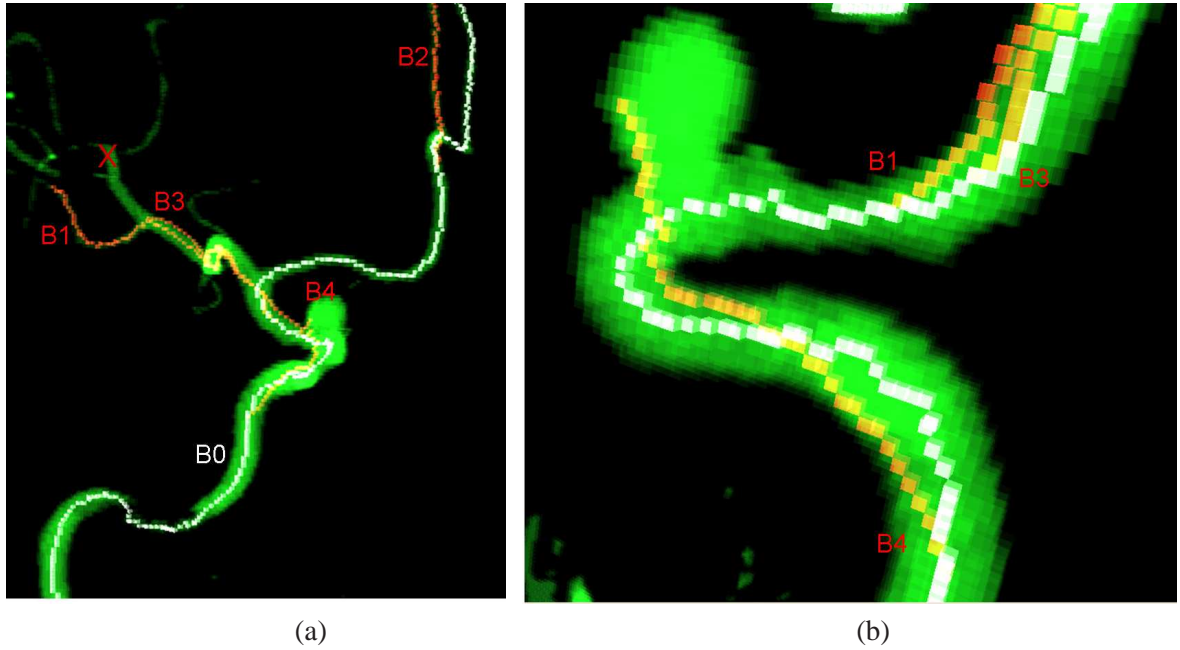


FIGURE 2.16: Le squelette extrait en utilisant l'idée de [Wan02] pour le même arbre qu'en Figure 2.15-(a).

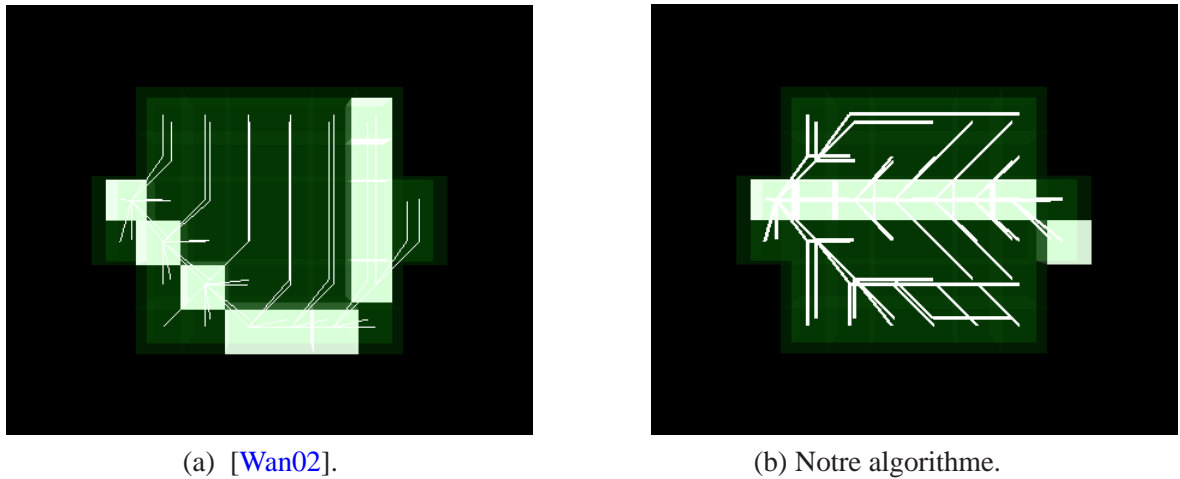


FIGURE 2.17: La ligne de centre principale dans un cas extrême (tous les voxels ont $DFB = 0$). Les lignes blanches représentent l'arbre construit.

3.3.3 La complexité de la méthode

La complexité de l'algorithme de Dijkstra pour construire un arbre des plus courts chemins est de $O(N^2)$, où N est le nombre de nœuds du graphe (le nombre de voxels dans notre cas). L'utilisation d'un tas ordonné diminue la complexité, et elle devient $O(N \log N)$. Notre modification sur l'algorithme de Dijkstra (qui consiste à changer la distance utilisée) n'entraîne aucun changement sur la complexité de l'algorithme. La complexité de la méthode pour extraire M branches est donc de $O(MN \log N)$, puisque la construction de l'arbre est faite pour chaque branche. Le tableau Table 2.1 montre les temps de calcul sur un ordinateur Intel(R) Core 2CPU 2.00 GHz.

Arbre	Nombre de voxels	Nombre de Branches	Temps ligne principale (s.)	Temps branches (s.)
(a)	22344	4	0.453	1.094
(b)	83668	21	2.281	29.875
(c)	67316	11	1.890	11.547
(d)	34626	22	0.968	11.813

TABLE 2.1: Les temps de calcul pour les arbres de la Figure 2.15

4 Détection automatique d'anévrismes

Pour détecter les anévrismes sacciformes, nous nous sommes basés sur la morphologie de ces anévrismes. Sur la Figure 2.1, nous pouvons constater que ces anévrismes ont la forme d'une hernie. Une de leurs caractéristiques est le collet (la zone rétrécie entre l'anévrisme et l'artère porteuse). La forme de l'anévrisme peut être plus complexe qu'un simple ballon. En comparaison avec un vaisseau normal nous pouvons dire que le diamètre d'un anévrisme varie énormément alors que celui d'un vaisseau est presque constant car le vaisseau a une forme plutôt cylindrique.

Pour étudier l'évolution du diamètre de chaque vaisseau nous proposons de calculer pour chaque branche extraite B_i un groupe de points $P_i = \{p_j, j \in [0, N] : N \text{ le nombre de voxels de } B_i\}$. Chaque point p_j correspond à un voxel v_j de la branche B_i et a deux coordonnées x et y que nous calculons de la façon suivante :

Le x correspond à la distance entre le voxel v_j et le voxel d'origine J de la branche B_i .

Pour calculer y , nous déterminons le plan passant par le voxel v_j et perpendiculaire à la branche B_i . Ce plan coupe la surface du vaisseau en les voxels $v_k, 1 \leq k \leq m$. Notons y_k la distance entre le voxel v_k et le voxel v_j , alors nous définissons y comme la moyenne de y_k :

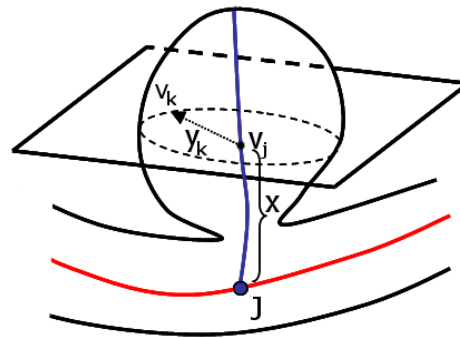
$$y = \frac{1}{m} \sum_{k=1}^m y_k.$$

Puisque les lignes de centre que nous obtenons avec notre méthode sont bien centrées, et comme les connexions entre les branches sont correctement situées, nous pouvons utiliser le groupe de points P_i pour connaître l'évolution de la branche B_i correspondante.

Nous utilisons la méthode des moindres carrées pour trouver la meilleure correspondance entre un groupe de points P_i et une fonction quadratique : $y = a + bx + cx^2$. Les coefficients de la fonction trouvée nous donnent ensuite une idée sur la forme de la branche. Si la valeur de c est très faible, nous pouvons dire que la fonction est presque linéaire et donc le diamètre de la branche ne change pas considérablement, ce qui correspond à un vaisseau. Au contraire, si la valeur de c est importante, le diamètre de la branche change considérablement, ce qui correspond à un anévrisme.

La Figure 2.19 montre les quatre groupes de points P_1, P_2, P_3, P_4 correspondant respectivement aux branches B_1, B_2, B_3, B_4 et les fonctions trouvées f_1, f_2, f_3, f_4 pour l'arbre vasculaire de la Figure 2.15-(a).

Les valeurs des coefficients a, b, c trouvés pour les différentes branches sont données dans le ta-

FIGURE 2.18: La coordonnée x et un des y_k utilisé dans le calcul du y d'un voxel.

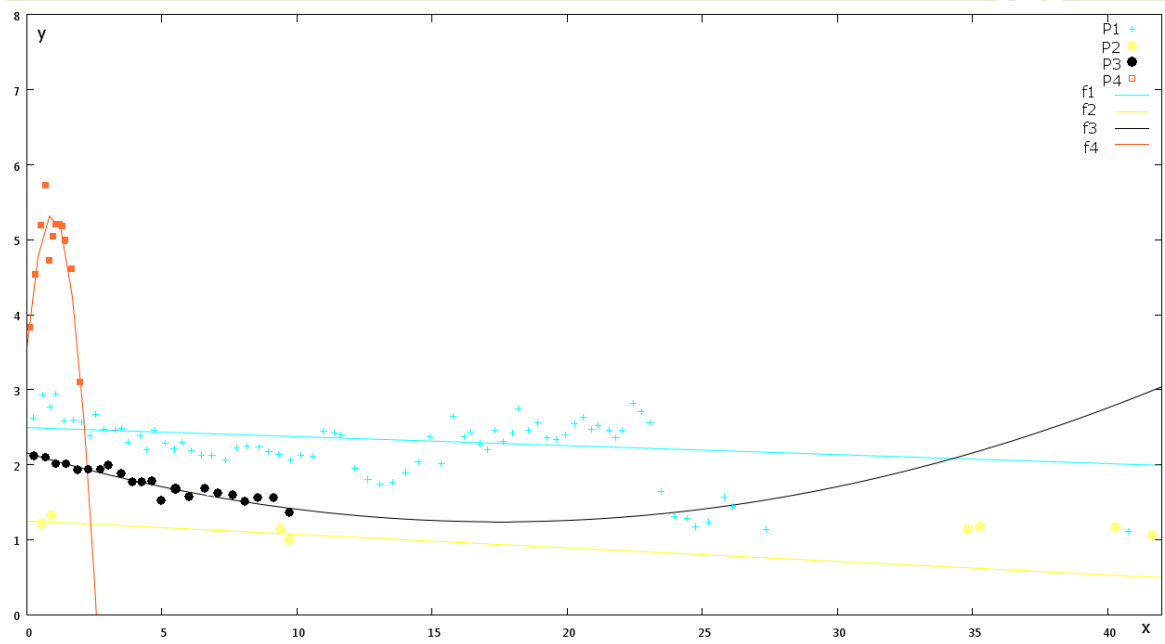


FIGURE 2.19: Les groupes de points et les fonctions trouvées pour l'arbre vasculaire de la Figure 2.15-(a).

Branch	a	b	c
B_1	2.494	-0.012	0.000
B_2	1.250	-0.018	0.000
B_3	2.156	-0.105	0.003
B_4	3.509	3.831	-2.006

TABLE 2.2: Les coefficients trouvés pour les branches du squelette de l'arbre vasculaire de la Figure 2.15-(a)

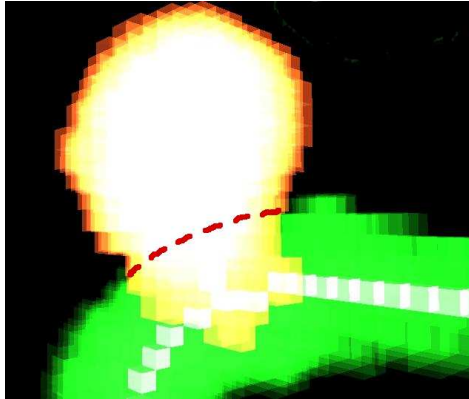
bleau 2.2 avec quatre chiffres significatifs. Ces valeurs montrent que la valeur de c est très faible pour les branches correspondant aux vaisseaux (B_1, B_2, B_3), alors qu'elle est très importante (en valeur absolue) pour la branche correspondant à l'anévrisme (B_4). De même, la valeur de b semble aussi varier, mais ce changement n'a pas la même signification. La valeur de b (quand c est négligeable) représente la pente de la ligne tracée par la fonction (les points), et nous montre donc à quel point le diamètre diminue ou augmente. Cette diminution/augmentation est due, dans la plupart des cas, au fait qu'une branche peut parcourir plusieurs vaisseaux successifs (comme la branche B_1 , Figure 2.15-(a)) et donc son diamètre varie. Une autre raison est que la segmentation peut faire mincir un vaisseau petit à petit jusqu'à disparition. Nous pouvons donc avec un simple seuillage sur c dire quelles branches correspondent aux vaisseaux et lesquelles correspondent aux anévrismes. Le seuil que nous avons choisi est de 0.2, car cette valeur maximise les vrais positifs et minimise les faux positifs pour les différents cas étudiés. Ce seuillage est fait pour chaque branche extraite, et nous gardons une liste des branches détectées comme correspondant à un anévrisme. Ces branches sont appelées les "branches anévrismales" dans la suite de ce mémoire.

Maintenant que nous avons une liste des branches anévrismales, nous pouvons passer à l'étape suivante et quantifier les anévrismes détectés.

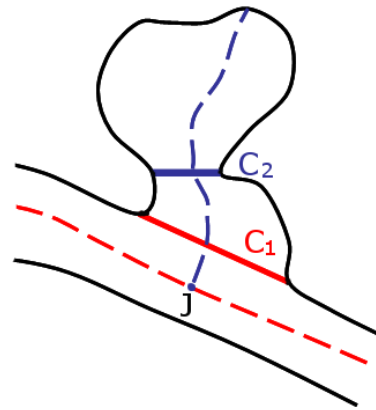
5 Quantification d'anévrismes

Dans cette étape, nous fournissons différentes mesures pour chacun des anévrismes détectés. Pour cela nous effectuons les étapes suivantes pour chacune des branches anévrismales :

Premièrement : trouver tous les voxels qui sont liés aux voxels de la branche. Les voxels sont liés selon l'arbre des plus courts chemins. Ce groupe de voxels (G) contient, en plus des voxels de l'anévrisme, des voxels de l'artère porteuse. En effet, la branche anévrismale débute à l'intérieur de l'artère porteuse, voir Figure 2.20-(a). Tous les voxels sous la ligne rouge appartiennent à l'artère porteuse.



(a) En jaune le groupe des voxels qui sont liés aux voxels d'une branche anévrismale.



(b) Les deux définitions possibles du collet d'un anévrisme.

FIGURE 2.20: Calcul du collet de l'anévrisme.

Deuxièmement : éliminer les voxels qui ne font pas partie de l'anévrisme et déterminer le collet. Il y a deux possibilités selon la définition de collet que l'on considère :

1. Définition 1 collet C1 : est la paroi de l'artère porteuse qui aurait pu exister si il n'y avait pas d'anévrisme. Dans ce cas, et pour éliminer les voxels qui ne font pas partie de l'anévrisme, on calcule pour chaque voxel de (G) sa distance à la ligne de centre du vaisseau porteur. On ne garde que les voxels dont la distance est supérieure à $DFB[J]$. J est le voxel où la branche anévrismale rejoint la branche du vaisseau porteur. Ces voxels sont ceux de l'anévrisme et seront étiquetés comme "anévrismaux".

Ensuite, le collet de l'anévrisme est la limite entre l'anévrisme et le vaisseau porteur, c'est-à-dire, l'ensemble des voxels anévrismaux qui sont à la fois surfaciques et qui ont pour voisin au moins un voxel non-anévrisimal.

2. Définition 2 collet C2 : est la zone rétrécie entre l'anévrisme et le vaisseau. Dans ce cas on travaille dans l'autre sens. On calcule d'abord le collet puis on considère que tous les voxels au-delà du collet sont étiquetés comme "anévrismaux".

Pour détecter le collet, nous cherchons la courbe fermée la plus courte parmi les voxels surfaciques de G . Nous proposons un algorithme (Algorithme 1) pour trouver une courbe fermée de voxels passant par un certain voxel s . Pour bien choisir le voxel s nous cherchons à être le plus proche possible de la zone rétrécie de l'anévrisme. Parmi les voxels de la branche anévrismale nous cherchons le voxel V le plus proche du bord de l'objet. Le voxel s est le voxel surfacique de G le plus proche de V . La courbe ainsi construite est une très bonne approximation du collet de l'anévrisme. La Figure 2.21 illustre notre algorithme.

Dans la plupart des cas, les deux définitions donnent presque le même collet. La Figure 2.20-(b) montre un anévrisme où ce n'est pas le cas.

Algorithme 1 Détection d'une courbe fermée

- 1: Appliquer l'algorithme de Dijkstra en prenant le voxel s comme source.
 - 2: Associer à chaque voxel son voisin le plus proche avec lequel il ne partage aucun prédécesseur (selon l'arbre des plus courts chemins) autre que la source. Cela nous donne un ensemble de couples des voxels, voir Figure 2.21-(d) et (e).
 - 3: Parmi les couples de voxels dont l'angle entre les deux sous-arbres est maximal, choisir celui dont la distance à la source est minimale, voir Figure 2.21-(f). L'angle maximal nous permet d'exclure les couples de voxels qui donnent des petites courbes qui n'entourent pas l'objet.
 - 4: Parcourir les voxels en remontant l'arbre de Dijkstra à partir des deux voxels du couple jusqu'à la source pour construire la courbe.
-

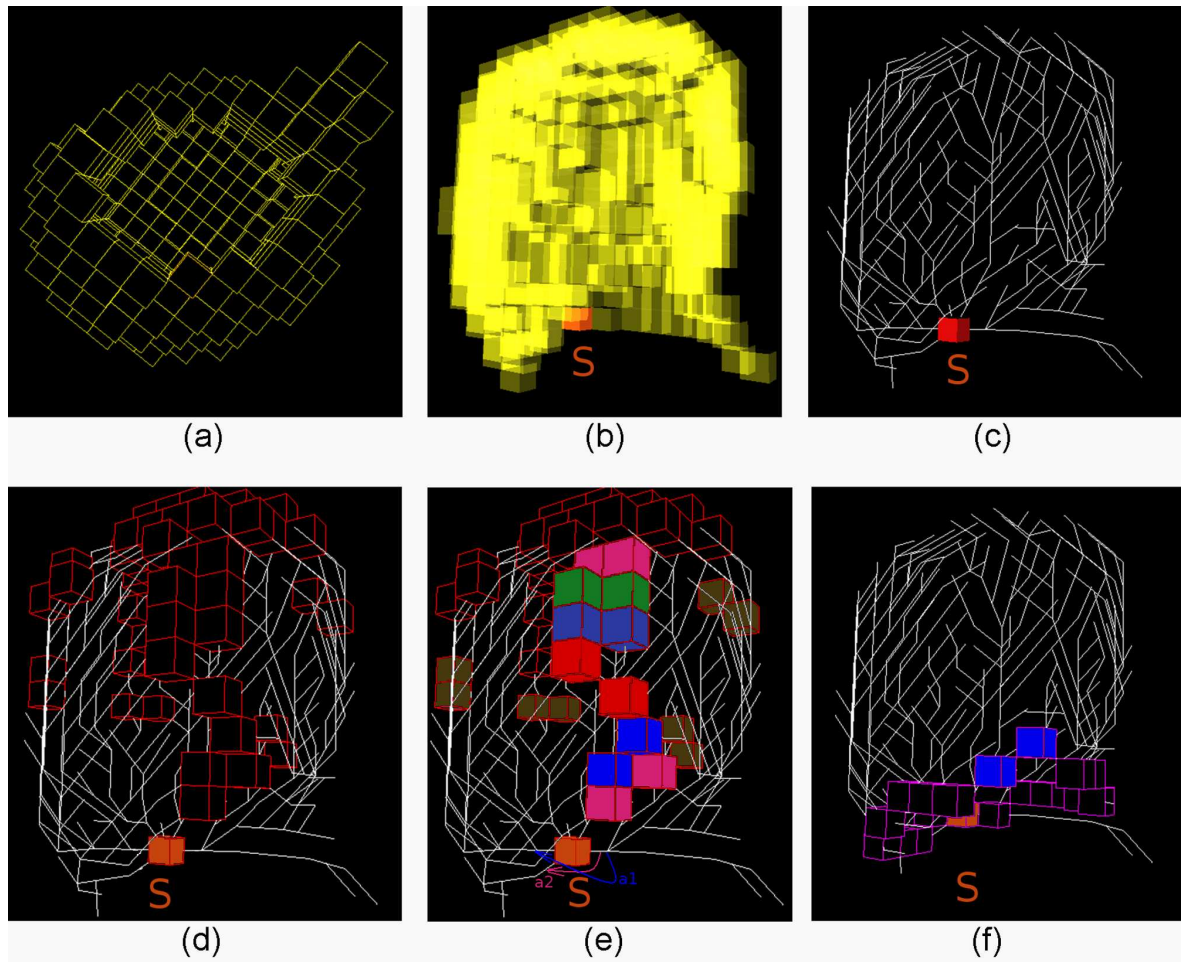


FIGURE 2.21: Détection d'une courbe fermée : (a) Le groupe des voxels surfaciques de la branche anévrismale G . (b) Le voxel source s choisi. (c) L'arbre des plus courts chemins à la source s . (d) L'ensemble des couples de voxels, (e) quelques couples affichés par des couleurs distinctes, et les angles associés à deux de ces couples (bleu et rose), les couples en marron sont ceux avec des angles minimaux. (f) Le couple choisi en bleu, et en violet la courbe fermée obtenue.

Troisièmement : fournir des mesures sur l'anévrisme. L'unité de mesure est le voxel. Les mesures que nous fournissons à l'utilisateur sont :

- la taille de l'anévrisme : nombre de voxels étiquetés "anévrismaux".

- la longueur du collet : longueur de la courbe fermée.

- la longueur maximale selon l'axe vertical de l'anévrisme (*Diam1*) : pour calculer cette longueur, nous cherchons le voxel surfacique de l'anévrisme le plus loin de l'origine *J* de la branche anévrismale. *Diam1* est la distance entre ce voxel et *J*.

- la longueur maximale selon l'axe horizontal de l'anévrisme (*Diam2*) : cette longueur correspond à la profondeur du voxel le plus profond de la branche anévrismale : nous cherchons le voxel *m* de la branche anévrismale avec la DFB la plus élevée, puis $Diam2 = 2 \times DFB[m]$.

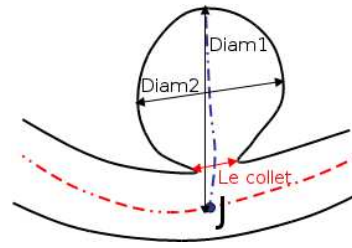


FIGURE 2.22: Mesures données sur un anévrisme.

6 Déterminer les vaisseaux porteurs d'anévrismes

Le vaisseau porteur de l'anévrisme est l'un des facteurs qui joue un rôle non négligeable dans le choix du traitement. Nous souhaitons donc trouver le nom de ce vaisseau automatiquement. La localisation d'anévrismes nous ramène au problème de mise en correspondance entre les données spécifiques au patient et nos connaissances a priori sur l'arbre vasculaire sain. Ce problème se décompose en deux parties : comment représenter un arbre vasculaire de référence, puis comment faire la mise en correspondance entre cet arbre et celui du patient.

6.1 Les connaissances a priori

Représenter les connaissances a priori sur l'arbre vasculaire revient à représenter les (des) invariants anatomiques relatifs aux vaisseaux. Plusieurs façons existent pour représenter ces invariants dépendant du type d'invariant que l'on veut représenter et de l'application que l'on vise. Les invariants anatomiques d'un arbre vasculaire peuvent être ([Passat05]) :

Quantifiables : invariants qui peuvent être représentés par des champs scalaires ou vectoriels comme l'épaisseur du vaisseau, son orientation, sa position, etc.

Non-quantifiables : invariants qui requièrent l'utilisation d'objets plus complexes pour les représenter comme : la forme des vaisseaux, leur topologie, leurs relations avec d'autres structures.

L'une des façons de modéliser les invariants quantifiables est l'utilisation d'atlas vasculaires. Ces atlas sont généralement construits d'une manière statistique. Une explication détaillée sur la création de ces atlas ainsi que leurs types peut être trouvée dans [Passat05].

Dans notre cas, nous sommes intéressés par la topologie des vaisseaux, pour cela nous représentons les connaissances a priori sous forme de graphe.

Pour construire un graphe de référence, nous avons collaboré avec un médecin spécialiste d'anatomie. De plus, nous avons constaté que dans la plupart des cas d'observation d'anévrisme, une seule moitié de l'arbre vasculaire (gauche ou droite) est représentée dans les images. Les vaisseaux qui ont un nom connu sont les vaisseaux principaux, les petits vaisseaux n'ont pas de nom spécifique, de plus

ces vaisseaux sont propres au patient et ne peuvent pas être considérés comme invariants. Heureusement, les anévrismes apparaissent dans la majorité des cas sur les vaisseaux principaux. Pour toutes ces raisons, nous nous sommes limités aux trois graphes de référence exposés à la Figure 2.23.

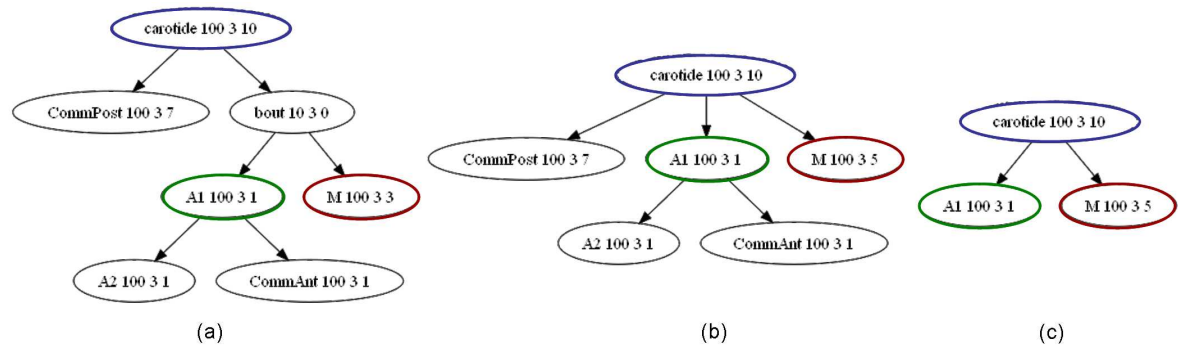


FIGURE 2.23: Les graphes de référence utilisés

Les graphes correspondent à différents niveaux de résolution de l'arbre vasculaire de référence. Ces différents niveaux sont nécessaires car certains vaisseaux peuvent ne pas apparaître dans les images. Ces trois graphes se sont avérés suffisants pour les cas que nous avons testés, cependant d'autres graphes de référence peuvent être ajoutés si besoin. L'utilisation de ces graphes ainsi que la signification des chiffres inclus dans les nœuds est expliquée dans la section suivante.

6.2 La mise en correspondance

Maintenant que nous disposons de graphes de référence, il suffit de construire un graphe correspondant au patient étudié puis de faire une mise en correspondance pour localiser l'anévrisme. Pour construire un graphe correspondant au patient, nous profitons du squelette calculé auparavant. Cette idée (squelettisation → représentation sous forme du graphe → mise en correspondance) n'est pas tout à fait nouvelle. Elle a été utilisée pour différentes structures arborescentes : les voies aériennes des poumons ([Pisupati96], [Tschirren02]), les vaisseaux hépatiques ([Charnoz04],[Charnoz05]). Dans tous les cas, une bifurcation dans l'arbre est représentée par un nœud du graphe, et les voies aériennes (ou les vaisseaux) entre les bifurcations sont représentées par des arêtes du graphe.

Deux types de mise en correspondance sont proposés dans la littérature :

- Intra-sujet : les deux graphes correspondent au même patient, et nous pouvons donc essayer de mettre en correspondance l'ensemble du graphe en prenant en compte les différences dues à la segmentation. Généralement, cette mise en correspondance intra-sujet a pour but de suivre l'évolution de l'état du patient. Par exemple, le suivi d'une tumeur [Charnoz05], ou les effets de l'inspiration [Pisupati96] pour évaluer les poumons.
- Inter-sujets : les deux graphes correspondent à différents sujets et donc on ne doit prendre en compte que les parties communes entre les sujets. Pour notre cas d'arbre vasculaire, on ne prend en compte que les principaux vaisseaux communs entre tous les êtres humains.

La mise en correspondance peut aussi profiter des propriétés des structures à mettre en correspondance. Par exemple, les longueurs et les diamètres des vaisseaux hépatiques sont utilisés pour calculer le coût d'un appariement entre deux graphes [Charnoz05], et pour choisir la meilleure solution. Ou encore, dans un travail sur les arbres vasculaires [Tang06], un coût de mise en correspondance

entre deux vaisseaux comprend les coûts d'appariements entre leurs rayons, orientations et longueurs. Même si cette utilisation semble importante dans les cas de mise en correspondance intra-sujets, elle est très difficile à réaliser entre un sujet et un graphe de référence général, car il est difficile de fournir une valeur de référence pour ces propriétés (longueur, rayon et orientation d'un vaisseau). Pour notre part, nous proposons d'autres propriétés.

Avant de détailler notre méthode pour construire un graphe à partir du squelette, nous rappelons que notre squelette a par construction une structure arborescente sans cycle. Notons également que pour chaque voxel nous avons sa distance au bord (DFB) et que les voxels de l'anévrisme sont marqués "anévrismaux".

6.2.1 Définitions

- Une *bifurcation* est un voxel du squelette qui a trois voisins dans le squelette.
- Une *extrémité* est un voxel du squelette qui a un seul voisin dans le squelette.
- Un *segment* est la suite des voxels du squelette entre deux bifurcations ou entre une bifurcation et une extrémité.
- La *longueur d'un segment* est le nombre de ses voxels.
- Le *rayon d'un segment* est la moyenne de la distance au bord de ses voxels.

6.2.2 Construction du graphe

La première étape consiste à extraire les segments de notre squelette. Pendant l'extraction nous construisons une liste de segments *Segments*, et nous stockons pour chaque segment : sa longueur, son rayon et ses deux extrémités.

Nous construisons un graphe orienté. Les nœuds du graphe représentent les segments du squelette. Une bifurcation est représentée par deux arêtes qui vont du nœud du segment avant la bifurcation vers les deux nœuds des segments après la bifurcation. Pour faciliter la mise en correspondance entre le graphe construit et les graphes de référence, et pour suivre la circulation du sang, nous souhaitons que la racine de notre graphe soit la carotide (le point d'entrée du sang). Étant donné que la carotide est le vaisseau de l'arbre vasculaire cérébral le plus large, il suffit donc de prendre le segment avec le plus grand rayon (sans prendre en compte les anévrismes).

Chaque nœud possède quatre attributs : un nom, la longueur du segment correspondant, son rayon et le nombre de fils supprimés (voir plus loin). Le nom peut être l'un des trois suivants :

- *root* : pour le segment de plus grand rayon.
- *aneurysm* : pour les segments qui correspondent à des branches anévrismales.
- *segment* : pour le reste des segments.

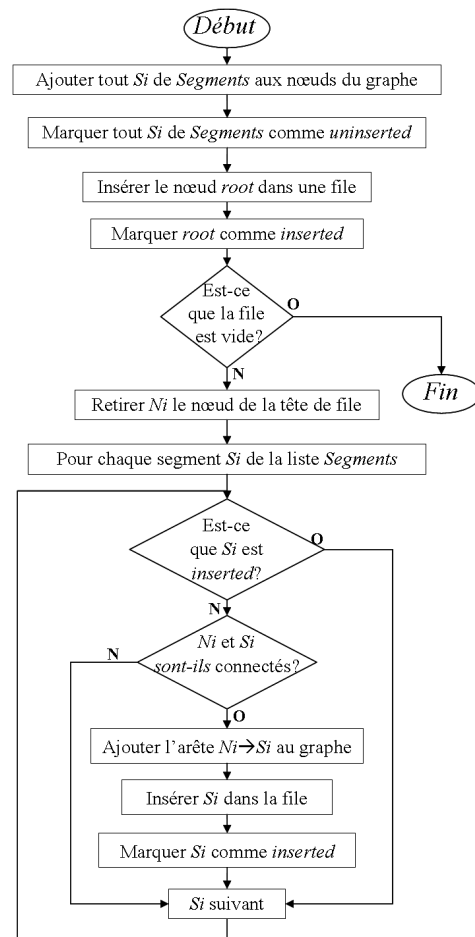


FIGURE 2.24: L'algorithme pour construire le graphe.

L'algorithme de la Figure 2.24 décrit notre méthode de construction du graphe. Tous les segments sont d'abord ajoutés comme nœuds du graphe. Puis, en commençant par la racine (le nœud *root*), nous ajoutons les arêtes en détectant les connexions entre les segments. La Figure 2.25-(a) montre le graphe correspondant à l'arbre vasculaire de la Figure 2.15-(a). Nous constatons que la quatrième propriété (nombre de fils supprimés) a la valeur zéro pour tous les nœuds sauf la racine (la valeur est 10). Et si on regarde le troisième attribut du nœud carotide sur les trois graphes de référence (Figure 2.23), nous constatons la même valeur (10) donnée à la racine. En effet, cette égalité sert à garantir lors de la mise en correspondance que la racine soit appariée avec la carotide.

Le graphe que l'on obtient contient un nœud correspondant à l'anévrisme (cf. Figure 2.25-(a)), or l'anévrisme n'existe dans aucun des graphes de référence. Pour cela nous devons supprimer ce nœud en gardant une trace de son existence. Comme nous souhaitons récupérer le nom du vaisseau porteur de l'anévrisme, la solution est de changer le nom du nœud père du nœud "anévrisme" en porteur d'anévrisme (*holdAneurysm*), le résultat de cette suppression est montré par la Figure 2.25-(b). De plus, les segments qui ont un rayon et/ou une longueur très petit correspondent aux vaisseaux qui sont très probablement spécifiques au patient et qui ne peuvent pas correspondre aux vaisseaux principaux de l'arbre vasculaire. Si ces segments sont des feuilles (n'ont pas de fils) dans le graphe, ils sont supprimés.

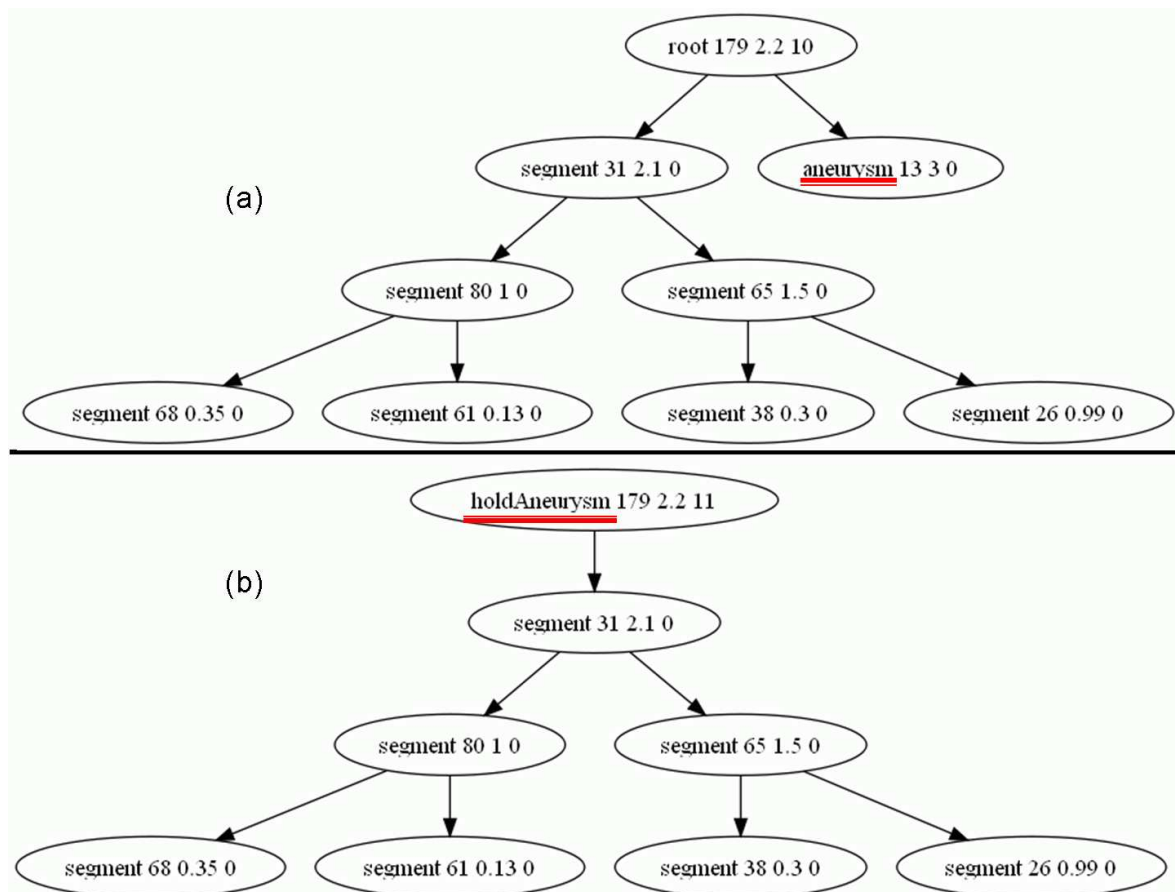


FIGURE 2.25: Le premier graphe obtenu (a), et le graphe après la suppression de l'anévrisme (b) pour l'arbre vasculaire de la Figure 2.15-(a)

Nous constatons qu'après la suppression, certains nœuds ne possèdent qu'un seul fils. Pour résoudre

ce problème, nous fusionnons les nœuds successifs en faisant attention à fusionner les propriétés comme il faut, c'est-à-dire en sommant les longueurs et sans perdre la trace de l'anévrisme (le nom *holdAneurysm* est prioritaire), voir Figure 2.26-(b).

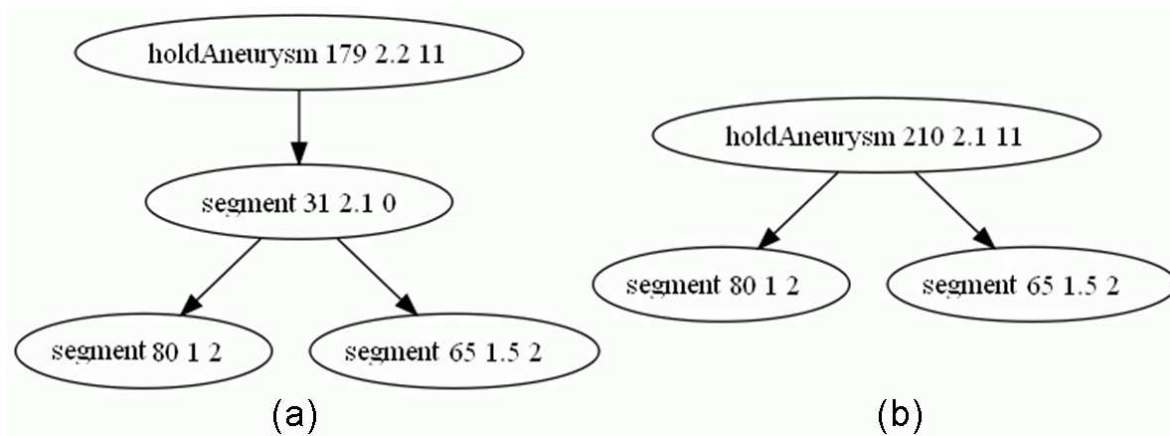


FIGURE 2.26: Le graphe après la suppression des vaisseaux de petit rayon (a), et le dernier graphe après la fusion entre nœuds successifs (b) pour l'arbre vasculaire de la Figure 2.15-(a).

Le rôle de la troisième valeur arrive à ce stade. Chaque fois qu'un nœud fils est supprimé, on augmente cette valeur de un. Cela nous aide à garder une trace des petits vaisseaux supprimés, et donc à distinguer deux vaisseaux bifurquant du même vaisseau père, l'un ayant plus de bifurcations que l'autre.

Cet attribut est pris en compte lors de la mise en correspondance, deux nœuds sont associés si les valeurs qu'ils ont pour cet attribut sont proches (différence maximale de 1). Pour cette raison, nous avons donné à cet attribut la valeur 7 pour *la composante postérieure*, 1 pour le vaisseau *A1* et 5 pour le vaisseau *M*, afin de distinguer ces trois vaisseaux lors de la mise en correspondance.

Maintenant que nous avons un graphe simple représentant l'arbre vasculaire du patient, et nos graphes de références, il suffit d'appliquer un algorithme de mise en correspondance de graphes (*graph matching*). Puisque l'arbre vasculaire du patient peut n'être qu'une partie de l'arbre vasculaire de référence, il nous faut un algorithme de mise en correspondance entre graphe et sous-graphe (*graph-subgraph matching*). Plusieurs algorithmes existent dans la littérature. Parmi les plus populaires, citons : l'algorithme d'Ullmann [Ullmann76], et l'algorithme VF [Cordella98]. Les comparaisons qui ont été faites entre ces deux algorithmes ([Cordella99], [Irniger03]) montrent l'efficacité de l'algorithme VF. Pour cette raison, nous utilisons cet algorithme pour trouver la correspondance entre le graphe du patient et les graphes de référence.

Nous commençons par essayer de trouver la correspondance avec le graphe de référence le plus détaillé (cf. Figure 2.23-(a)). Si aucune correspondance n'est trouvée, nous essayons avec le graphe de référence suivant moins détaillé (Figure 2.23-(b)), et ainsi de suite. Si aucune correspondance n'est trouvée, l'anévrisme est "non localisable".

7 Résultats

Dans cette section nous présentons les résultats de notre méthode (détection, quantification et localisation) sur un groupe de 20 patients. Le tableau 2.3 expose les données et les résultats numériques,

tandis que la Figure 2.28 montre en images les anévrismes détectés.

No	Âge	Sexe	Modalité	Paramètres	Dimensions de voxel (mm)	Nombre de voxels	Anévrisme			Temps (s.)
							Taille (mm ³)	Diam1 (mm)	Diam2 (mm)	
1	50	F	CT	-	0.42x 0.42x 0.42	98673	324.87	11.3 (10)	6.88 (6-7)	99
3	37	F	CT	-	0.42x 0.42x 0.42	27388	66	6.12	5.1	7
4	50	F	MR	Tr=23ms Te=6.9ms flip angle=20	0.39x 0.39x 0.80	43374	63.8	4.91 (4.8-5)	3.43	45
5	53	F	CT	-	0.42x 0.42x 0.42	34626	35.26	6.08 (5-6)	4.31	34
6	-					46422	9034 voxel	29.73	21.07	52
8	68	F	CT	-	0.41x 0.41x 0.45	29208	172.65	7.56 (7,5)	8,82 (9)	11
9	33	M	CT	-	0.37x 0.37x 0.45	47945	56,12	5,44	7,92	53
10	54	F	CT	-	0.39x 0.39x 0.45	42936	210	7,6 (8)	7,6	31
11	57	F	CT	-	0.38x 0.38x 0.45	27343	257,97	8,2 (6)	8,5	6
12	38	M	CT	-	0.41x 0.41x 0.45	28996	125,12	5,33 (5,7)	6,97 (6)	6
13	56	M	CT	-	0.41x 0.41x 0.45	75730	380,87	7,79 (7)	10,66 (12)	145
14	58	F	CT	-	0.36x 0.36x 0.45	27959	53,77	4,32 (4,4)	5,76 (7)	15
15	63	F	CT	-	0.38x 0.38x 0.45	57923	120,02	5,7 (6)	7,22 (7)	27
16	36	F	CT	-	0.37x 0.37x 0.45	36618	393,41	8,88 (6)	10,36 (11)	8
17	35	F	CT	-	0.38x 0.38x 0.45	40474	194,94	9,5	7,02	5
18	65	F	CT	-	0.39x 0.39x 0.45	23713	102,18	6,55	7,63	6
19	41	M	CT	-	0.42x 0.42x 0.45	24802	631,94	12,29	10,23	6

20	43	M	CT	-	0.41x 0.41x 0.45	17049	162,65	9,25	6,74	5
----	----	---	----	---	------------------------	-------	--------	------	------	---

TABLE 2.3: Résultats

Les images utilisées proviennent de deux modalités d'imagerie : l'imagerie par résonance magnétique (MR) et la tomodensitométrie (CT). Ces images ont été prises dans le centre hospitalier universitaire de Grenoble, sauf dans le cas du patient numéro 6 qui est une courtoisie de l'université de Tübingen⁶. Le temps de calcul reporté comprend à la fois le temps d'extraction du squelette, de la détection et de la quantification d'anévrismes. La localisation d'anévrisme ne prend presque pas de temps (moins d'une milliseconde). Les tests ont été faits sur une machine Pentium(R) 4 CPU 3.00GHz.

Les deux anévrismes fusiformes ne sont pas détectés par notre méthode, ce qui peut être expliqué par l'algorithme de détection que nous proposons (cf. section 4) qui est basé sur l'évolution anormale du diamètre de la branche de squelette dans l'anévrisme. Or, dans le cas d'anévrisme fusiforme, soit il n'a pas de branche (car sa longueur est inférieure au seuil), soit une branche est calculée dans l'anévrisme, mais à cause de sa forme, l'évolution du diamètre est linéaire (cf. Figure 2.27), et donc elle n'est pas détectée comme branche anévrismale. Une possibilité pour détecter des anévrismes fusiformes pourrait être d'étudier l'évolution du rayon du vaisseau porteur.

Les radiologues qui ont examiné les patients ont donné dans certains cas leur évaluation de l'anévrisme. Ces évaluations sont affichées dans le tableau 2.3 entre parenthèses à côté des quantités données par notre méthode. Nous pouvons constater que nos quantifications sont très proches de celles estimées par les radiologues. De plus, les radiologues n'arrivent pas toujours à estimer certaines mesures de l'anévrisme (cf. patients 4 et 5).

La localisation de tous les anévrismes détectés a été faite automatiquement, sauf pour les patients où les deux moitiés de l'arbre vasculaire sont présentes dans les images, ce qui ne correspond à aucun de nos graphes de référence conçus pour une seule moitié de l'arbre vasculaire. Les anévrismes que nous avons réussi à localiser sont distribués ainsi : six sur la carotide, deux sur l'artère A1, et deux sur l'artère communicante postérieure.

Dans certains cas, à cause de la segmentation, nous pouvons avoir un cycle dans l'arbre vasculaire, or notre méthode de squelettisation donne des squelettes de structure arborescente. Cela peut produire des faux positifs (détecter un faux anévrisme). La Figure 2.28-(a) montre un exemple d'un tel cas.

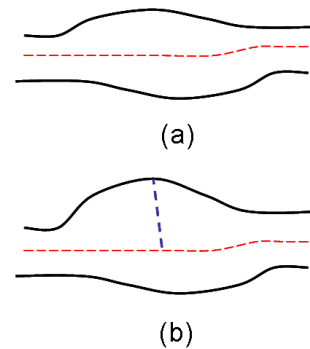


FIGURE 2.27: Anévrismes fusiformes : (a) pas de branche dans l'anévrisme, (b) une branche dans l'anévrisme.

8 Conclusion et perspectives

Au sein de ce chapitre, nous avons détaillé une méthode pour détecter, quantifier et localiser automatiquement les anévrismes sacciformes.

6. <http://www.gris.uni-tuebingen.de/edu/areas/scivis/volren/datasets/datasets.html>

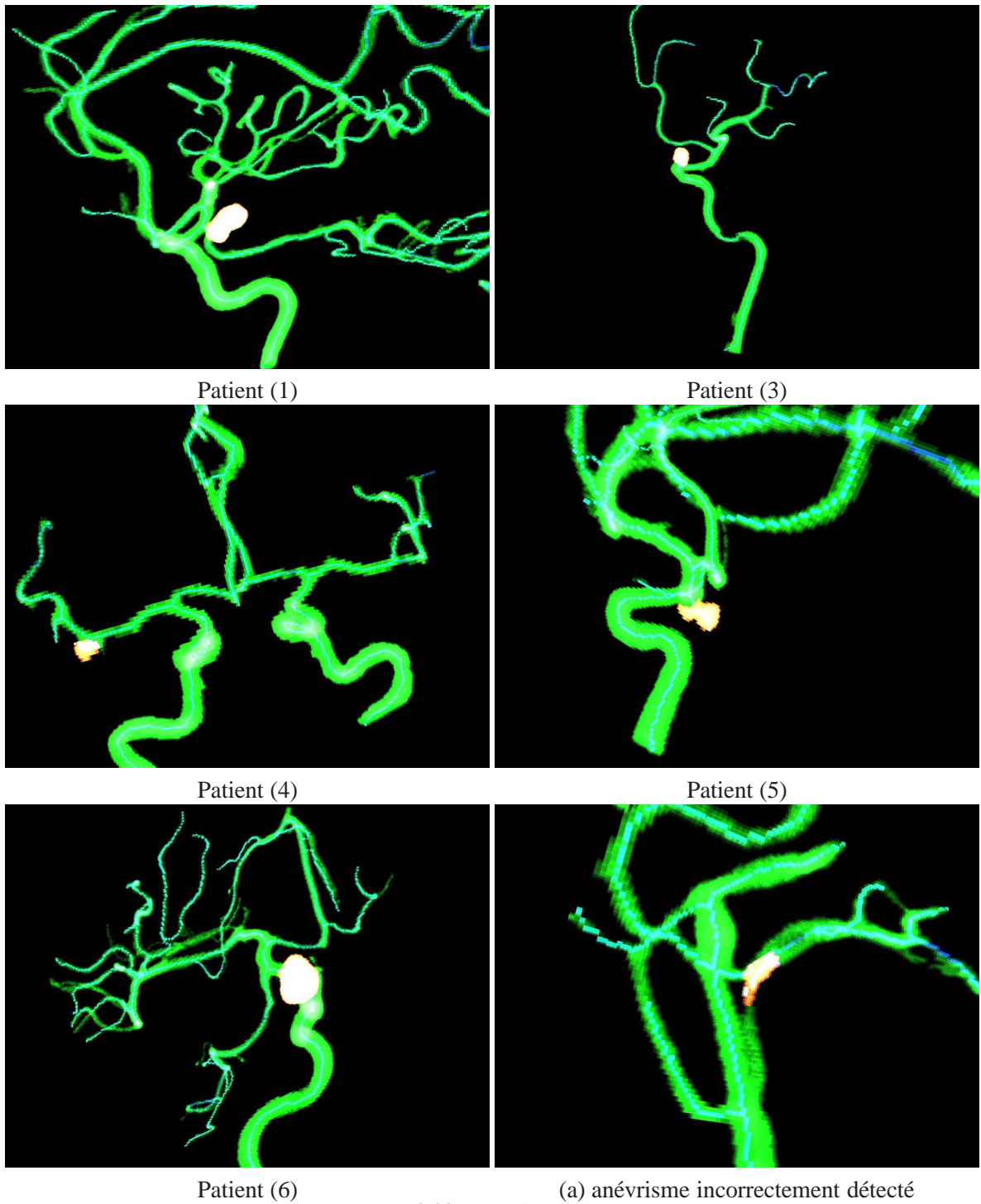


FIGURE 2.28: Anévrismes détectés

Pour cela, nous avons d'abord élaboré un algorithme de squelettisation. L'algorithme est basé sur l'algorithme de Dijkstra. Nous proposons une nouvelle fonction de distance pour renforcer la centricité du squelette. Un voxel qui est l'extrémité de l'un des vaisseaux est d'abord pris comme source S . Nous calculons l'arbre des plus courts chemins entre tous les voxels et S . La ligne de centre principale est le chemin que l'on obtient en remontant l'arbre du voxel le plus loin (pour la distance euclidienne) de la source jusqu'à la source. Cette opération (calcul d'arbre, trouver le voxel

le plus loin et remonter l'arbre pour extraire la branche) est répétée pour chaque branche du squelette, mais en mettant la distance à la source (DFS) des voxels de la dernière branche extraite à zéro. Cet algorithme nous donne un squelette connexe, mince, centré, et avec des jonctions perpendiculaires entre les branches.

Ensuite, nous avons utilisé les propriétés du squelette (centricité, jonctions entre les branches) pour détecter automatiquement les anévrismes sacciformes. Ces propriétés nous permettent d'utiliser le squelette pour étudier le comportement des diamètres des vaisseaux en les approchant par une fonction quadratique. Les coefficients de cette fonction sont ensuite utilisés pour déterminer si le changement de diamètre est négligeable (ce qui correspond à la forme cylindrique d'un vaisseau) ou s'il est important (ce qui correspond à un anévrisme). Après cette détection automatique d'anévrisme, nous calculons son collet et fournissons certaines mesures sur l'anévrisme.

Finalement, et pour localiser l'anévrisme, nous proposons de modéliser les connaissances *a priori* de la structure topologique de l'arbre vasculaire par un graphe. Le graphe représente la topologie et les connexions entre les vaisseaux principaux d'une moitié de l'arbre vasculaire. Pour couvrir plusieurs niveaux de détails, nous utilisons trois graphes de référence. Un graphe correspondant au patient est généré en utilisant le squelette déjà calculé. La localisation devient alors un problème de mise en correspondance entre graphes et nous avons choisi l'algorithme VF pour résoudre ce problème.

Les résultats sur les cas étudiés sont très prometteurs (cf. section 7), mais ils restent limités et ne couvrent pas tous les cas cliniques potentiels. Par exemple, dans notre groupe de patients, la plupart des anévrismes sont situés sur la carotide. De plus, aucun patient n'a plus d'un anévrisme. Plus de validation reste donc nécessaire. Pour la validation quantitative, les mesures de validation proposées par Firouzian *et al.* [Firouzian10] (voir Section 2.2) pourraient être utilisées.

Le squelette obtenu n'est pas homotope à l'objet, il est toujours sous forme d'arbre. Or, dans certains cas, nous pouvons avoir des cycles dans l'arbre vasculaire. Soit à cause de la segmentation, soit parce que tout l'arbre est présent dans les images et notamment le cercle de Willis (cycle de l'arbre vasculaire). Un algorithme de squelettisation qui prend en compte le cas des cycles et produit un squelette homotope sans perdre les autres qualités (centré, connexe, les jonctions) serait un grand avantage à ajouter à notre méthode. Une idée est de tester les extrémités des branches (par groupe de deux), et voir si on peut trouver entre eux un chemin (suite de voxels 26-connexe) qui soit dans l'objet. Cependant, ce chemin ne doit pas partir dans le sens des prédécesseurs des extrémités (dans l'arbre des plus courts chemins).

Dans ce chapitre, nous avons présenté une utilisation de connaissances anatomiques *a priori* pour la localisation d'anévrismes intra-crâniens. Dans ce cas précis, représenter ces connaissances sous forme d'un simple graphe était suffisant, puisque seule la relation père/fils entre les vaisseaux de l'arbre vasculaire était nécessaire. Néanmoins, dans d'autres cas d'utilisation, une représentation plus sophistiquée est demandée. Le chapitre suivant en est un exemple.

Application2 : Segmentation de maillage guidée par ontologie

1 Introduction

APRÈS avoir présenté différentes façons de modéliser des connaissances *a priori* (chapitre 1), et une utilisation de ces connaissances pour localiser les anévrismes intra-crâniens (chapitre 2), nous présentons dans ce chapitre une nouvelle utilisation des connaissances *a priori* dans le contexte de la segmentation de maillages.

Nos données d'entrée sont des maillages surfaciques correspondant aux organes anatomiques. Ces maillages sont généralement construits à partir d'images médicales (IRM, scanner, ...). Nous souhaitons ajouter à ces maillages des informations pouvant être utilisées par exemple pour l'enseignement de l'anatomie, ou la reconstruction de parties manquantes.

En effet, les organes visibles par imagerie médicale sont reconstruits à partir d'images, cf. Section 1.1. Cela produit une représentation géométrique 3D de ces organes. Pour avoir un modèle 3D complet, il faut être capable d'ajouter les organes qui ne sont pas visibles par imagerie. L'identification des parties de ces organes est une étape primordiale pour cet ajout. Par exemple, les parties visibles dans un modèle du genou sont les os, alors que les ligaments sont invisibles, voir Figure 3.1. Pour placer les ligaments au bon endroit, il faut trouver les parties

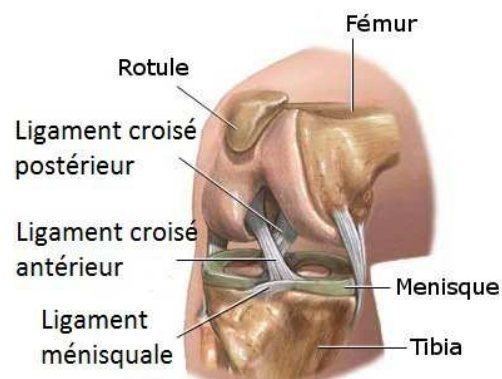


FIGURE 3.1: Modèle du genou humain.

http://www.nlm.nih.gov/medlineplus/ency/presentations/100117_1.htm

du fémur où s'insèrent/naissent les ligaments. L'ajout des parties manquantes peut être ensuite fait, par exemple en utilisant des logiciels de *sketching*.

1.1 Utilisation de maillages en imagerie médicale

Les techniques d'imagerie médicale d'aujourd'hui sont très variées afin de correspondre aux différentes caractéristiques des organes que l'on souhaite visualiser. Ces techniques permettent la création de modèles géométriques 3D pour ces organes. Ces modèles sont fondamentaux dans des domaines comme le diagnostic assisté par ordinateur, la planification d'intervention chirurgicale, et la planification de traitement.

Cependant, la création d'un modèle géométrique 3D d'un organe à partir d'images médicales est assez complexe. En effet, une image médicale contient généralement plusieurs structures anatomiques, et la création d'un modèle correspondant à un certain organe ou une certaine structure anatomique requiert l'élimination de toute autre structure de l'image. Ceci est l'objectif d'un domaine de recherche très vaste appelé la segmentation d'image.

Les différents algorithmes que l'on trouve dans le domaine de la segmentation d'image, qu'ils soient automatiques ou manuels, dépendent de la technique d'imagerie utilisée et de la structure anatomique à segmenter. Dans la plupart des cas, le résultat de ces techniques est une représentation 3D de la structure segmentée, souvent sous forme de maillage surfacique.

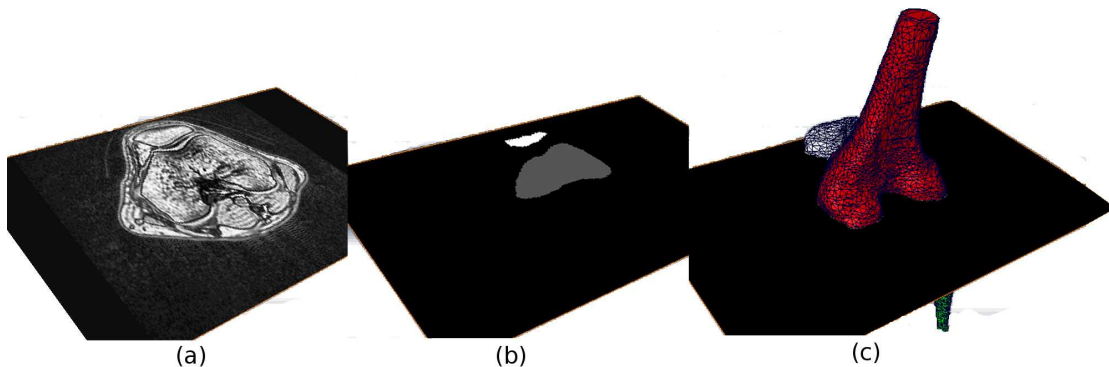


FIGURE 3.2: Reconstruction de maillages à partir d'images médicales.

La Figure 3.2 montre un exemple de segmentation d'images médicales (faite par Guillaume Bousquet), effectuée manuellement à l'aide du logiciel Amira¹. Un scanner du genou fournit une pile d'images 2D, Figure 3.2-(a). On cherche ici à obtenir un modèle 3D des os ; ceux-ci sont donc segmentés, Figure 3.2-(b). Dans cet exemple, la segmentation est faite manuellement, c'est-à-dire que l'utilisateur dessine à la main les contours des régions d'intérêt. Le logiciel construit ensuite un maillage par structure segmentée, Figure 3.2-(c).

La segmentation manuelle n'est pas la seule possibilité. Beaucoup de méthodes de segmentation automatiques et semi-automatiques ont été proposées dans la littérature : méthodes statistiques, par croissance de région, etc. Le détail de ces méthodes sort du cadre de ce mémoire. Le lecteur pourra se référer à [Lamecker08] pour un état de l'art complet.

1. <http://www.amira.com/>

1.2 Intérêt de l'apport des connaissances *a priori*

Malgré le grand développement du domaine de l'imagerie médicale, certaines structures anatomiques restent peu visibles quelle que soit la technique utilisée. Si nous prenons par exemple le cas du genou, certaines parties sont détectées par imagerie, comme les os, et donc reconstruites. En revanche d'autres comme les ligaments sont peu (ou pas du tout) visibles sur les images.

Les maillages 3D dont nous disposons correspondent principalement aux os. Pour placer les ligaments, et obtenir un modèle plus complet du genou, nous pouvons utiliser des connaissances *a priori*. En effet ces connaissances nous indiquent par exemple que le ligament croisé antérieur « naît sur la face médiale du condyle latéral et se termine dans la cavité glénoïde médiale ». Afin de pouvoir utiliser ce genre d'informations, nous allons donc chercher à identifier les différentes parties indiquées (« condyle latéral », « sa face médiale », et « la cavité glénoïde médiale ») sur les maillages reconstruits des organes visibles (dans notre cas, les os).

Le problème d'identification des parties d'un organe devient alors un problème de segmentation de maillage. Pour résoudre ce problème, nous nous référons aux connaissances que nous avons sur cet organe : le nombre de ces parties, leur position, leur forme, leur taille, les relations spatiales entre elles, etc. Puisque ces informations sont complexes et de différents types, un atlas ne suffit pas pour les représenter, ni un graphe, contrairement au cas traité précédemment de l'arbre vasculaire (chapitre 2). Il nous faut un modèle plus complexe permettant de modéliser toutes ces informations. Nous décidons d'utiliser une ontologie pour modéliser toutes les connaissances anatomiques dont nous avons besoin.

Un bref état de l'art sur les méthodes de segmentation de maillages est présenté dans la section 2. Nous expliquons ensuite (section 3) la méthode que nous proposons pour effectuer la segmentation guidée par une ontologie, et comment l'ontologie est enrichie (section 4) afin d'être utilisée (section 5) pour guider la segmentation. Les résultats obtenus sont présentés en section 6, et nous concluons en section 8.

2 Segmentation de maillages

2.1 Aperçu de l'état de l'art

2.1.1 Définitions

Un maillage M tridimensionnel est défini par un triplet $\{V, E, F\}$ de : sommets $V = \{p_i | p_i \in \mathbb{R}^3, 1 \leq i \leq m\}$, arêtes $E = \{(p_i, p_j) | p_i, p_j \in V\}$, et de faces généralement triangulaires $F = \{(p_i, p_j, p_k) | p_i, p_j, p_k \in V\}$ (mais d'autres types de faces, par exemple quadrangulaires, sont possibles). Soit S l'ensemble des éléments du maillage M , S peut correspondre à V , E , ou F . Une segmentation Σ de M est un ensemble de sous-maillages $\{M_0, M_1, \dots, M_{k-1}\}$ induit par le partitionnement de S en k sous-ensembles disjoints. La plupart des algorithmes de segmentation de maillages partitionnent les faces du maillage ($S = F$). Un sous-maillage M_i est appelé un *segment*.

Le graphe G dual du maillage M est obtenu en représentant chaque élément de S (l'ensemble des éléments du maillage) par un nœud dans G . Les arêtes sont obtenues en utilisant la relation

d'adjacence entre les éléments de S dans M . Par exemple, en prenant $S = F$, chaque face de M est alors représentée par un nœud dans G , et les arêtes du graphe relient les faces adjacentes. Nous utiliserons cette notion de graphe dual plus tard.

2.1.2 La segmentation vue comme un problème d'optimisation

La segmentation de maillages est un problème courant et commun à plusieurs domaines de recherche. Elle est notamment utile à des fins de modélisation, simplification, détection de collision, reconnaissance de forme et squelettisation de forme. Les algorithmes de segmentation de la littérature dépendent souvent de l'application envisagée. Néanmoins, [Shamir04] propose de voir le problème de segmentation comme un problème d'optimisation sous contraintes d'une fonction critère.

Fonction critère

Cette fonction dépend de la ou des caractéristique(s) que les segments doivent avoir. Si les segments doivent ressembler à une ou plusieurs primitive(s), cette fonction peut être définie comme une distance à cette primitive [Cohen-Steiner04, Mortara04, Attene06, Wu05, Yan06]. Si la courbure est la caractéristique que l'on veut capter, c'est le changement de courbure (par segment) qui doit être minimisé [Lavoue05], et ainsi de suite.

Contraintes [Shamir04] Les contraintes peuvent être de plusieurs types :

1. contraintes de cardinalité :
 - borne sur le nombre maximum et/ou minimum d'éléments par segment. Cette borne élimine les segments trop petits et/ou trop grands ;
 - borne sur le ratio entre le nombre maximum et minimum d'éléments dans tous les segments. Cette borne crée une segmentation équilibrée ;
 - borne sur le nombre maximum ou minimum de segments, si ces nombres ne sont pas fixés *a priori*. Cette borne permet aussi d'équilibrer la segmentation ;
2. contraintes géométriques : ces contraintes agissent sur les segments. Citons par exemple la surface maximale/minimale d'un segment, ou la longueur maximale/minimale du diamètre d'un segment ;
3. contraintes topologiques : elles agissent également sur les segments. Parmi celles-ci, citons la contrainte qu'un segment soit formé d'une seule composante connexe, ou qu'il soit homotope à un disque.

2.1.3 Types de segmentations

Nous distinguons deux types de segmentations [Shamir04] :

- segmentation de type plaque (*Patch-type*) : dans ce cas les segments sont homotopes à un disque. Ce type de segmentation est utilisé pour le plaquage de textures [Sander01], la simplification, et le remaillage [Cohen-Steiner04], et est illustré sur la Figure 3.6-(a) ;
- segmentation de type partie (*Part-type*) : dans ce cas les segments sont généralement plus importants et correspondent aux parties "physiques" de l'objet représenté, voir Figure 3.6-(b).

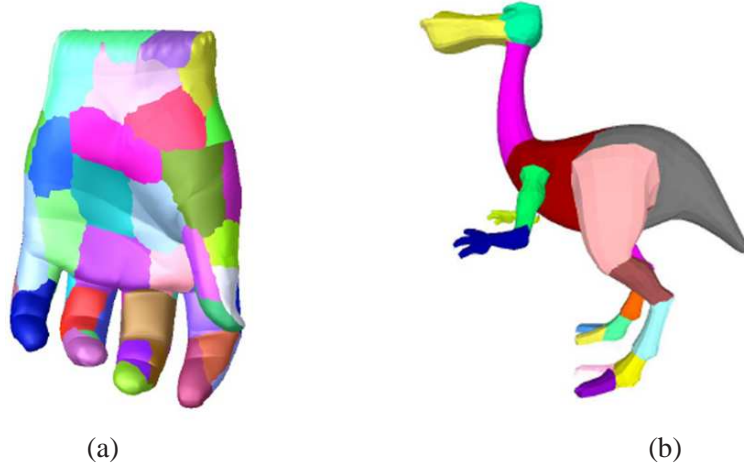


FIGURE 3.3: Types de segmentation : (a) type plaque [Sander01], (b) type partie [Katz03].

2.1.4 Méthodes de segmentation

Les catégories suivantes, selon la méthode de segmentation, ont été proposées par Ariel Shamir [Shamir04] :

- la croissance de régions ([Lavoue05] par exemple) : l'idée de ces méthodes est de faire croître une région à partir d'un élément germe placé sur le maillage. La différence entre les différents algorithmes de cette catégorie réside dans la manière de placer les éléments germes et le critère d'ajout d'un élément à une région existante.
Le résultat des algorithmes appartenant à cette catégorie dépend largement du placement et du nombre des éléments germes ;
- la segmentation hiérarchique ([Attene06, Katz03] par exemple) : dans cette catégorie, chaque face est initialement considérée comme un segment. Puis, pour chaque paire de faces adjacentes, un coût de fusion est calculé. Des fusions sont ensuite faites dans l'ordre croissant des coûts ;
- la segmentation itérative ([Cohen-Steiner04] par exemple) : dans cette catégorie, le nombre de segments est connu. La base des algorithmes appartenant à cette catégorie est l'algorithme des K -means. Cet algorithme procède de la façon suivante.

Soit (x_1, x_2, \dots, x_n) un ensemble de n observations (dans notre cas, les sommets), le but de l'algorithme des K -means est de partitionner cet ensemble en K sous-ensembles ($K \leq n$) : $S = \{S_1, S_2, \dots, S_K\}$ de façon à minimiser

$$\sum_{i=1}^K \sum_{x_j \in S_i} \|x_j - \mu_i\|^2, \text{ où } \mu_i \text{ est le barycentre des points de } S_i.$$

Cet algorithme est généralement réalisé d'une façon itérative. Le processus itératif démarre avec K éléments représentant K segments, et à chaque itération les K représentants sont recalculés. L'algorithme s'achève quand une stabilisation du résultat est acquise, c'est-à-dire que le déplacement des K représentants reste en-dessous d'un certain seuil ;

- l'analyse spectrale ([Liu04] par exemple) : dans cette catégorie le problème de segmentation de maillages est transformé en un problème de partitionnement du graphe dual G du maillage (cf. section 2.1.1). L'analyse spectrale est l'un des outils de partitionnement du graphe. Elle consiste à représenter le graphe G par une matrice d'adjacence A . Soit D une matrice diagonale où $D_{ii} = d_i$ est le degré du sommet i . Le Laplacien L du graphe G correspond à la matrice $L = D - A$. Le

graphe G peut être plongé dans l'espace R^d en utilisant les d premiers vecteurs propres de L , puis la segmentation du maillage devient un problème de partitionnement de l'espace spectral. Il existe d'autres méthodes, qui combinent par exemple la segmentation itérative et le partitionnement de graphe [Katz03]. D'autres méthodes utilisent une squelettisation du maillage [Tierny07].

Pour notre part, nous proposons une classification basée sur l'objectif (critère) de la segmentation, c'est-à-dire les caractéristiques que l'on cherche à trouver lors de la segmentation.

Algorithmes basés sur l'ajustement de primitives : le critère, pour les algorithmes de cette catégorie, est l'erreur d'ajustement entre la primitive considérée et un segment. Une seule primitive peut être utilisée pour l'ensemble des segments, comme le cylindre dans l'algorithme *Plumber* [Mortara04], ou le plan [Cohen-Steiner04]. Plusieurs primitives peuvent également être considérées : (cylindre, sphère, plan) pour l'algorithme [Attene06] (voir Figure 3.4-(a)), (cylindre, sphère, plan et balle roulante) [Wu05], ou encore toutes les surfaces quadratiques [Yan06]. Une méthode pour découvrir les primitives intrinsèques est récemment proposée par Solomon *et al.* [Solomon11]. Le fait d'utiliser la géométrie intrinsèque du maillage rend la méthode invariable aux déformations presque isométriques.

Algorithmes de groupage flou (*fuzzy clustering*) [Katz03] : dans ce type d'algorithme, au lieu d'associer chaque face du maillage à un segment, la probabilité de l'associer à plusieurs segments est calculée. Puis un algorithme de coupe minimale (*minimum cut*) est utilisé comme critère pour trouver la meilleure segmentation.

Algorithmes basés sur l'extraction de caractéristiques du maillage, comme la courbure [Lavoue05] (voir Figure 3.4-(c)), ou un ensemble de points caractéristiques et le noyau (*core*) de l'objet [Katz05] (voir Figure 3.4-(b)).

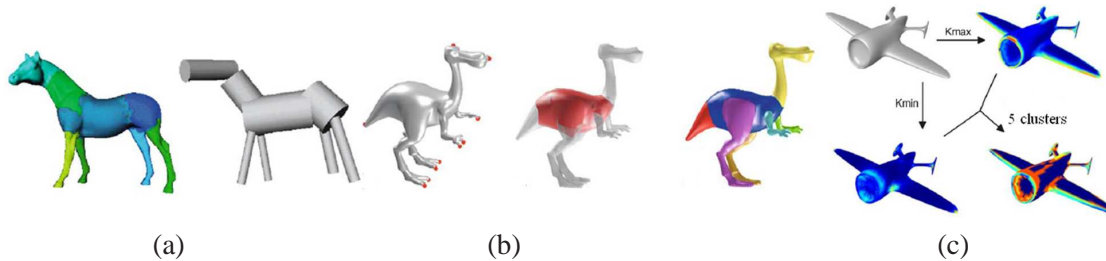


FIGURE 3.4: Différentes méthodes de segmentation : (a) Ajustement de primitives [Attene06], (b) extraction du noyau (au milieu) de l'objet [Katz03], (c) méthode basée sur la courbure : classification des sommets du maillage en 5 parties, on voit également les sommets avec une courbure maximale (K_{max}) et ceux avec une courbure minimale (K_{min}).

Algorithmes avec une dimension sémantique : très peu d'algorithmes utilisent des informations sémantiques pour créer une segmentation. Cependant, certains algorithmes ajoutent une dimension sémantique après la segmentation. Par exemple, [Mortara06] étiquette les segments résultats de l'algorithme *Plumber* [Mortara04] en prenant en compte leurs attributs géométriques. L'utilisateur de *Shape Annotator* [Attene09] choisit entre plusieurs algorithmes (ou les combine) pour segmenter le maillage. Puis, il crée un lien entre les segments du maillage et une ontologie. Cela est fait manuellement, cependant certaines caractéristiques décrivant les segments (rayon de meilleur cylindre ajusté au segment par exemple) sont calculées automatiquement. La dimension sémantique associée à la segmentation est utilisée pour la mise en correspondance entre des parties dans [vanKaick11]. L'information sémantique est dérivée grâce à un

training set de maillages segmentés et étiquetés. Cet ensemble permet de créer un *classifieur* par étiquette. Cette information est ensuite incorporée dans le processus d'étiquetage joint (*joint labeling*) pour la mise en correspondance entre un maillage et le *training set*.

Cette classification est faite pour insister sur les caractéristiques recherchées pendant la segmentation. Le lecteur pourra se référer à [Shamir08] pour un état de l'art complet, ou à [Attene06a, Chen09] pour une comparaison entre différentes méthodes de segmentation de maillages.

2.2 Algorithme de Simari et al.

Au meilleur de notre connaissance, le seul algorithme qui utilise des informations sémantiques pour générer une segmentation est celui de Simari et al. [Simari09]. La Figure 3.5 expose le schéma de cet algorithme.

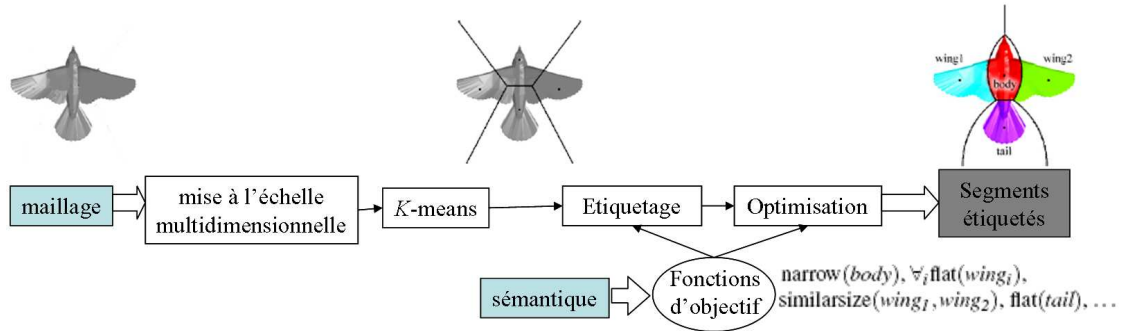


FIGURE 3.5: Schéma de l'algorithme de Simari et al. [Simari09].

Puisque la méthode que nous proposons est inspirée de cet algorithme, nous allons le détailler. Les données d'entrée de cet algorithme sont le maillage à segmenter et des informations sémantiques. Ces informations sont traduites en fonctions d'objectif. L'algorithme propose plusieurs fonctions et c'est à l'utilisateur de choisir celles qui décrivent le mieux la segmentation souhaitée. Ces fonctions d'objectif peuvent être :

- unaires : elles ne concernent qu'un segment S . Parmi les fonctions d'objectif unaires, citons : "étroit" (*narrow*), "compact" (*compact*), "plat" (*flat*) ;
- ou d'ordre supérieur à un : elles concernent deux segments ou plus. Parmi ces fonctions, citons : "de taille similaire" (*similarsize*($S1, S2$)), "perpendiculaires" (*perpendicular*($S1, S2$)), "segments convexes" (*convexparts*).

Simari et al. utilisent l'analyse par composantes principales pour calculer ces fonctions d'objectif. Voici par exemple quelques définitions de fonctions :

- $narrow(S) = 0.5 \left(\frac{S.scale_2 + S.scale_3}{S.scale_1} \right)$
- $compact(S) = 1 - narrow(S)$
- $flat(S) = 0.5 \left(\frac{S.scale_3}{S.scale_1} + \frac{S.scale_3}{S.scale_2} \right)$

avec $S.scale_1, S.scale_2, S.scale_3$ les racines carrées des valeurs propres de la matrice de covariance du segment S .

Les quatre étapes de l'algorithme sont détaillées dans les sections suivantes.

2.2.1 Étape1 : mise à l'échelle multi-dimensionnelle (*multidimensional scaling*)

Cette première étape transforme les sommets du maillage en une représentation invariante par rapport à la pose. Cette transformation dévoile la structure géométrique des éléments donnés en mesurant la ressemblance ou dissemblance entre elles. Cette dissemblance est représentée par des distances dans un espace euclidien de dimension m . Plus deux éléments sont dissemblables, plus leur distance dans cet espace est grande [Cox94]. Dans le cas des maillages, la dissemblance entre deux sommets est mesurée par leur distance géodésique, c'est-à-dire leur distance sur la surface. Cette étape est nécessaire pour rendre la segmentation insensible à la pose du maillage. Cela ne concerne que les maillages d'objets pouvant prendre plusieurs poses, c'est-à-dire principalement les objets articulés.

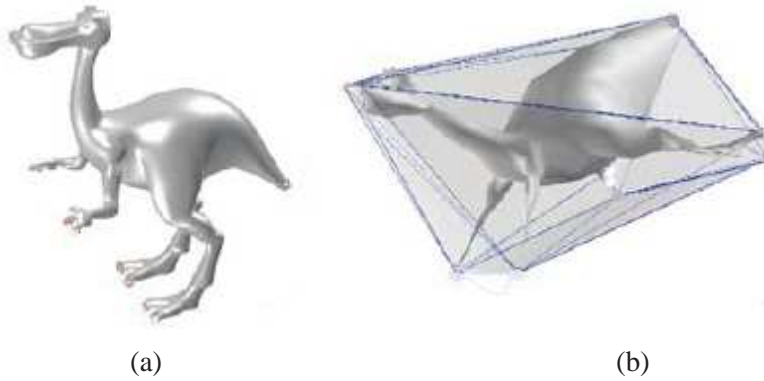


FIGURE 3.6: Mise à l'échelle multi-dimensionnelle : (a) avant, (b) après [Katz05].

2.2.2 Étape2 : algorithme des K -means (K -means clustering)

Cet algorithme (cf. section 2.1.4) est utilisé pour créer une segmentation initiale. Les centres des segments obtenus sont par construction les centres des régions d'une partition de Voronoï.

2.2.3 Étape3 : étiquetage

Dans cette étape chaque segment est étiqueté selon les informations sémantiques. Si les fonctions d'objectif utilisées pendant l'étiquetage sont unaires, les coûts de toutes les possibilités d'étiquetage sont représentés par une matrice de coût C de taille $n \times n$ où C_{ij} est le coût d'attribution au segment i de l'étiquette j . Ce coût est la somme des fonctions d'objectif (unaires) décrivant l'étiquette j . Étant donnée la matrice des coûts, l'étiquetage est alors un problème d'affectation optimale. Pour résoudre ce problème l'algorithme Hongrois [Kuhn55] est utilisé.

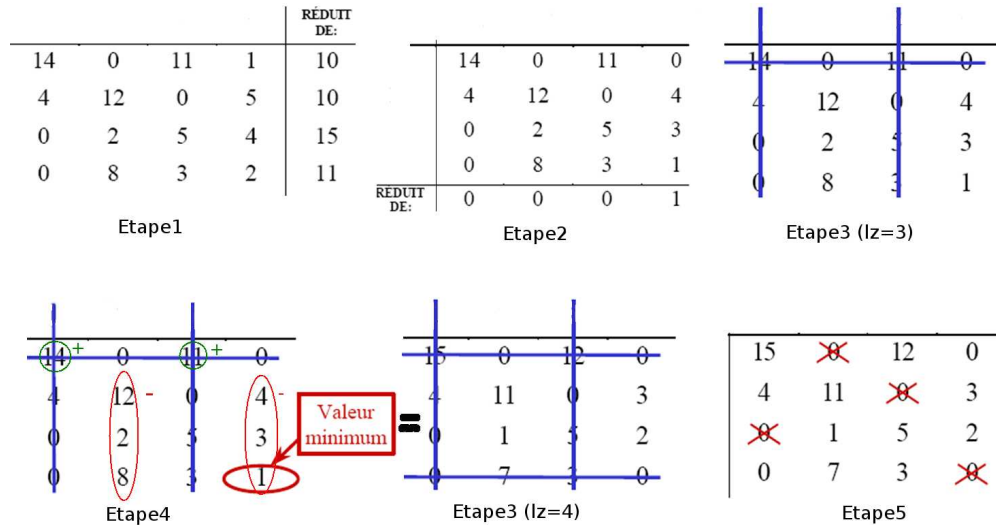
L'algorithme Hongrois est une méthode d'optimisation combinatoire. Étant donnée la matrice des coûts C , l'algorithme procède en quatre étapes, décrites dans l'Algorithme 2.

La Figure 3.7 montre un exemple d'application de cet algorithme avec une matrice des coûts de départ :

$$C = \begin{bmatrix} 24 & 10 & 21 & 11 \\ 14 & 22 & 10 & 15 \\ 15 & 17 & 20 & 19 \\ 11 & 19 & 14 & 13 \end{bmatrix}$$

Algorithme 2 L'algorithme Hongrois [Kuhn55].

- 1: Étape 1 : pour chaque ligne i de C , trouver la valeur minimale a_i et la soustraire de chaque cellule de la ligne.
- 2: Étape 2 : pour chaque colonne j de C , trouver la valeur minimale a_j et la soustraire de chaque cellule de la colonne.
- 3: Étape 3 : déterminer l_z le nombre minimal de traits sur les lignes et les colonnes nécessaires pour couvrir tous les zéros.
- 4: **if** $l_z = n$ **then**
- 5: Aller à l'étape 5.
- 6: **else**
- 7: Aller à l'étape 4.
- 8: **end if**
- 9: Étape 4 : trouver la valeur minimale non couverte par un trait. Soustraire cette valeur de toutes les cellules non couvertes. Ajouter cette valeur aux cellules à l'intersection de deux traits. Retourner à l'étape 3.
- 10: Étape 5 : déterminer la solution optimale : un segment l est donné l'étiquette c si la case de la ligne l et la colonne c contient un zéro. On commence par les lignes qui contiennent un seul zéro (ligne 2 et 3) et ainsi de suite, cela restreint les choix pour les lignes avec plusieurs zéros (lignes 1 et 4).

**FIGURE 3.7:** Exemple d'application de l'algorithme Hongrois.

Si les fonctions d'objectif sont d'ordre supérieur, les auteurs proposent un étiquetage évolutif. Cet étiquetage est fait en appliquant l'Algorithme 3. Un individu de la population est un vecteur x de longueur n tel que $x[i] = j$ signifie que le segment i possède l'étiquette j . Chaque individu propose un coût qui correspond à la valeur de la fonction d'objectif avec l'étiquetage induit par l'individu. La sélection "par tournoi" est utilisée comme technique de sélection : une petite partie arbitraire de la population est prise, et l'individu avec le meilleur coût est choisi. La génération par croisement uniforme consiste à prendre deux individus x_1, x_2 , et un vecteur b de

bits uniformément arbitraire². Un troisième individu y est généré en prenant $y[i] = x_1[i]$ si $b[i] = 1$ et $y[i] = x_2[i]$ si $b[i] = 0$, sauf si $x_2[i]$ apparaît déjà dans y venant de x_1 .

La mutation d'un individu peut être faite en choisissant d'une façon uniformément arbitraire deux entrées de l'individu et en les inter-changeant. Les valeurs prises pour les constantes *taillePopulation* et *fractionDeCroisement* sont de 20, et 80% respectivement. Pratiquement, un étiquetage optimal a été trouvé en moins de dix itérations [Simari09].

Algorithme 3 L'algorithme d'étiquetage évolutif utilisé dans [Simari09].

```

1: // Initialisation de la population
2: for  $i = 1$  to taillePopulation do
3:    $P[i] \leftarrow$  une permutation aléatoire des  $n$  étiquettes
4: end for
5: loop
6:   // Evaluation
7:   for  $i = 1$  to taillePopulation do
8:      $F[i] \leftarrow$  valeur de la fonction d'objectif pour l'étiquetage  $P[i]$ 
9:     Garder le meilleur étiquetage
10:  end for
11:   $i \leftarrow 1$ 
12:  // Génération des individus par croisement uniforme
13:  while  $i \leq \text{fractionDeCroisement} \times \text{taillePopulation}$  do
14:     $x_1 \leftarrow \text{sélectionTournoi}(P, F)$ 
15:     $x_2 \leftarrow \text{sélectionTournoi}(P, F)$ 
16:     $\hat{P}[i++] \leftarrow \text{croisementUniforme}(x_1, x_2)$ 
17:  end while
18:  // Génération des individus par mutation
19:  while  $i \leq \text{taillePopulation}$  do
20:     $x \leftarrow \text{sélectionTournoi}(P, F)$ 
21:     $\hat{P}[i++] \leftarrow \text{mutation}(x)$ 
22:  end while
23:   $P \leftarrow \hat{P}$ 
24: end loop
25: return l'individu avec la meilleur valeur pour la fonction d'objectif

```

2.2.4 Étape4 : optimisation

Une segmentation composée de n segments peut être paramétrée par le vecteur x de dimension $m = 4n$ tel que :

$$x = \{x_1, y_1, z_1, w_1, x_2, y_2, z_2, w_2, \dots, x_n, y_n, z_n, w_n\}$$

, où (x_i, y_i, z_i) sont les coordonnées 3D du centre de la région de Voronoï numéro i , et w_i est le poids associé à ce centre. Nous rappelons que la segmentation initiale est obtenue en utilisant l'algorithme K -means, et que les centres de clusters (segments) obtenus sont par construction des centres de Voronoï.

2. Toutes les valeurs possibles du vecteur ont la même probabilité d'être générées.

Ce que l'on cherche à optimiser (minimiser) est le coût total de la segmentation, c'est-à-dire une fonction d'objectif globale f_g décrivant la segmentation souhaitée. Cette fonction peut être une somme pondérée des fonctions utilisées lors de l'étiquetage, ou elle peut contenir des fonctions supplémentaires comme par exemple la fonction de segments convexes :

$$\text{convexparts}(\text{Seg}) = \left(\frac{1}{V} \left| \left(\sum_{S \in \text{Seg}} H(S) \right) - V \right| \right)^{\frac{1}{3}},$$

où $H(S)$ est le volume de l'enveloppe convexe du segment S .

Puisque les paramètres de l'optimisation (vecteur x) et la fonction d'objectif globale à minimiser sont déterminés, il suffit d'appliquer un algorithme d'optimisation. L'algorithme choisi par les auteurs est l'algorithme de *Generalized Pattern Search* [Torczon97, Audet02]. Cet algorithme procède ainsi : pour un pas de taille donnée Δ (la valeur initiale est de $0.2 \times d$, d est la diagonale de la boîte englobante du maillage), à chaque itération la fonction d'objectif globale f_g est évaluée pour tous les $2m$ voisins obtenus par l'ajout/soustraction de Δ à chaque coordonnée de x . Si l'un de ces voisins produit un coût inférieur au coût actuel, l'itération est considérée comme réussie, x est mis à jour et Δ est multiplié par un facteur d'expansion (1.2). Sinon, l'itération a échoué, il n'y a pas de mise à jour et Δ est multiplié par un facteur de contraction (0.5). L'algorithme s'arrête quand Δ tombe au dessous d'un seuil donné ($10^{-4} \times d$). La Figure 3.8 montre l'algorithme de cet algorithme.

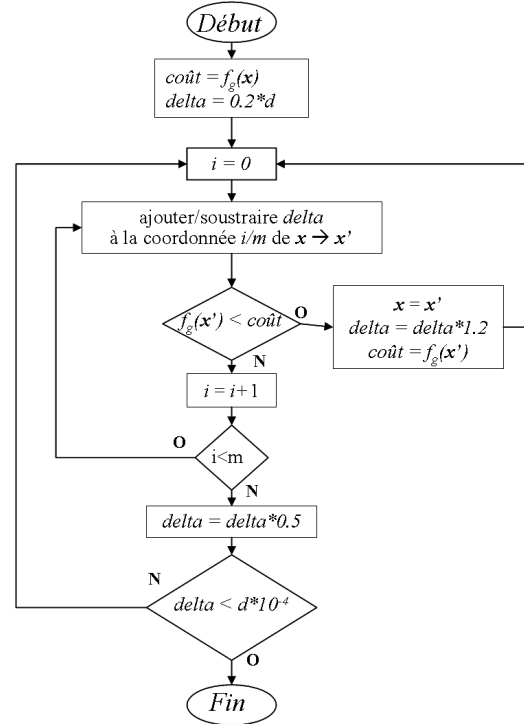


FIGURE 3.8: Algorithme de l'algorithme d'optimisation *Generalized Pattern Search* utilisé dans [Simari09].

3 Segmentation de maillage guidée par une ontologie : principe

Nous proposons une méthode de segmentation de maillages inspirée de la méthode détaillée ci-dessus [Simari09], et adaptée aux maillages issus de l'imagerie médicale. Notre méthode est totalement automatique. L'utilisateur n'a pas à choisir les fonctions qui décrivent le mieux les segments et leurs inter-relations. Nous allons chercher ces informations dans l'ontologie *MyCorporisFabrica*, décrite auparavant (Chapitre 1).

Cependant, les informations fournies par l'ontologie telle qu'elle nous a été fournie restent insuffisantes pour créer une segmentation sémantique du maillage d'une entité anatomique. Nous ajoutons donc à l'ontologie davantage d'informations, qui améliorent le processus de la segmentation.

Nos principales contributions sont :

- l'ajout d'informations géométriques sur les entités anatomiques à l'ontologie *MyCorporisFabrica* ;
- l'utilisation de l'ontologie, avec toutes les informations explicites ou implicites qui y sont contenues, pour guider la segmentation.

Nous détaillons par la suite ces deux contributions.

4 Enrichissement de l'ontologie

L'ontologie *MyCorporisFabrica* telle qu'elle nous était fournie n'était pas suffisamment complète pour pouvoir guider un algorithme de segmentation. Nous avons donc proposé d'ajouter deux types d'informations géométriques à l'ontologie : une forme géométrique approchée de l'entité anatomique, et sa taille relative.

4.1 Forme géométrique approchée

Nous proposons une description géométrique explicite pour chaque entité anatomique (organe). Après discussion avec un spécialiste d'anatomie, il s'est avéré que la plupart des organes du corps humain peuvent être approchés par des formes géométriques simples, ou une combinaison de celles-ci. Nous avons même pu limiter le nombre de ces formes à quatre primitives qui sont : la sphère, le cylindre, le plan, et le cône.

Pour cela, et grâce à l'extensibilité de l'ontologie *MyCorporisFabrica*, il suffit d'ajouter à la liste d'attributs un nouvel attribut que l'on appelle « *geometric shape* ». Nous ajoutons cet attribut à une entité qui peut être approchée par une primitive simple en lui donnant en valeur le nom de cette primitive (*plan*, *sphere*, *cylinder*, *cone*). La Figure 3.9 montre comment l'attribut géométrique est ajouté dans le cas du fémur.

Selon l'ontologie le fémur est composé de trois parties : l'épiphyse distale (*distal epiphysis*), l'épiphyse proximale (*proximal epiphysis*), et la diaphyse du fémur (*diaphysis of the femur*). Nous pouvons constater que la forme allongée de la diaphyse peut être approchée par une primitive simple (cylindre), alors que pour les deux autres parties ce n'est pas possible. Nous ajoutons donc à l'entité *Diaphysis of the femur* un attribut de type *Geometric shape* et de valeur *cylinder*.

Même si certaines entités anatomiques ne peuvent pas être directement approchées par des formes géométriques simples (les épiphyses du fémur par exemple), leurs parties ou sous-parties pourront l'être. En d'autres termes, toute entité est approchée par une primitive, ou un ensemble de primitives.

4.2 Taille relative

La relation « partie de » relie une entité anatomique (*entité mère*) avec ses composantes (les *entités filles*). D'une façon similaire, et pour améliorer la qualité de la segmentation, nous ajoutons à la liste d'attributs l'attribut *taille relative*. Cet attribut nous donne une idée sur la taille de l'*entité fille* par rapport à celle de son *entité mère*.

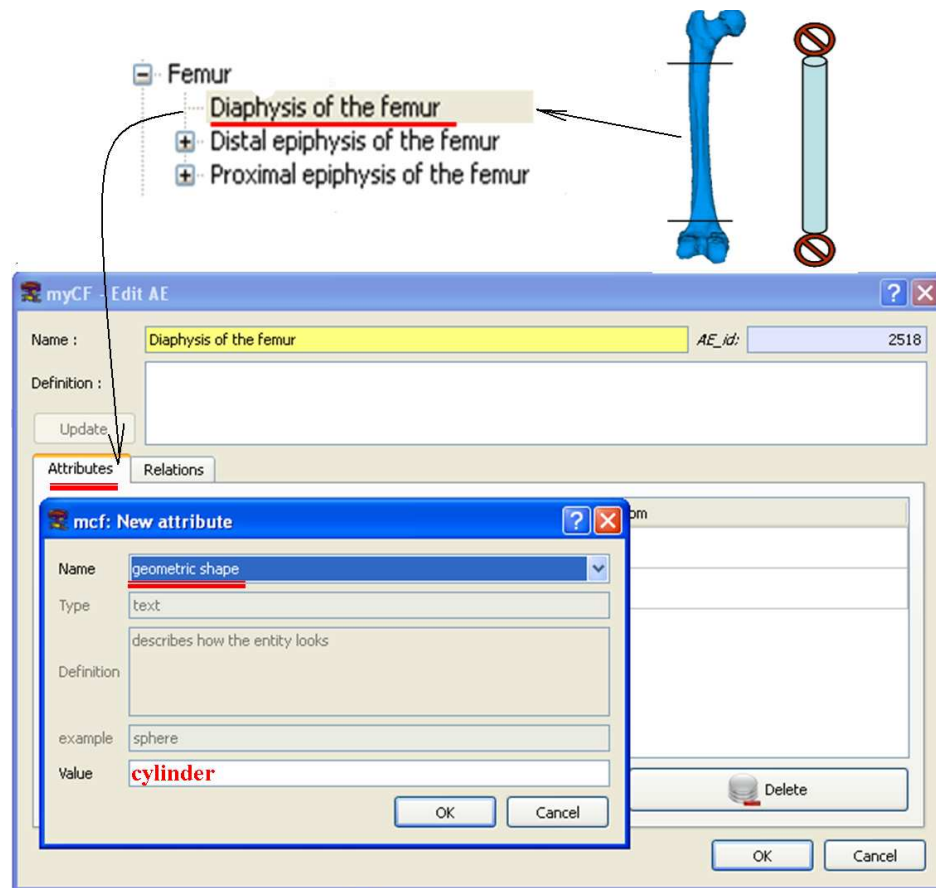


FIGURE 3.9: Ajout d'un attribut de type *Geometric shape* à l'entité *Diaphysis of the femur* en lui donnant la valeur *cylinder*.

Si nous revenons sur l'exemple du fémur, nous ajoutons à l'entité *Diaphysis of the femur* un attribut de type *relative size* et de valeur 0.45.

Pour le moment, afin de trouver la valeur à donner à un attribut *relative size*, nous comptons sur l'expertise d'un spécialiste d'anatomie et sur l'expérimentation. Une autre façon serait de calculer la valeur sur plusieurs instances segmentées manuellement et d'en prendre la moyenne.

Comme l'attribut de forme géométrique approchée, cet attribut n'est pas obligatoire pour toute entité, et son absence ne bloque pas notre méthode de segmentation. Il permet juste d'obtenir une segmentation plus fiable.

5 Utilisation de l'ontologie pour la segmentation de maillages

Nous proposons un algorithme de segmentation de maillages d'organes anatomiques, utilisant l'ontologie *MyCorporisFabrica*. Le schéma de notre méthode est illustré sur la Figure 3.10.

En entrée, nous disposons d'un maillage correspondant à une entité anatomique de l'ontologie. Cette entité est appelée par la suite l'*entité mère*. Nous disposons également de toutes les informations que nous offre l'ontologie. Ces informations peuvent être catégorisées selon deux critères : présence

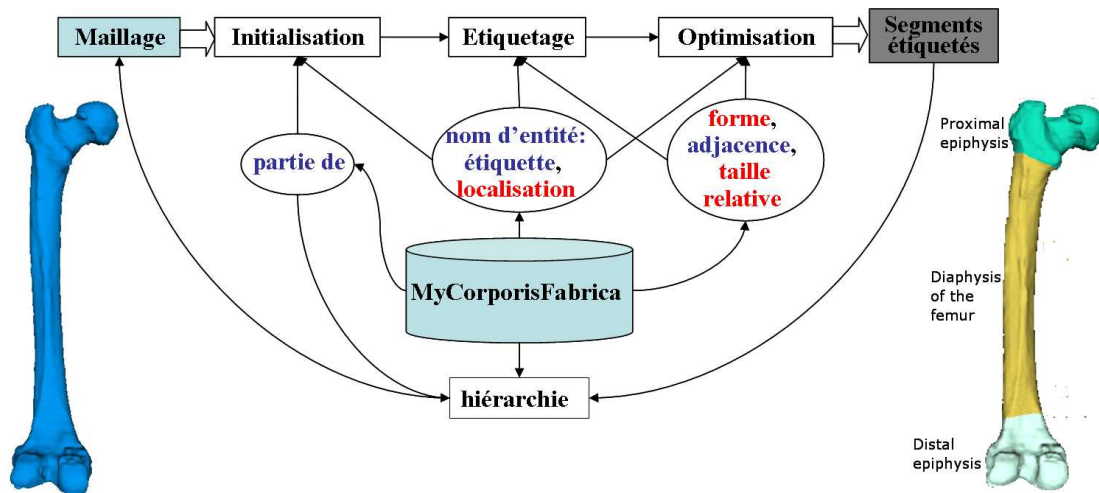


FIGURE 3.10: Algorithme de segmentation de maillages guidé par l'ontologie *MyCorporisFabrica*. Les informations qui sont toujours disponibles dans l'ontologie sont en bleu, celles qui ne le sont pas toujours en rouge.

explicite ou implicite dans l'ontologie, et disponibilité permanente ou non.

La présence d'une information dans l'ontologie peut être :

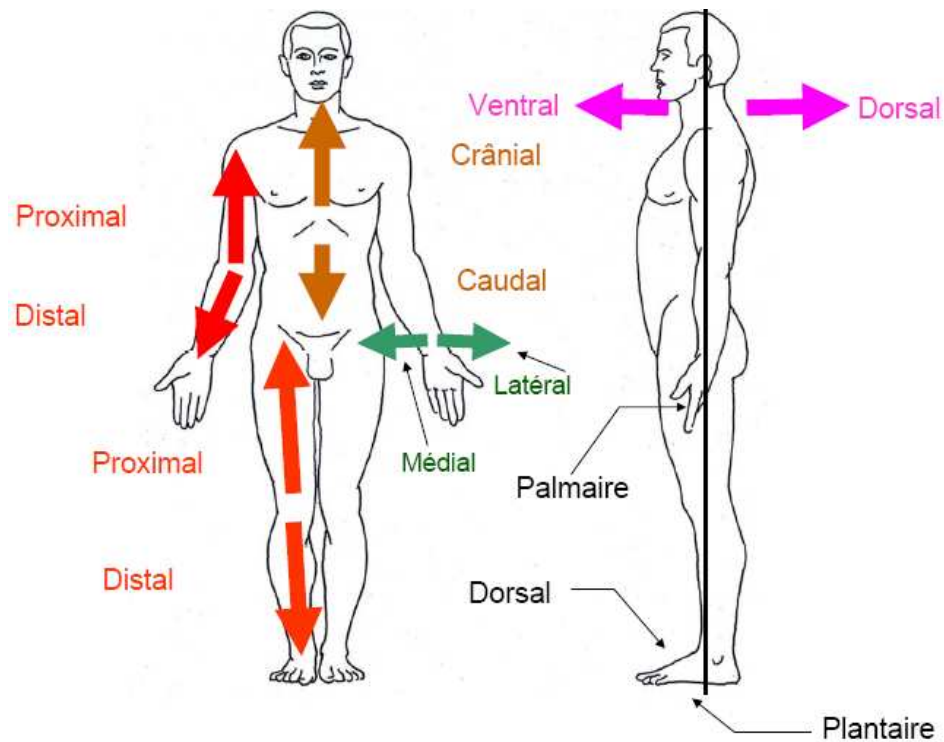
- **explicite** : c'est le cas des informations suivantes : nombre de segments, étiquette, adjacence entre entités, forme géométrique approchée et taille relative.
En effet, nous recherchons les entités anatomiques liées à l'entité mère par la relation « *partie de* ». Chacune de ces entités est une *entité fille*. Le nombre des *entités filles* constitue le nombre de segments de la segmentation résultat.
Les noms de ces entités sont les étiquettes des segments. La relation « *adjacent to* », nous permet de savoir quels segments sont voisins et lesquels ne le sont pas.
La forme géométrique et la taille relative sont des attributs que l'on a ajoutés à l'ontologie, voir la section 4 ;
- **implicite** : il s'agit des informations que l'on peut déduire à partir des noms des entités anatomiques. Les qualificatifs d'orientation médicale (proximal, distal, etc.) nous donnent une idée sur la localisation de ces entités, grâce à la nomenclature anatomique (voir Figure 3.11). Ces informations sont appelées par la suite *informations de localisation*.

De même une information peut être :

- **toujours disponible** : c'est le cas du nombre de segments, des étiquettes, et de la relation d'adjacence.
- **pas nécessairement disponible** : c'est le cas de la forme géométrique approchée, de la taille relative, et des informations de localisation.

Le schéma de la Figure 3.10 montre les différentes étapes de notre méthode. On peut constater que les informations fournies par l'ontologie sont utilisées à toutes les étapes. Nous allons détailler chacune de ces étapes et le rôle joué par l'ontologie.

3. http://fr.wikipedia.org/wiki/Système_de_référence_en_anatomie

FIGURE 3.11: Qualificatifs d'orientation.³.

5.1 Initialisation

Lors de cette étape, nous cherchons à obtenir une segmentation initiale du maillage. Cette segmentation sera ensuite optimisée. L'idée est de faire une croissance de régions à partir de points germes. Les questions qui se posent sont : combien de germes nous faut-il ? Comment les positionner ?

Le nombre de points germes correspond au nombre N de segments, qui correspond à son tour au nombre d'entités filles de l'entité mère à laquelle correspond le maillage de départ. Cette information est toujours présente dans l'ontologie.

5.1.1 Détermination des points germes

Les informations de localisation sont très utiles pour positionner les points germes, or ces informations ne sont pas toujours présentes dans l'ontologie. Nous distinguons donc deux cas.

Lorsque des informations de localisation sont disponibles :

chaque axe d'un repère de l'espace est associé à un qualificatif anatomique, voir Figure 3.12. Les informations de localisation sont traduites selon ces conventions puis utilisées pour positionner les points germes. Ainsi si nous prenons comme exemple le cas du fémur, l'interrogation de l'ontologie nous dit que l'entité *Femur* a **trois** entités filles qui ont les noms : *Diaphysis of the femur*, *Distal epiphysis of the femur*, *Proximal epiphysis of the femur*. En cherchant les mots-clés des qualificatifs d'orientation (*Proximal*, *distal*, *lateral*, *medial*, etc.) dans les noms de ces entités, nous trouvons : *Proximal*, *distal*. Puisque cette orientation est associée à l'axe z , nous cherchons donc dans notre maillage deux triangles dont les barycentres ont les valeurs minimale et maximale en z . Ces deux barycentres sont deux points germes, et correspondent aux entités *distale* et *proximale* respectivement.

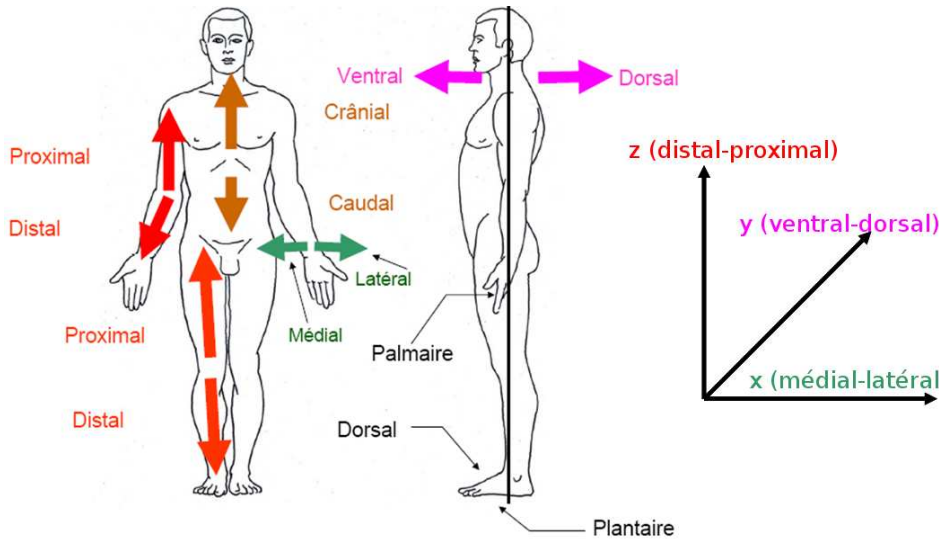


FIGURE 3.12: Utilisation des informations de localisation.

Pour le troisième point, nous prenons le barycentre des points germes déjà calculés (nous rappelons que les points germes ne sont pas forcément sur le maillage). Cette opération est répétée, dans le cas général, jusqu'à l'obtention d'autant de points germes que le nombre de segments.

Lorsque le point germe est trouvé grâce à l'information de localisation, il est directement étiqueté par le nom de l'entité fille correspondante.

Lorsqu'aucune information de localisation n'est disponible :

L'ontologie ne nous donne que le nombre N de segments. Nous appliquons l'algorithme du K -means pour obtenir les points germes. Puisque ces points ne correspondent pas aux entités anatomiques, contrairement au cas précédent, leurs étiquettes sont initialement des nombres allant de 1 à N .

5.1.2 Croissance de régions

Maintenant que les points germes sont déterminés, et pour obtenir des segments connexes, nous appliquons un algorithme de propagation par minimisation de distance (*distance-minimization flooding*), en utilisant une file de priorité comme suggéré par [Cohen-Steiner04]. L'algorithme procède de la façon suivante.

Puisque un point germe ne se trouve pas forcément sur le maillage, pour chaque point germe $p_i = (x_i, y_i, z_i)$, nous cherchons le triangle T_i le plus proche. Nous insérons les trois triangles T_j adjacents à T_i dans la file avec une priorité égale à :

$$P(T_j, T_i) = \frac{1}{w_i} d(B_j, B_i), \quad \text{où : } d \text{ est la distance euclidienne,}$$

B_j est le barycentre du triangle T_j , et B_i celui de T_i ,
 w_i est le poids associé au point p_i , initialement 1.0.

Nous donnons aussi à chaque triangle T_j un "tag" numPGerme égal à i pour savoir par rapport à quel point germe la priorité a été calculée.

Puis l'algorithme de croissance de régions consiste à retirer itérativement la tête de la file (c'est-à-

dire le triangle avec la distance minimale) jusqu'à ce que la file soit vide. Pour chaque triangle retiré nous regardons s'il est étiqueté. Si ce n'est pas le cas nous lui donnons l'étiquette du point germe indiqué par son tag *numPGerme*. Quand la file est vide, tout triangle du maillage est étiqueté.

La Figure 3.13 montre le résultat obtenu sur trois exemples : pour les deux premiers des informations de localisation étaient disponibles, et pour le troisième aucune ne l'était.

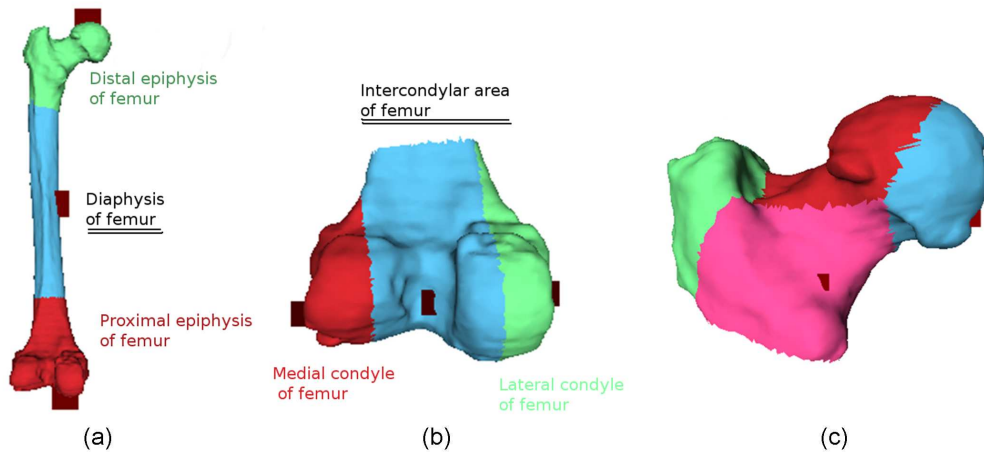


FIGURE 3.13: Segmentation initiale obtenue quand les informations de localisation sont disponibles : (a) fémur, (b) épiphyse distale du fémur, et quand il ne le sont pas (c) épiphyse proximale du fémur. Les étiquettes soulignées sont déduites des autres de manière évidente.

Si les points germes sont étiquetés grâce aux informations de localisation, nous pouvons profiter des propriétés des *entités filles* indiquées par les étiquettes lors de la croissance de régions. Plus précisément, si l'étiquette L_j donnée au point p_i indique une *entité fille* qui a une *tailleRelative_j*, on met $w_i = \text{tailleRelative}_j$. Sinon, w_i garde sa valeur initiale de 1.

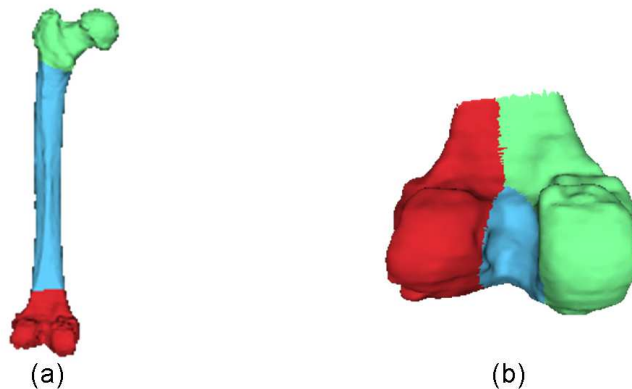


FIGURE 3.14: Segmentation initiale obtenue quand les points germes sont étiquetés, et la *tailleRelative* inscrite dans l'ontologie est utilisée comme poids : (a) fémur, (b) épiphyse distale du fémur. Les étiquettes sont identiques à celles de la Figure 3.13.

La Figure 3.14 montre le résultat d'utilisation des tailles relatives comme poids dans le cas du fémur (a) et de l'épiphyse distale du fémur (b). Cette utilisation a été possible car les points germes pour ces deux cas étaient étiquetés. En comparaison avec la Figure 3.13 où les tailles relatives n'ont

pas été utilisées, nous constatons d'une part que l'utilisation des tailles relatives donne une segmentation initiale qui est plus proche du résultat final (voir la Section 6, Figure 3.20). D'autre part, les relations d'adjacence obtenues peuvent être différentes, comme dans le cas de l'épiphyse distale du fémur. La segmentation initiale obtenue sans utiliser les tailles relatives ne satisfait pas les relations d'adjacence. En effet, l'entité « Condyle latéral » est adjacente à l'entité « Condyle médial » selon l'ontologie, et elles ne le sont pas selon la segmentation, Figure 3.13-(b). En revanche, le fait d'utiliser les tailles relatives a diminué la taille de l'entité « Aire intercondylaire » (en bleu) ce qui a créé une frontière commune entre les deux condyles et a ainsi satisfait la relation « adjacent to » entre ces deux entités.

Le contraire, c'est à dire les relations d'adjacence sont vérifiées sans utiliser les tailles relatives et ne le sont pas en les utilisant, est moins probable, car l'utilisation d'une information de plus nous rapproche logiquement davantage de la segmentation recherchée. Cependant, l'utilisation de ces tailles ne garantit pas une segmentation initiale qui satisfasse les relations d'adjacence.

5.2 Étiquetage

Le but de cette étape est de donner à chacun des segments une étiquette (nom d'entité fille). Le meilleur étiquetage est celui qui correspond le mieux aux informations fournies par l'ontologie. Ces informations comprennent :

- les informations de localisation : si elles sont disponibles, alors certains segments sont déjà étiquetés à l'issue de l'étape d'initialisation, voir Figure 3.13 ;
- les relations d'adjacence : la relation « *adjacent to* » nous dit quelles entités sont adjacentes à une entité donnée. Ces informations doivent être respectées lors de l'étiquetage ;
- l'attribut forme géométrique : le coût d'un étiquetage est défini comme la somme des coûts C_{ij} d'attribution à un segment S_i de l'étiquette L_j . Si l'étiquette L_j est une entité anatomique qui a un attribut de type forme géométrique, alors le coût correspond au coût d'ajustement de la primitive géométrique indiquée par l'attribut au segment S_i . Sinon, ce coût est nul. La façon de calculer ce coût est détaillée en section 5.3.1.

Puisque le nombre de segments à étiqueter est relativement petit (inférieur à 6 dans notre cas d'entités anatomiques), nous avons choisi de tester tous les étiquetages possibles et de choisir celui qui correspond au mieux à l'ontologie. C'est-à-dire, parmi les étiquetages qui satisfont les informations de localisation et d'adjacence, nous choisissons le moins coûteux.

Nous distinguons trois cas :

- cas 1 : le résultat de l'étiquetage global satisfait les relations d'adjacence, même lorsqu'il est déjà fait pendant l'initialisation (grâce aux informations de localisation). C'est par exemple le cas du fémur, Figure 3.13-(a). Dans ce cas, nous poursuivons normalement à l'étape suivante ;
- cas 2 : le résultat de l'étiquetage global ne satisfait pas les relations d'adjacence, mais il est fait pendant l'initialisation (grâce aux informations de localisation). C'est par exemple le cas de l'épiphyse distale du fémur, Figure 3.13-(b). Dans ce cas, et puisqu'il peut s'avérer qu'il n'y ait pas d'autre possibilité d'étiquetage global satisfaisant les informations de localisation (si un seul segment reste à étiqueter après l'initialisation), nous conservons cet étiquetage global et poursuivons à l'étape suivante, mais après avoir pénalisé le coût de cette segmentation initiale en le multipliant par un facteur f . Ce facteur est utilisé dans l'étape de l'optimisation, il est de 1 si la segmentation satisfait les relations d'adjacence et les informations de localisation, et de 10 sinon (cf. Section 5.3, Approche 2). ;
- cas 3 : aucun étiquetage global satisfaisant les relations d'adjacence n'est possible, et aucun

étiquetage n'est fait pendant l'initialisation (absence d'informations de localisation). C'est par exemple le cas de l'épiphyse proximale du fémur, Figure 3.13-(c). Dans ce cas, nous utilisons une autre technique pour créer une segmentation initiale étiquetée qui satisfasse l'adjacence.

La technique que nous utilisons afin d'obtenir une segmentation initiale étiquetée satisfaisant l'adjacence dans le cas 3 est la suivante.

1. Nous choisissons l'entité possédant le plus grand nombre de voisins (selon la relation « adjacent to ») et nous utilisons l'algorithme des *K-means* pour trouver autant de points germes que le nombre $N - 1$ d'entités restantes. Puis nous lançons l'algorithme de *minimum distance flooding* pour faire croître les régions tout en respectant l'adjacence. Avant d'étiqueter un triangle, nous vérifions que cet étiquetage ne brise pas les contraintes d'adjacence entre régions. Si on prend l'exemple de l'épiphyse proximale, l'entité qui a le plus de voisins est le cou du fémur (« neck of the femur »). Nous laissons cette entité de côté et faisons une croissance de régions à partir de seulement trois points germes (correspondant au nombre d'entités restantes). Nous constatons sur la Figure 3.15-(b) qu'il reste des triangles non étiquetés (en bleu) : ces triangles correspondent aux frontières entre les régions. Ces triangles ne sont pas étiquetés car les trois entités ne sont pas voisines, et ne doivent donc posséder aucune frontière commune.
2. Nous relançons l'algorithme de *minimum distance flooding* mais après avoir inséré tous les triangles non étiquetés dans la file de priorité. Cette étape va faire grossir d'une façon connexe cet ensemble correspondant aux frontières entre régions non supposées être voisines. L'étiquette que l'on donne à la nouvelle région est le nom de l'entité laissée de côté.
Le fait de laisser de côté l'entité avec le plus grand nombre de voisins permet d'obtenir un étiquetage qui a les plus fortes chances d'être proche de l'étiquetage correct. En effet cette entité va correspondre au segment séparant les segments qui ne doivent avoir aucune frontière en commun, et qui sera donc voisin de tous ces segments. Donc plus cette entité a de voisins, plus cet étiquetage sera proche de l'étiquetage correct, voir Figure 3.15-(c).
3. L'étape suivante consiste à trouver le dernier point germe, correspondant à la région obtenue à l'étape précédente. Nous prenons le barycentre des barycentres des triangles de cette région comme point germe de la région.
4. La dernière étape lance la croissance de régions avec tous les points germes étiquetés. Si les entités indiquées par les étiquettes ont l'attribut *taille relative*, les valeurs de ces attributs sont utilisées comme poids des points germes correspondant. La Figure 3.15-(d) montre la segmentation initiale obtenue à la fin de ce processus.

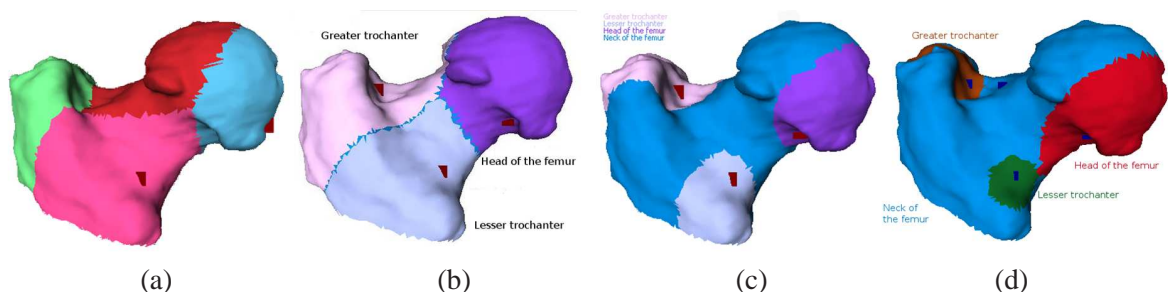


FIGURE 3.15: Technique pour obtenir la segmentation initiale la plus susceptible de satisfaire l'adjacence, appliquée à l'épiphyse proximale du fémur.

5.3 Optimisation

Dans cette étape nous cherchons à modifier la segmentation initiale de manière à optimiser la forme des segments, mais sans modifier leurs relations d'adjacence. Les paramètres de l'optimisation sont les points germes. Ces paramètres sont représentés (comme dans l'algorithme de [Simari09]) par le vecteur x de dimension $m = 4 \times N$ tel que :

$$x = \{x_1, y_1, z_1, w_1, x_2, y_2, z_2, w_2, \dots, x_N, y_N, z_N, w_N\}$$

, où (x_i, y_i, z_i) sont les coordonnées 3D du i -ème point, et w_i est le poids associé à ce point.

Le coût que l'on cherche à minimiser est :

$$\sum_i (|taille\ relative\ de\ S_i - tailleRelative_j|) + C_{i,j}$$

où :

taille relative de S_i est la taille relative du segment S_i par rapport au maillage ;

tailleRelative_j est la taille relative de l'entité anatomique indiquée par l'étiquette L_j inscrite dans l'ontologie. Si cette information n'est pas disponible, la différence n'est pas utilisée (remplacée par zéro) ;

$C_{i,j}$ est le coût d'attribution à un segment S_i de l'étiquette L_j . Nous rappelons que si l'étiquette L_j correspond à une entité anatomique qui possède un attribut de type forme géométrique, alors ce coût est le coût d'ajustement de la primitive géométrique indiquée par l'attribut au segment S_i . Sinon, ce coût est nul.

L'algorithme utilisé pour l'optimisation est l'algorithme de *Generalized Pattern Search*, voir Figure 3.8. Nous distinguons deux approches pour l'utilisation des informations fournies par l'ontologie.

Approche 1 : dans cette approche, à chaque itération de l'algorithme, les informations suivantes de l'ontologie sont utilisées (cf. Figure 3.10) :

- la localisation :
les nouveaux points germes inclus dans x' , que l'on obtient en ajoutant/enlevant Δ d'une coordonnée de x , doivent respecter les informations de localisation. Sinon, l'itération est tout de suite considérée comme échouée et on continue avec le x' suivant ;
- les relations d'adjacence :
si les nouveaux points germes respectent la localisation, l'algorithme de *distance-minimization flooding* est appliqué pour créer une partition pondérée de Voronoï (voir section 5.1). La segmentation obtenue doit respecter les relations d'adjacence. Sinon, l'itération est toute de suite considérée comme échouée et on continue avec le x' suivant ;
- la taille relative :
si l'une des *entités filles* a un attribut de type *taille relative*, nous devons vérifier que la segmentation satisfait cet attribut. Nous devons donc calculer pour ces segments (qui correspondent aux entités ayant un attribut *relative size*) le rapport de leur taille sur celui de l'ensemble du maillage (qui correspond à l'*entité mère*).

Puisqu'un tel rapport doit être calculé à chaque itération, il nous faut une manière efficace de l'estimer. Calculer la taille exacte prise par un maillage (ou un segment d'un maillage) est très coûteux en temps ; aussi, nous utilisons pour approcher cette taille l'aire du maillage. L'aire

A_i de chaque segment peut être calculée au moment de la croissance de régions : il suffit de cumuler l'aire des triangles associés à chaque segment. L'aire A de l'ensemble du maillage est la somme des aires des segments. La taille relative *taille relative de S_i* pour un segment S_i est alors approchée par $\frac{A_i}{A}$.

La taille relative est utilisée à deux reprises :

- cette taille ne doit pas être éloignée de celle inscrite dans l'ontologie *tailleRelative_j* pour l'entité anatomique correspondant à l'étiquette L_j du segment S_i . Nous calculons donc la différence $|taille\ relative\ de\ S_i - tailleRelative_j|$, si elle est supérieure à un certain seuil (de 5% dans nos tests), l'itération est considérée comme échouée et on continue avec le x' suivant ;
- la différence est aussi introduite dans le montant à minimiser. Plus précisément, le coût C_{ij} est multiplié par $|taille\ relative\ de\ S_i - tailleRelative_j|$. Ainsi, plus les tailles relatives sont respectées, moins le coût de la segmentation est élevé.

Si la segmentation correspondant au vecteur x' passe tous les tests (localisation, adjacence, taille relative) avec succès, son coût est calculé et l'algorithme continue comme précédemment expliqué en section 2.

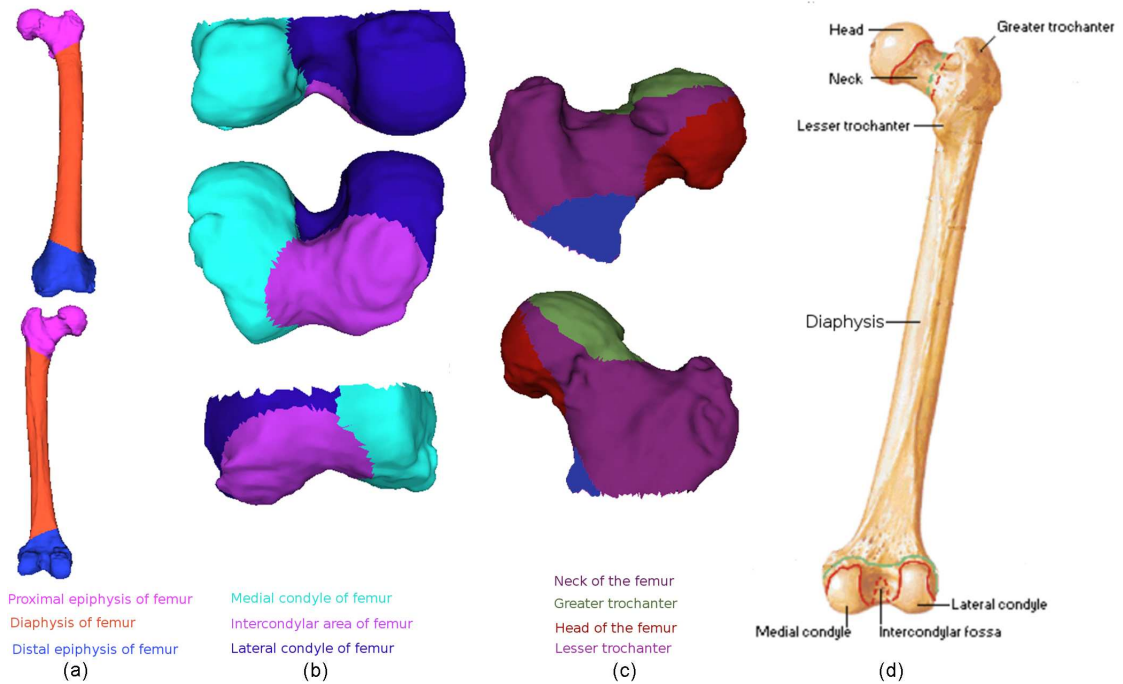


FIGURE 3.16: (a), (b) et (c) segmentations obtenues pour le fémur, l'épiphyse distale et l'épiphyse proximale respectivement en utilisant l'**Approche 1**, (d) segmentation de référence.

La Figure 3.16 montre le résultat de cette approche sur trois maillages. Le point fort de cette approche est que la segmentation finale satisfait forcément les relations d'adjacence et les informations de localisation si fournies. En revanche, elle est très dépendante de la segmentation initiale, et peut négliger des solutions très intéressantes si elles brisent, même à un triangle près, l'une des informations fournies par l'ontologie (localisation, adjacence).

Approche 2 : dans cette approche, les informations de l'ontologie sont utilisées en multipliant le coût calculé à chaque itération par un facteur f . La valeur de f est mise à 1 si la segmentation satisfait

les relations d'adjacence et les informations de localisation si celles-ci sont indisponibles, sinon on lui donne la valeur de 10. Ainsi, les segmentations insatisfaisantes sont pénalisées mais non rejetées.

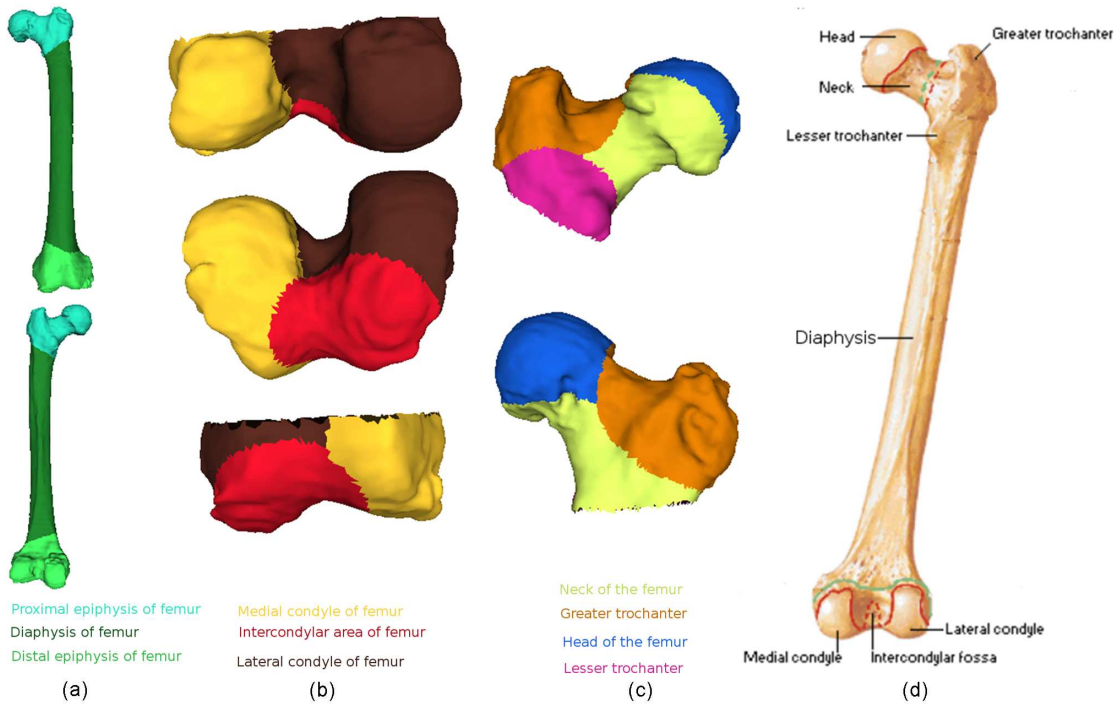


FIGURE 3.17: (a), (b) et (c) segmentations obtenues pour le fémur, l'épiphyse distale et l'épiphyse proximale respectivement en utilisant l'**Approche 2**, (d) segmentation de référence.

La Figure 3.17 montre le résultat de cette approche sur les mêmes entités anatomiques de la Figure 3.16. Dans cette approche, on peut avoir une segmentation finale qui ne satisfait pas l'une des informations fournies par l'ontologie. C'est le cas de l'épiphyse proximale (Figure 3.17-(c)) où l'adjacence n'est pas respectée : l'entité « Lesser trochanter » ne devrait pas être adjacente à l'entité « Greater trochanter », et l'entité « Head of the femur » ne devrait pas être adjacente à l'entité « Greater trochanter ».

De plus, cette approche est plus coûteuse en temps que la précédente **Approche 1**, voir Table 3.1. Le tableau Table 3.1 montre les temps de calcul sur un ordinateur muni d'un processeur Intel(R)

Entité	Approche 1		Approche 2	
	Nombre d'itérations	temps (s.)	Nombre d'itérations	temps (s.)
Fémur	19	877.84	19	1059.58
Épiphyse distale	27	266.09	23	232.45
Épiphyse proximale	24	267.38	25	354.03

TABLE 3.1: Tableau de comparaison entre les deux approches.

Core 2CPU 2.00 GHz. Nous constatons que pour le même maillage et le même nombre d'itérations, l'approche 2 prend légèrement plus de temps que l'approche 1. Cela s'explique par le fait que chaque itération dans l'approche 2 amène à un calcul du coût, alors qu'une itération dans l'approche 1 qui produit une segmentation insatisfaisante va être rejetée avant le calcul du coût.

Puisque les informations d'adjacence sont toujours présentes dans l'ontologie, nous pouvons ad-

mettre que la segmentation initiale respecte toujours ces informations. Pour cela, nous choisissons d'appliquer l'**approche 1** pour optimiser cette segmentation initiale sans perdre le respect des relations d'adjacence, et en gagnant du temps. Le point faible de l'approche 1 est qu'une segmentation très intéressante mais qui ne satisfait pas les relations d'adjacence (même à un triangle près) ne sera pas du tout prise en compte. Ainsi, nous pouvons nous retrouver avec la segmentation initiale. Pour conclure, nous pouvons dire que le choix entre ces deux approches est la réponse à la question : quelle information de l'ontologie est plus fiable et plus respectée par notre méthode : les relations d'adjacence ou le coût basé sur la forme et la taille relative.

5.3.1 Coût d'ajustement d'une primitive géométrique à un segment

Si l'étiquette L_j donnée à un segment S_i correspond à une entité anatomique qui a un attribut de type forme géométrique, le coût C_{ij} est le coût d'ajustement de la primitive géométrique indiquée par la valeur de forme géométrique au segment S_i . Nous rappelons que la valeur de cet attribut est choisie parmi les primitives suivantes : plan, sphère, cylindre, et cône. Nous avons testé deux façons de calculer/approcher le coût d'ajustement d'une de ces primitives à un segment de maillage.

En utilisant l'analyse par composantes principales (ACP)

Simari et al. [Simari09] utilisent des fonctions d'objectif basées sur l'analyse par composantes principales. Ils calculent la matrice de covariance des sommets du segment S . Soit \vec{V} le vecteur de p sommets du segment S , la matrice de covariance de ce vecteur est définie ainsi :

$$\Sigma_X = \begin{bmatrix} Var(v_1) & cov(v_1, v_2) & \dots & cov(v_1, v_p) \\ cov(v_2, v_1) & \ddots & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ cov(v_p, v_1) & \dots & \dots & Var(v_p, v_p) \end{bmatrix}$$

, où $Var(v) = \sum_i ((\vec{v}) - \bar{\vec{v}})((\vec{v}) - \bar{\vec{v}})^T$ et $cov(v_1, v_2) = 0$ quand v_1 et v_2 sont indépendants. Cette matrice est symétrique, et ses valeurs propres sont nulles ou positives.

Les racines carrées des valeurs propres ($scale_1$, $scale_2$, $scale_3$) de cette matrice sont ensuite utilisées pour définir les fonctions d'objectif. Certaines de ces fonctions peuvent être utilisées pour décrire d'une façon approximative des primitives géométriques. Voici les primitives géométriques existantes dans l'ontologie ainsi que les fonctions d'objectif que nous reprenons pour évaluer le coût d'approximation par un segment S .

- Cylindre : puisqu'un cylindre possède une direction privilégiée et que les deux directions orthogonales sont indifférenciées (section transversale circulaire), nous pouvons choisir la fonction $narrow(S)$ pour estimer le coût d'approcher le segment S par un cylindre. Nous rappelons la définition de cette fonction :

$$narrow(S) = 0.5 \left(\frac{S.scale_2 + S.scale_3}{S.scale_1} \right)$$

La direction privilégiée correspond à une valeur propre élevée par rapport aux autres, c'est-à-

dire :

$$\begin{aligned} S.scale_1 \gg S.scale_2 \text{ et } S.scale_1 \gg S.scale_3 &\Rightarrow \frac{S.scale_2}{S.scale_1} \ll 1, \text{ et } \frac{S.scale_3}{S.scale_1} \ll 1 \\ &\Rightarrow 0.5 \frac{S.scale_2}{S.scale_1} + 0.5 \frac{S.scale_3}{S.scale_1} \ll 1 \\ &\Rightarrow narrow(S) \ll 1 \end{aligned}$$

La valeur de cette fonction sera donc très petite dans le cas d'un cylindre allongé et fin.

- Sphère : dans ce cas, il n'y a pas de direction privilégiée, et la fonction $compact(S)$ est utilisée. Elle est définie de la manière suivante :

$$compact(S) = 1 - narrow(S)$$

Contrairement au cas d'une direction privilégiée, ici les trois valeurs propres sont très proches, c'est-à-dire :

$$\begin{aligned} S.scale_1 \approx S.scale_2 \text{ et } S.scale_1 \approx S.scale_3 &\Rightarrow \frac{S.scale_2}{S.scale_1} \approx 1, \text{ et } \frac{S.scale_3}{S.scale_1} \approx 1 \\ &\Rightarrow 0.5 \frac{S.scale_2}{S.scale_1} + 0.5 \frac{S.scale_3}{S.scale_1} \approx 1 \\ &\Rightarrow narrow(S) \approx 1 \\ &\Rightarrow 1 - narrow(S) \ll 1 \\ &\Rightarrow compact(S) \ll 1 \end{aligned}$$

La valeur de cette fonction est donc nulle dans le cas d'une sphère parfaite.

- Plan : un des trois axes principaux (correspondant à la valeur propre la moins élevée) est presque inexistant par rapport aux deux autres, nous utilisons donc la fonction :

$$flat(S) = 0.5 \left(\frac{S.scale_3}{S.scale_1} + \frac{S.scale_3}{S.scale_2} \right)$$

L'axe inexistant correspond à une valeur propre très petite par rapport aux deux autres, c'est à dire :

$$\begin{aligned} S.scale_3 \ll S.scale_1 \text{ et } S.scale_3 \ll S.scale_2 &\Rightarrow \frac{S.scale_3}{S.scale_1} \ll 1, \text{ et } \frac{S.scale_3}{S.scale_2} \ll 1 \\ &\Rightarrow 0.5 \frac{S.scale_3}{S.scale_1} + 0.5 \frac{S.scale_3}{S.scale_2} \ll 1 \\ &\Rightarrow flat(S) \ll 1 \end{aligned}$$

La valeur de cette fonction est donc nulle dans le cas d'un plan parfait.

- Cône : pour cette primitive, nous n'avons pas trouvé de fonction basée sur les valeurs propres pour estimer le coût. En effet, on ne peut pas caractériser un cône uniquement à partir de 3 axes principaux.

En utilisant l'erreur entre le segment et la primitive la mieux ajustée au segment

L'idée est de trouver la primitive qui approche le mieux le segment, puis de considérer l'erreur d'ajustement, c'est-à-dire la somme des distances entre les sommets du segment et la primitive trouvée, comme le coût de la segmentation. Nous nous sommes basés sur le travail d'Attene et al. [Attene06] pour à la fois trouver la primitive la mieux ajustée à un segment et puis calculer le coût d'ajustement. Voici le détail, par primitive.

- Dans le cas du plan, on calcule la matrice de covariance cov_v des sommets du segment S :

$$cov_v = \sum_i a(v_i)(v_i - \bar{v})(v_i - \bar{v})^T, \quad \bar{v} = \frac{\sum_i a(v_i)v_i}{\sum_i a(v_i)}$$

, où v_i est un sommet, et $a(v_i)$ est l'aire limitée de Voronoï (*Voronoi stricted area*) du sommet v_i ⁴. Dans le cas où tous les sommets ont la même importance, cette aire est mise à 1 pour tous les sommets.

Le meilleur plan approchant le segment passe par \bar{v} , et sa normale \vec{n} est le vecteur propre correspondant à la valeur propre minimale de cov_v .

L'erreur d'ajustement peut être alors calculée comme la somme pondérée des distances au carré entre les sommets du segment et le plan :

$$L^2 = \sum_i a(v_i)(\vec{n} \cdot (v_i - \bar{v}))^2$$

- Dans le cas de la sphère, notons P l'ensemble des points (x_i, y_i, z_i) du segment, et : $(x - c_x)^2 + (y - c_y)^2 + (z - c_z)^2 - R^2 = 0$, l'équation implicite de la sphère dont le centre est (c_x, c_y, c_z) et le rayon est R . La distance euclidienne au carré entre un point p_i et la sphère Sp est de la forme :

$$d^2(p_i, Sp) = \left(\sqrt{(x_i - c_x)^2 + (y_i - c_y)^2 + (z_i - c_z)^2} - R \right)^2$$

Trouver la sphère la mieux ajustée aux points P consiste à trouver (c_x, c_y, c_z) et R qui minimisent la somme des distances des points à la sphère :

$$\min \left(\sum_i d^2(p_i, Sp) \right)$$

Les auteurs de [Attene06] proposent de réécrire l'équation implicite de la sphère de la façon suivante :

$$x^2 + y^2 + z^2 + c_x^2 + c_y^2 + c_z^2 - 2c_x x - 2c_y y - 2c_z z - R^2 = 0$$

ou sous forme vectorielle :

$$\begin{bmatrix} 2x & 2y & 2z & 1 \end{bmatrix} \begin{bmatrix} c_x \\ c_y \\ c_z \\ R^2 - c_x^2 - c_y^2 - c_z^2 \end{bmatrix} = x^2 + y^2 + z^2$$

où les inconnues c_x, c_y, c_z , et R sont isolées dans un vecteur. En remplaçant x, y et z par les coordonnées des point de P nous obtenons le système d'équations linéaires sur-déterminé suivant :

$Aw = b$,

où :

$$A = \begin{bmatrix} 2x_1 & 2y_1 & 2z_1 & 1 \\ 2x_2 & 2y_2 & 2z_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 2x_n & 2y_n & 2z_n & 1 \end{bmatrix}, \quad b = \begin{bmatrix} x_1^2 + y_1^2 + z_1^2 \\ x_2^2 + y_2^2 + z_2^2 \\ \vdots \\ x_n^2 + y_n^2 + z_n^2 \end{bmatrix}, \quad \text{et} \quad w = \begin{bmatrix} c_x \\ c_y \\ c_z \\ R^2 - c_x^2 - c_y^2 - c_z^2 \end{bmatrix}$$

4. cette aire est un tiers de la somme des aires des triangles incidents à v_i et appartenant au segment S

La solution d'un tel système est donnée par : $w = (A^T A)^{-1} b$. Selon [Attene06], cette solution est une bonne approximation de la meilleure sphère passant par un groupe de points, surtout quand le nombre de points est élevé. Étant donnée la meilleure sphère passant par les sommets du segment S , l'erreur d'approximation est ensuite calculée comme la somme pondérée des distances au carré entre les sommets et la sphère :

$$L^2 = \sum_i a(v_i) (\|v_i - c\|_2 - R)^2$$

- Dans le cas du cylindre, l'idée est de trouver d'abord l'axe du cylindre. Pour cela, notons que pour un ensemble de triangles d'un cylindre parfait, la variation de normales serait presque nulle dans une direction. Il nous faut donc la direction \vec{n} pour laquelle la variation des normales des triangles du segment S est minimale. Pour trouver cette direction, une matrice de covariance des normales est calculée ainsi :

$$Cov_c = \sum_i |e_i| \beta(e_i) \bar{e}_i \bar{e}_i^T$$

, où

e_i est une arête interne du segment,

$|e_i|$ est sa longueur,

\bar{e}_i est un vecteur unitaire parallèle à e_i ,

et βe_i est l'angle entre les normales des deux triangles partageant e_i (positif si convexe, négatif si concave).

La direction pour laquelle la variation est minimale est le vecteur propre correspondant à la valeur propre maximale de la matrice Cov_c . L'axe du cylindre étant trouvé, il reste à déterminer le rayon et le centre du cylindre. Ce problème se réduit au problème d'ajustement d'un cercle en projetant les sommets du segment sur le plan dont la normale est \vec{n} et qui passe par le centre de masse c_m du segment. Soit la base orthonormée $\langle \vec{n}, \vec{e}_x, \vec{e}_y \rangle$ où \vec{e}_x et \vec{e}_y peuvent être les deux autres vecteurs propres de la matrice Cov_c , ou n'importe quelle paire de vecteurs orthogonaux entre eux et avec \vec{n} . On transforme chaque sommet v_i en $\vartheta(v_i) = \langle (v_i - c_m) \vec{e}_x, (v_i - c_m) \vec{e}_y \rangle$. Le centre (c_x, c_y) et le rayon r du meilleur cercle passant par les points $\vartheta(v_i)$ peuvent être trouvés d'une façon similaire à celle utilisée pour trouver la meilleure sphère, en diminuant la dimension de un. Le centre du cylindre est trouvé en re-transformant le centre du cercle : $c = c_m + \vec{e}_x c_x + \vec{e}_y c_y$.

Maintenant, ayant trouvé la direction \vec{n} , le centre c , et le rayon r du meilleur cylindre, nous pouvons calculer l'erreur comme la somme pondérée des distances au carré entre les sommets et le cylindre :

$$L^2 = \sum_i a(v_i) (\|(v_i - c) \cdot \vec{n}\|_2 - r)^2$$

- Le cas du cône n'est pas encore traité. Néanmoins il reste le même type de problème que pour le cylindre et la sphère. Les paramètres du cône sont son axe, son origine $s(x, y, z)$, et l'angle α formé par l'axe et une génératrice. On trouve d'abord le cône C (plus exactement, les paramètres du cônes) qui minimise la somme de distances des sommets du segment au cône. Puis l'erreur d'ajustement est la somme des distances des sommets du segment au cône C . L'ajustement d'un cône à un nuage de points a été traité par Schnabel *et al.* [schnabel2007] en se basant sur le principe de RANSAC.

Résultats

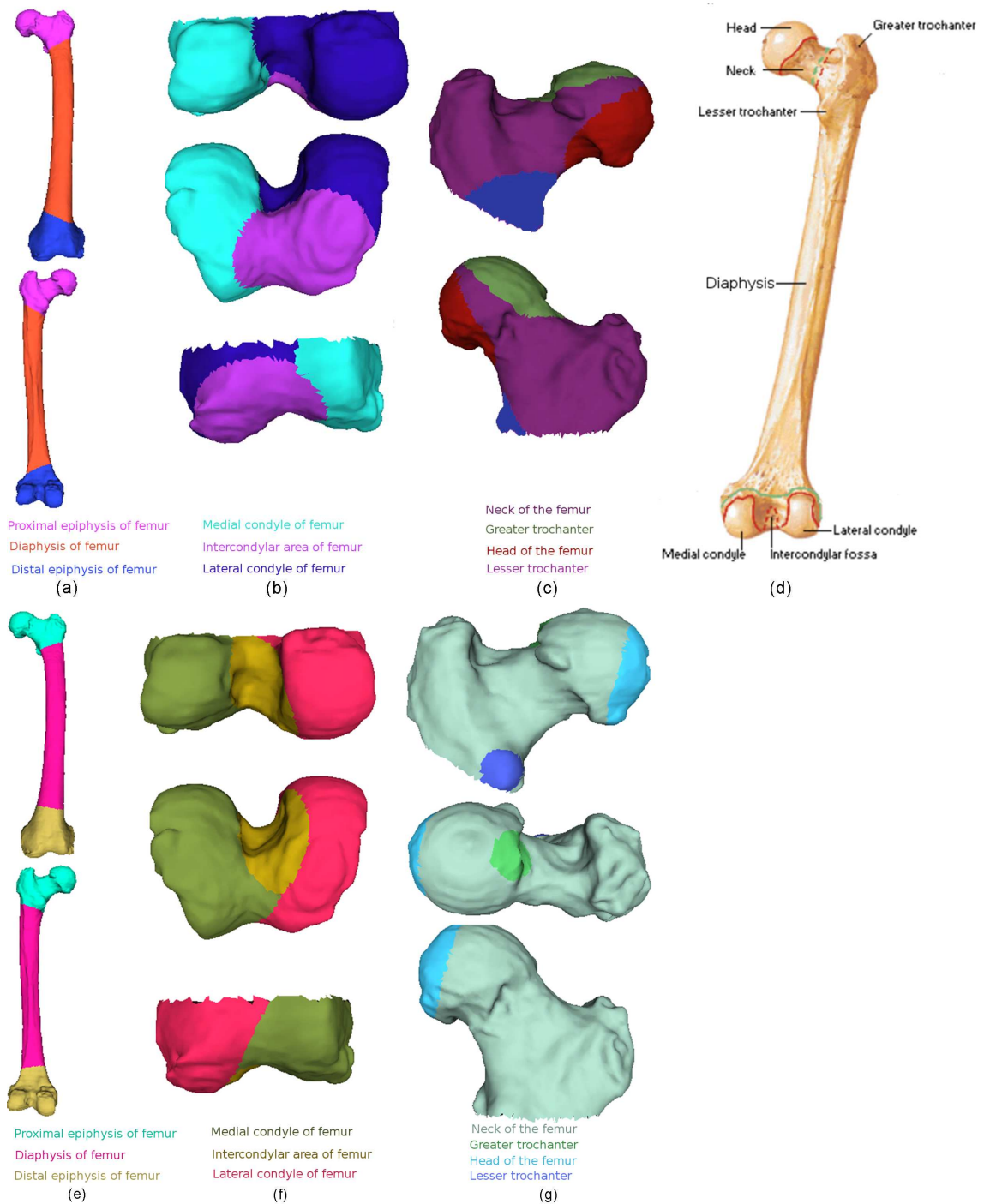


FIGURE 3.18: (a), (b) et (c) segmentations obtenues pour le fémur, l'épiphyse distale et l'épiphyse proximale respectivement en utilisant l'ACP. (d) segmentation de référence. (e), (f), (g) segmentations obtenues en utilisant l'erreur d'ajustement.

La Figure 3.18 nous montre les résultats obtenus pour le fémur, l'épiphyse distale et l'épiphyse proximale du fémur respectivement, en utilisant les deux méthodes de calcul du coût décrites ci-dessus et l'**Approche 1** pour l'optimisation. Nous pouvons constater que les résultats sont presque

identiques dans le cas du fémur. Cela peut être expliqué par le fait que le cylindre (selon l'ontologie) que l'on cherche à ajuster pour le diaphyse du fémur a une forme allongée et étroite, ce qui est bien capturé par la fonction d'objective *narrow()* et donc l'erreur approchée par cette fonction (dans le cas de l'ACP) est très petite, comme celle calculée par la seconde méthode.

On constate en revanche que les résultats sont loin d'être similaires dans le cas de l'épiphyse distale. Ici, l'aire intercondylaire du fémur (*Intercondylar area of femur*) est approchée (selon l'ontologie) par un cylindre, or ce cylindre n'est ni étroit ni allongé. Pour cette raison, l'utilisation de la fonction *narrow* pour calculer le coût fonctionne moins bien que l'utilisation de la recherche du meilleur cylindre.

En conclusion, la deuxième méthode est plus précise, mais aussi souvent légèrement plus coûteuse au niveau du temps de calcul. Le tableau 3.2 nous montre les temps de calcul pour les deux méthodes sur les trois entités sur un ordinateur muni d'un processeur Intel(R) Core 2CPU 2.00 GHz.

Entité	ACP		Erreur d'ajustement	
	Nombre d'itérations	temps (s.)	Nombre d'itérations	temps (s.)
Fémur	19	877.84	7	620.62
Épiphyse distale	27	266.09	16	369.70
Épiphyse proximale	24	267.38	12	377.84

TABLE 3.2: Tableau de comparaison entre les deux méthodes de calcul du coût.

Nous constatons que la méthode utilisant l'erreur d'ajustement s'achève en moins d'itérations que celle utilisant l'analyse par composante principale. En revanche, même avec un nombre d'itérations inférieur, le temps de calcul de cette méthode est supérieur.

Puisque les primitives que l'on cherche à ajuster sont déjà des approximations des formes des entités anatomiques, nous choisissons de calculer le coût en utilisant l'erreur d'ajustement car il est plus précis que celui basé sur l'ACP.

5.4 Hiérarchie

La dernière étape de notre méthode consiste à profiter directement d'une relation trouvée dans l'ontologie. La relation « *part of* » non seulement nous énumère les *entités filles* de l'*entité mère* correspondant au maillage de départ, mais aussi nous permet de continuer la segmentation, en descendant la hiérarchie anatomique représentée par cette relation. Une fois le segment correspondant à une *entité fille* déterminé, il sert de variable d'entrée (i.e. d'*entité mère*) pour une deuxième application de la méthode de segmentation. Ceci permet de le segmenter à son tour, et ainsi de suite jusqu'à arriver aux entités feuilles, i.e. sans descendant dans la relation. Nous pouvons ainsi calculer toutes les parties et sous-parties d'un organe, jusqu'à la plus petite partie décrite par l'ontologie, et obtenir une *segmentation hiérarchique*.

La Figure 3.19 nous montre comment la hiérarchie anatomique est suivie pour trouver toutes les parties et sous-parties du fémur.

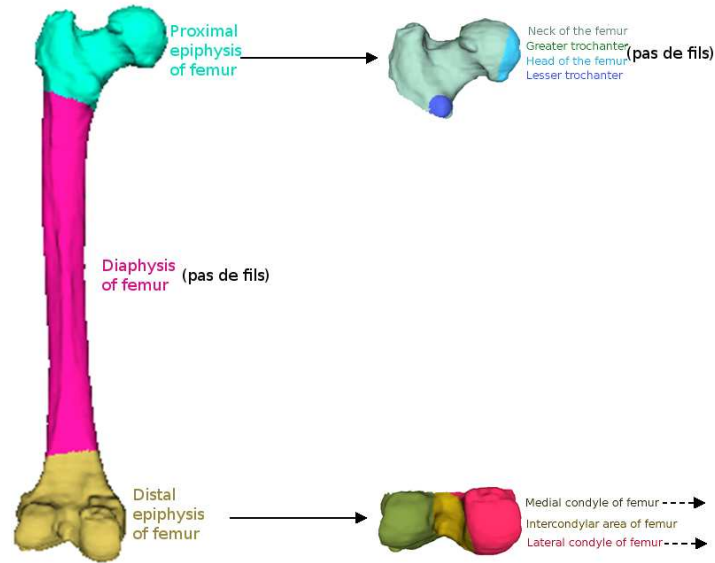
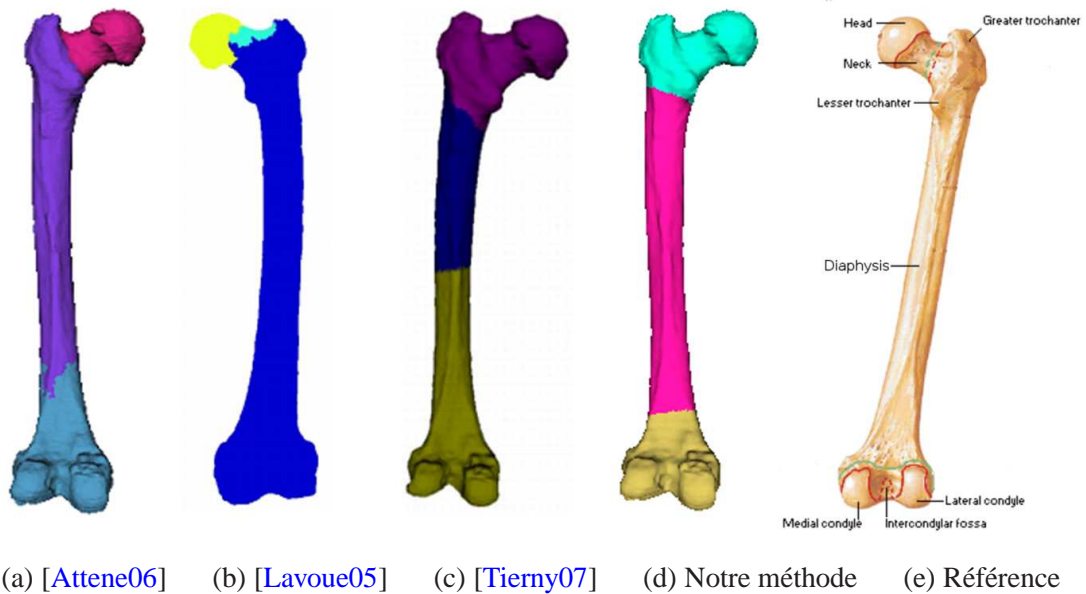


FIGURE 3.19: Suivi de la hiérarchie de l'ontologie.

6 Résultats

6.1 Comparaison avec d'autres méthodes de segmentation

D'abord, nous faisons une brève comparaison entre un résultat obtenu avec notre méthode et ceux obtenus en utilisant d'autres méthodes de la littérature. Pour cela, nous avons appliqué notre algorithme ainsi que trois autres : [Attene06], [Lavoue05], [Tierny07], pour segmenter le fémur. La figure 3.20 montre les résultats.



(a) [Attene06] (b) [Lavoue05] (c) [Tierny07] (d) Notre méthode (e) Référence
FIGURE 3.20: Comparaison entre plusieurs méthodes de segmentation sur le modèle du fémur.

Pour que la comparaison soit la plus juste possible, nous avons fourni aux autres méthodes les

paramètres que la nôtre trouve automatiquement dans l'ontologie. Pour l'algorithme d'ajustement des primitives [Attene06], nous avons fourni le nombre de segments (3) et les types de primitives à utiliser qui sont les formes géométriques trouvées dans l'ontologie (cylindre). Pour les deux autres méthodes [Lavoue05, Tierny07], le seul paramètre à fournir était le nombre de segments.

Bien que la comparaison entre les différentes méthodes de segmentation ne soit pas évidente, et qu'elle fait même l'objet de plusieurs travaux de recherches [Chen09, Benhabiles09], notre résultat semble correspondre le mieux aux entités anatomiques recherchées, voir le fémur de référence Figure 3.20-(e). En effet la limite entre le diaphyse et l'épiphyse proximale n'est pas bien placée dans le résultat de [Attene06]. La méthode de [Lavoue05] ne segmente pas entre le diaphyse et l'épiphyse distale, alors que la limite entre ces deux entités est mal placée avec la méthode de [Tierny07]. Cela est attendu, puisque notre méthode profite d'informations que les autres n'ont pas : la localisation, les relations d'adjacence et la taille relative.

6.2 Segmentation d'autres modèles

Nous avons également testé notre méthode de segmentation guidée par ontologie sur d'autres modèles, non issus du domaine de l'anatomie. Nous avons fourni manuellement les informations qui se trouvent normalement dans l'ontologie : le nombre de segments, leur étiquette, leur forme géométrique, leur taille relative, et les relations d'adjacence. Cependant, il n'était pas possible de fournir les informations de localisation manuellement. Ces informations se trouvent implicitement dans les noms des entités anatomiques (étiquettes). À cause de l'absence des informations de localisation, les points germes pour la croissance de régions ont été trouvés en utilisant le *K*-means, cf. Section 5.1.1. La Figure 3.21 nous montre les résultats obtenus ainsi que les informations fournies manuellement.

Le fait d'utiliser des primitives géométriques qui captent bien la forme des segments aide notre méthode à trouver une segmentation satisfaisante. De plus, l'utilisation des tailles relatives et des relations d'adjacence permet d'une part de commencer par une bonne segmentation initiale, et d'autre part d'accélérer la méthode. Le tableau Table 3.3 montre les temps de calcul pour tous les maillages. L'approche utilisée pendant l'optimisation est l'**Approche 1** et le coût d'ajustement est calculé en utilisant l'erreur d'ajustement d'une primitive.

Maillage	Nombre de sommets	Nombre de segments	Nombre d'itérations	Temps s.
Fémur	25002	3	7	620.62
Épiphyse distale	5304	3	16	369.70
Épiphyse proximale	7652	4	12	377.84
Table	13206	5	19	4466.03
Vase	13920	2	10	352.37
Vase	13920	3	10	825.00

TABLE 3.3: Tableau de temps de calcul.

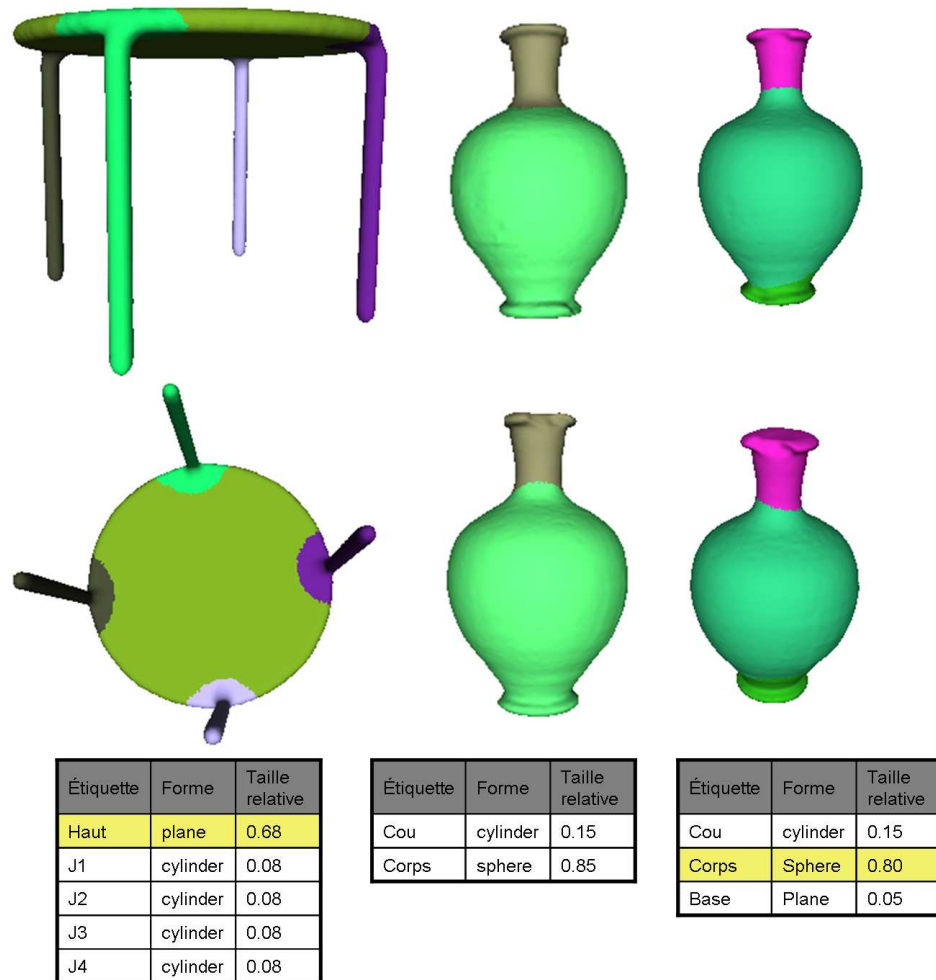


FIGURE 3.21: Résultats sur d'autres modèles. Les lignes en jaune des tableaux sont les segments qui sont voisins à tous les autres. Les autres ne sont mutuellement pas voisins.

Le temps de calcul dépend de plusieurs facteurs :

- le nombre de segments : nous constatons sur le tableau 3.3 que pour le même maillage (Vase), la segmentation en 3 segments prend plus du temps que celle en 2 segments. En effet, chaque itération de l'algorithme d'optimisation (*Generalized Pattern search*) consiste à tester les coûts de segmentation obtenus en testant les $2 \times m$, $m = 4 \times n$ voisins des n points germes correspondant aux n segments ;
- les formes géométriques : si elles sont présentes, l'algorithme prend plus de temps que dans le cas de leur absence. De plus, le temps pour trouver le coût d'ajustement d'un cylindre est plus élevé que celui pour ajuster un plan ou une sphère ;
- les informations sur la localisation et l'adjacence : la présence des informations sur la localisation et l'adjacence permet de rejeter des segmentations pendant l'itération avant de calculer leur coût, ce qui diminue le temps de calcul ;
- le nombre de sommets : plus il y a de points (sommets) à ajuster à une primitive, plus le temps de calcul augmente.

7 Discussion et perspectives

7.1 Résumé

Au sein de ce chapitre, une méthode de segmentation de maillages a été présentée. Cette méthode est basée sur l'utilisation de connaissances *a priori* organisées sous la forme d'une ontologie. Le but est d'identifier les parties (*entités filles*) d'un organe (*entité mère*) en segmentant le maillage le représentant en sous-maillages.

L'ontologie choisie pour guider la segmentation est « MyCorporisFabrica ». Bien que « MyCorporisFabrica » soit une ontologie récente, nous l'avons choisie car elle est facile à utiliser et à enrichir.

Avant l'utilisation de l'ontologie pour produire la segmentation, un enrichissement de l'ontologie était nécessaire (Section 4). Deux types d'informations ont été ajoutés à l'ontologie : la forme géométrique approchée d'un organe ou d'une de ses parties, et sa taille relative. Chaque entité anatomique (organe, partie d'organe) est approchée par une primitive géométrique simple : plan, sphère, cylindre et cône, ou un ensemble de ces primitives. Puis la taille d'une *entité fille* par rapport à son *entité mère* est inscrite dans l'ontologie. Ces deux informations sont ajoutées sous forme d'attributs.

Ces informations « ajoutées » et les informations déjà existantes dans l'ontologie (noms des entités, relations d'adjacence et informations de localisation) sont ensuite utilisées pour guider la segmentation. La technique utilisée pour la segmentation est la croissance des régions (Section 5.1.2). Si les informations de localisation sont disponibles, elles sont utilisées pour placer les points germes, sinon ceux-ci sont placés en appliquant l'algorithme des *K-means*. Chaque point est alors associé à une *entité fille* et il prend comme poids la taille relative de cette entité si elle existe, 1 sinon. Puis la croissance de région est lancée et une segmentation initiale est obtenue.

Ensuite, la segmentation est optimisée en cherchant à minimiser son coût (Section 5.3). Le coût d'une segmentation est la somme des coûts d'association d'*entité fille* à chaque segment. Ce coût est la somme du coût d'ajustement à la primitive géométrique décrivant cette entité d'une part, et la différence entre la taille relative de l'entité (inscrite dans l'ontologie) et celle du segment (calculée par rapport au maillage) d'autre part. Si l'une de ces informations (forme géométrique, taille relative) n'est pas présente dans l'ontologie, le coût associé est 0.

L'optimisation est faite en utilisant l'algorithme *Generalized Pattern Search*. À chaque itération de cet algorithme, une nouvelle segmentation est produite en déplaçant les points germes d'un pas Δ . Si cette segmentation satisfait les informations de localisation et d'adjacence, son coût est calculé. Si le coût est inférieur au coût inscrit, l'itération est réussie, la nouvelle position des points germes est prise, le nouveau coût est enregistré et le pas est diminué. Sinon, l'itération a échoué, et le pas est augmenté. L'algorithme s'achève quand le pas tombe en-dessous d'un certain seuil.

Finalement, toutes les opérations sont répétées pour chaque *entité fille* trouvée, pour identifier les sous-parties, en suivant la hiérarchie de l'ontologie (Section 5.4).

7.2 Perspectives

Les résultats obtenus sont très prometteurs, que ce soit dans le domaine médical ou au-delà (Section 6). Cependant, des améliorations sont envisageables selon les deux axes suivants :

1. Au niveau de l'ontologie : les informations déjà présentes dans l'ontologie et celles que nous avons ajoutées sont très utiles pour la segmentation. Cependant, davantage d'informations peuvent encore améliorer celle-ci. Par exemple, le fait d'approcher une entité par une primitive géométrique peut être plus ou moins précis selon l'entité et la forme géométrique. Pour concrétiser cette information, on pourrait associer à chaque forme géométrique un poids représentant la fiabilité d'approcher l'entité par cette forme. Ce poids pourrait être ensuite intégré dans l'optimisation en multipliant la fonction du coût d'ajustement de la primitive par ce poids.
2. Au niveau de la segmentation :
 - intégration d'autres relations : pendant la segmentation, nous profitons de deux relations inscrites dans l'ontologie : *fait partie de* et *adjacente à*. D'autres relations pourraient être prises en compte pendant la segmentation. Par exemple, la relation entre l'entité *Épicondyle latéral du fémur* et l'entité *Face latérale du condyle latéral du fémur* est plus complexe que la simple relation « adjacente à ». Ces deux entités filles de l'entité *Condyle latéral du fémur* sont l'une sur l'autre. Ce type de relation devrait être ajouté à l'ontologie et pris en compte lors de la segmentation ;
 - optimisation des frontières : les résultats obtenus sont très proches de nos attentes, mais les frontières requièrent une petite amélioration. Cette amélioration pourrait être faite en utilisant par exemple des informations de courbure ;
 - méthode de segmentation : nous nous sommes largement inspirés de la méthode de Simari *et al.* [Simari09] pour faire la segmentation. Or, une autre technique peut être envisageable. Le fait de commencer par une segmentation initiale qui satisfait les informations fournies par l'ontologie, peut faire penser à une optimisation qui consiste à bouger les frontières entre les segments sans utiliser les points germes. Nous pouvons, par exemple, commencer par le segment correspondant le moins aux critères (taille relative, coût d'ajustement, ...), et essayer d'améliorer ses frontières en ajoutant/enlevant des triangles qui font les frontières du segment. De plus, la croissance de région utilise la distance euclidienne au point germe (triangle) pour étiqueter les triangles (Section 5.1.2). Cependant, des critères dépendant de la forme géométrique potentiellement incluse dans l'étiquette d'un point germe peuvent être utilisés [Li09].

Finalement, plus de tests sur une variété d'organes sont nécessaires pour valider la méthode de segmentation. Une comparaison quantitative entre le résultat de notre méthode et celui fait manuellement par un anatomiste d'une part, et celui des autres méthodes d'ajustement de primitives d'autre part montrera l'intérêt de l'utilisation de l'ontologie pour guider la segmentation.

8 Conclusion

Dans ce chapitre une autre utilisation des connaissances *a priori* a été présentée. Contrairement au cas d'utilisation du chapitre précédent (cf. Chapitre 2), où les connaissances *a priori* ont été présentées sous forme de graphe, une représentation plus complexe a été nécessaire pour la segmentation du maillage : une ontologie.

À notre connaissance, l'idée d'utiliser une ontologie pour guider une segmentation est abordée ici pour la première fois. Or, cette idée s'avère très utile, car bien souvent la segmentation "idéale" d'un maillage dépend du domaine d'application. Il est presque impossible de trouver une méthode qui donne les résultats souhaités pour toutes les applications existantes. Il semble donc logique de décrire d'abord le résultat attendu sous forme d'ontologie, puis d'utiliser cette description pour guider la segmentation effective.

Le domaine médical, et plus spécifiquement l'anatomie, a été choisi pour cette application pour plusieurs raisons. D'abord, il s'agit d'un domaine où les connaissances *a priori* sont disponibles et formelles. Ensuite, le fait de trouver les sous-parties d'un organe automatiquement possède de nombreuses applications potentielles, que cela soit à des fins pédagogiques ou pour l'ajout de parties invisibles sur les images initiales.

Conclusion

L'imagination est plus important que le savoir.

Albert EINSTEIN

CE MÉMOIRE décrit des travaux réalisés pour deux applications médicales distinctes, mais avec le point commun de l'utilisation de connaissances anatomiques *a priori*. Le but de la première est la quantification et la localisation d'anévrismes, et la deuxième concerne la segmentation de maillages pour l'identification des parties d'un organe.

Puisque les connaissances *a priori* sont utilisées différemment dans ces deux applications, leur représentation est différente. En effet, la représentation de connaissances *a priori* est très liée à l'application envisagée.

Pour la première application, l'analyse d'anévrismes, un simple graphe représentant l'arbre vasculaire sain était suffisant pour la tâche de localisation d'anévrismes.

Pour la segmentation de maillages, une ontologie était nécessaire pour modéliser toutes les connaissances dont on a besoin pour guider l'algorithme de segmentation.

Dans ce qui suit, nous résumons tout d'abord les contributions de cette thèse, puis nous proposons des pistes de travaux futurs, avant de conclure.

Résumé des contributions

La localisation d'anévrismes

La localisation d'anévrismes est la dernière étape dans une chaîne de traitement de l'arbre vasculaire segmenté du patient et représenté sous forme de voxels. Les étapes de cette chaîne, entièrement automatiques, sont : l'extraction des lignes de centre des vaisseaux sanguins, la détection d'anévrismes, la quantification d'anévrismes et la localisation d'anévrismes. Si chacune de ces étapes prise indépendamment a déjà fait l'objet de travaux de recherche, le fait de les combiner ensemble dans le contexte de la détection, quantification et localisation d'anévrismes est, à notre connaissance, nouveau.

Dans l'étape d'extraction des lignes de centre des vaisseaux, nous proposons un algorithme basé

sur l'algorithme de Dijkstra pour extraire la ligne de centre principale. Au lieu d'utiliser la distance euclidienne, nous proposons une fonction de distance qui dépend de la profondeur des voxels (leur distance au bord de l'objet), et qui garantit ainsi une ligne de centre bien centrée. Ensuite, l'algorithme est répété pour extraire les branches successivement et dans l'ordre décroissant de leur longueur. À chaque itération, tous les voxels du centre (la ligne principale et les branches extraites) sont considérés comme source. Cette considération permet d'avoir des connexions nettes et intuitives entre les lignes de centre des vaisseaux. Le résultat de cette étape est le squelette de l'arbre vasculaire sous la forme d'un arbre (graphe avec une racine et sans cycle).

Pour l'étape de détection automatique d'anévrismes, nous proposons de différencier les vaisseaux des anévrismes en étudiant leurs diamètres. Un vaisseau a une forme tubulaire, aussi son diamètre est presque constant. En revanche la forme irrégulière d'un anévrisme fait que son diamètre est loin d'être constant. Nous proposons une fonction quadratique pour étudier l'évolution du diamètre d'une branche (qui peut être la ligne de centre d'un vaisseau ou d'un anévrisme), puis en observant les coefficients de cette fonction, nous pouvons distinguer entre les anévrismes et les vaisseaux.

Lors de l'étape de quantification, nous déterminons le collet de l'anévrisme et donnons automatiquement des mesures sur l'anévrisme : sa taille, sa longueur maximale selon l'axe horizontal, etc.

Finalement, pour la localisation d'anévrisme, nous commençons par extraire le graphe représentant l'arbre vasculaire du patient. Puis nous faisons une mise en correspondance entre l'arbre obtenu et des graphes de référence représentant un arbre vasculaire sain avec plusieurs niveaux de détails. Le résultat est le nom du vaisseau porteur de l'anévrisme.

La segmentation de maillages guidée par une ontologie

Le but de cette application est d'identifier sur un maillage représentant un organe donné les parties et sous-parties de cet organe. Les connaissances *a priori* pour cette application, étant assez complexes, sont représentées dans une ontologie. L'ontologie anatomique utilisée (MyCorporisFabrica) contient des informations très utiles pour la segmentation comme : le nombre de parties de l'organe, le nom de chaque partie, et les relations d'adjacence.

Cependant, et pour avoir un meilleur résultat, nous proposons d'ajouter plus d'informations à l'ontologie. La première connaissance *a priori* que l'on propose d'ajouter est la forme géométrique d'un organe. Nous avons trouvé, après discussion avec un anatomiste, qu'une entité anatomique (organe, ou partie d'un organe) peut être approchée par une ou plusieurs primitives géométriques. Pour cela, nous avons ajouté l'attribut "forme géométrique" à l'ontologie.

Un autre attribut que nous avons ajouté à l'ontologie est la taille relative. Cet attribut nous donne la taille relative d'une entité par rapport à l'organe dont elle fait partie.

Pour identifier les parties d'un organe sur un maillage le représentant, nous proposons un algorithme de segmentation de maillages qui profite de toutes les informations trouvées dans l'ontologie. L'algorithme proposé considère la segmentation comme un problème d'optimisation, comme la plupart des algorithmes de la littérature. La différence tient au fait qu'à chaque étape (segmentation initiale, et optimisation) et pour la fonction de critère que l'on cherche à optimiser, l'ontologie est utilisée pour choisir les paramètres quand cela est possible.

La fonction de critère (le montant à minimiser) dépend de l'étiquetage, c'est-à-dire l'entité associée à chaque segment. La fonction est la somme des coûts d'étiquetage de chaque segment. Le

coût d'association d'une entité à un segment est le coût d'ajustement de la primitive géométrique, précisée par l'attribut "forme géométrique" de l'entité, au segment. Si la forme n'est pas précisée, le coût est nul. S'ajoute à ce coût, la différence entre la taille relative de l'entité inscrite dans l'ontologie et le ratio entre l'aire du segment et celle du maillage complet.

La première étape de l'algorithme concerne la segmentation initiale. Une croissance de régions est faite pour obtenir cette segmentation. L'étiquetage est obtenu soit lors du placement des points germes (en profitant de l'information de localisation retirée des noms des entités), soit les points germes sont d'abord placés en utilisant l'algorithme de *K-means*, puis ils sont étiquetés d'une façon à minimiser la fonction de critère et en respectant les relations d'adjacence.

La deuxième étape consiste à utiliser l'algorithme de *Generalized Pattern Search* pour optimiser la segmentation obtenue initialement. Lors de cette étape, nous utilisons l'ontologie pour éliminer toute segmentation qui ne satisfait pas les relations d'adjacence fournies par l'ontologie.

Travaux futurs

Localisation d'anévrismes

La méthode que nous proposons a réussi à détecter, quantifier tous les anévrismes sacciformes présents dans un groupe test de vingt patients. Ce groupe ne suffit pas pour couvrir tous les cas cliniques possibles. De plus, la localisation est possible seulement quand une hémisphère est prise en compte. Si les deux hémisphères sont présentes, l'arbre vasculaire contiendra un cycle correspondant au polygone de Willis, alors que la présence de cycle n'est pas prise en compte par l'algorithme de squelettisation car il produit un squelette sous forme d'arbre sans cycle. Par conséquent, la localisation n'est pas possible car le graphe est extrait à partir du squelette.

Ainsi, une première extension de notre méthode serait de prendre en compte les cas où l'arbre vasculaire a des cycles. Pour cela, il faudrait développer un algorithme de squelettisation qui produit un squelette homotope à l'objet.

Ensuite, une validation clinique est nécessaire pour couvrir plus de cas. D'abord, il faut prendre un groupe avec plus de patients ayant des anévrismes dans des localisations variées. Puis, pour mesurer l'apport de notre approche, il faut qu'un médecin note la décision qu'il aurait prise sans les quantifications fournies par notre méthode. De plus, des mesures de validation de la littérature peuvent être utilisées [[Firouzan10](#)].

Segmentation de maillages

Puisque notre méthode de segmentation dispose de plus d'informations, il est normal que le résultat de la segmentation soit plus proche des attentes des anatomistes qu'avec les méthodes n'utilisant pas ces informations. Cependant, il reste des informations (relations, attributs) fournies par l'ontologie et utiles pour la segmentation qui ne sont pas encore intégrées dans l'algorithme. L'intégration de ces informations serait donc une première piste pour améliorer notre méthode.

De plus, grâce à l'extensibilité de l'ontologie MyCorporisFabrica, davantage de connaissances utiles pour la segmentation peuvent être ajoutées à l'ontologie. Nous pouvons, par exemple, ajouter

un attribut aux entités ayant une forme géométrique pour exprimer le degré de fiabilité du choix de la primitive. Cet attribut peut être ensuite intégré dans la fonction de critère en le multipliant par le coût d'ajustement de la primitive associée.

Une nouvelle idée pour l'optimisation pourrait aussi être testée. Au lieu d'utiliser l'algorithme de *Generalized Pattern Search*, nous pouvons juste déplacer les frontières entre les segments afin de minimiser le coût de la segmentation. Les triangles des frontières peuvent d'abord être classés selon leur distance au point germe, et nous pouvons commencer par déplacer les plus lointains, et ainsi de suite.

Pour conclure

L'idée d'utiliser les connaissances anatomiques *a priori* pour réaliser des applications médicales est une idée récente. Les résultats des applications présentées sont très prometteurs, et encourageant à continuer dans la direction d'investir des connaissances *a priori* pour comprendre, étiqueter et compléter des modèles géométriques 3D.

Ces applications sont deux exemples sur l'apport de connaissances aux algorithmes graphiques dans le domaine médical. L'idée est cependant généralisable aux autres domaines de recherche comme la reconnaissance de forme, l'analyse de forme et la correspondance entre objets.

Annexe

Techniques d'imagerie médicale

1 Imagerie par Résonance Magnétique (IRM)

L'imagerie par résonance magnétique est une technique d'imagerie non invasive. Elle est basée sur la résonance magnétique nucléaire. Les noyaux atomiques tournent sur eux-mêmes produisant une petite aimantation appelée "*spin*"¹. En l'absence d'un champ magnétique, un spin a une orientation

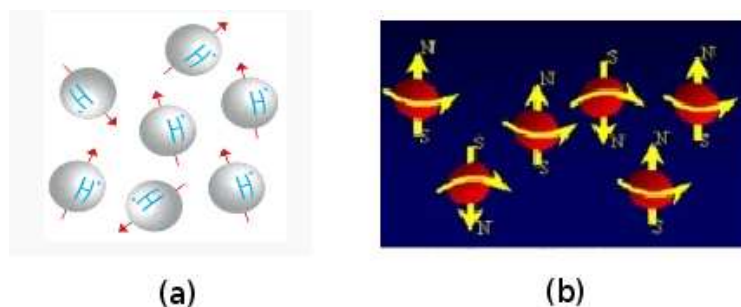


FIGURE A.1: Les protons : (a) en absence d'un champ magnétique, (b) en présence d'un champ magnétique (http://www.dossier-irm.fr/le_principe_de_la_resonance_magnetique_nucleaire.html).

aléatoire (Figure A.1- (a)), et la somme des orientations est donc nulle. En présence d'un champ magnétique, les spins s'orientent, à cause de la polarisation, dans la direction du champ externe mais pas forcément tous dans le même sens (Figure A.1). Certains auront le même sens que le champ (ils sont dits parallèles) et d'autres le sens contraire (antiparallèles), ceux qui ont le même sens que le champ sont plus nombreux. La différence entre le nombre de spins parallèles et le nombre de spins antiparallèles est proportionnel à la puissance du champ magnétique externe. L'alignement des spins n'est pas parfait ; lorsqu'ils tournent sur eux-mêmes ils font un mouvement de précession². La vitesse

1. Plus de détails et d'images d'illustration sur le site : http://www.dossier-irm.fr/le_principe_de_la_resonance_magnetique_nucleaire.html

2. La précession est le nom donné au changement graduel d'orientation de l'axe de rotation d'un objet. (<http://fr.wikipedia.org/wiki/Pr%C3%A9cession>)

de cette précession (la fréquence de Larmor) est proportionnelle à l'intensité du champ.

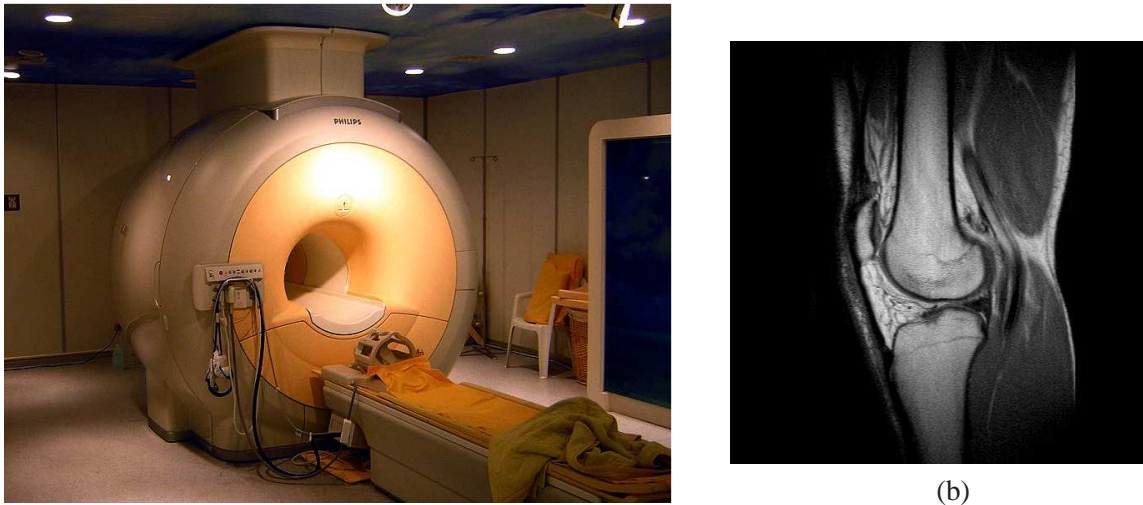


FIGURE A.2: (a) La machine IRM (http://fr.wikipedia.org/wiki/Imagerie_par_r%C3%A9sonance_magn%C3%A9tique), (b) IRM du genou en coupe sagittale (<http://www.ac-grenoble.fr/disciplines/sti/biotec/articles.php?lng=fr&pg=131#IRM>).

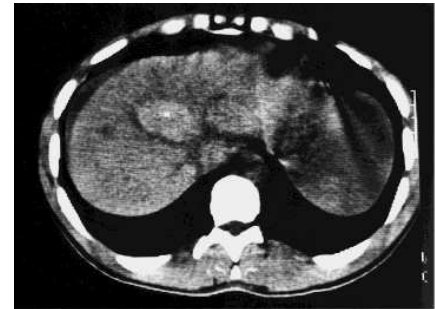
En IRM, on apporte de l'énergie sous forme d'une onde de radiofréquence. Lorsque la fréquence de Larmor est exactement équivalente à celle de l'onde, le système entre en résonance. L'effet de cette résonance est un débattement plus ample du mouvement de précession (angle de bascule). Pendant cette phase d'excitation, l'aimantation tissulaire (somme des aimantations des spins) bascule. Cette bascule se traduit par une diminution de l'aimantation longitudinale (selon l'axe du champ) et l'apparition d'une aimantation transversale. Cette aimantation apparaît grâce au phasage des spins. Après l'arrêt de l'onde de radiofréquence, le système passe à une phase de relaxation en restituant l'énergie sous la forme d'un signal de radiofréquence recueilli par une antenne dans le plan xy . L'aimantation longitudinale repousse suivant une loi exponentielle grâce au retour à l'état initial des spins. Cette loi est caractérisée par le temps $T1$ nécessaire pour un tissu pour regagner 63% de son aimantation maximale. L'aimantation transversale elle, tombe à cause du déphasage des spins en suivant une courbe décroissante. Le temps $T2$, caractérisant cette courbe, est le temps nécessaire pour que le tissu perde 63% de son aimantation transversale maximale. Le contraste en IRM est dû à la différence entre ces deux temps.

Les parties principales d'une machine d'IRM sont : (1) l'aimant qui produit le champ magnétique, la puissance du champ est mesuré en Tesla (T), (2) les bobines de gradient du champ qui permettent de varier le champ selon les axes x , y , et z contrôlant ainsi l'épaisseur des coupes (transversales, frontales, et sagittales), (3) les antennes qui captent le signal de radiofréquence. La Figure A.2 montre une machine d'IRM et une image 2D d'une IRM du genou.

2 La tomodensitométrie (CT-Scan)

Cette technique d'imagerie est, contrairement à l'IRM irradiante, basée sur l'absorption des rayons X par les tissus. En utilisant une extension du théorème de Radon, un objet peut être reconstitué à

partir d'une série de mesures prises par coupe (tomographie). Pour acquérir les données nécessaires à cette reconstitution, on soumet le patient au balayage d'un faisceau de rayons X. Puis un ordinateur associe à chaque pixel une valeur de gris proportionnelle au montant des rayons absorbés par le tissu correspondant³.



(b)

(a)
FIGURE A.3: (a) Le CT-scan à double source (<http://fr.wikipedia.org/wiki/Tomodensitom%C3%A9trie>), (b) Examen tomodensitométrique du foie (http://www.sfar.org/acta/dossier/archives/ca96/html/ca96_029/96_29.htm).

La Figure A.3-(a) montre une machine (scanner) qui utilise cette technique, un examen concernant le foie est illustré sur la même figure (b).

3 L'angiographie

Cette technique est utilisée pour l'imagerie des vaisseaux sanguins. Traditionnellement, elle se base sur l'injection d'un produit de contraste radio-opaque avant de pratiquer une imagerie basée sur les rayons X. Le produit est injecté en insérant un cathéter pour accéder aux vaisseaux (artères ou veines) à partir de l'artère ou de la veine fémorale.

Les images sont souvent prises selon la technique d'angiographie numérisée avec soustraction (*Digital subtraction angiography- DSA*). Dans cette technique, plusieurs images sont prises toutes les secondes (2-3 images par seconde), puis les os et les autres organes sont soustraits des images et seulement les vaisseaux remplis par l'agent de contraste restent visibles.

Cependant le mot angiographie apparaît avec d'autres techniques d'imagerie comme l'IRM, produisant l'angiographie par résonance magnétique (ARM). Dans ce cas, les vaisseaux sont rendus plus visibles en utilisant plusieurs techniques, parmi celles-ci citons⁴ :

- l'angiographie par contraste (*Contrast-Enhanced MRA*) : un produit de contraste est injecté dans une veine, sans cathéter. Ce produit rend les vaisseaux plus visibles (Figure A.4-(a)) ;

3. <http://fr.wikipedia.org/wiki/Tomodensitom%C3%A9trie>

4. http://en.wikipedia.org/wiki/Magnetic_resonance_angiography

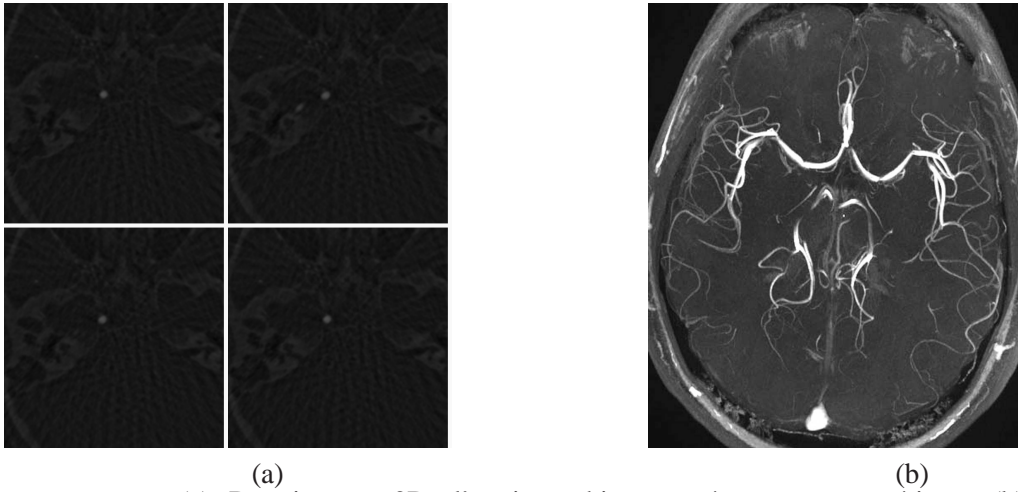


FIGURE A.4: (a) Des images 2D d'angiographie par résonance magnétique, (b) Une ARM de temps de vol du cercle de Willis du cerveau (http://en.wikipedia.org/wiki/Magnetic_resonance_angiography).

- l'angiographie de temps de vol (*Time of Flight (TOF) angiography*) : celle-ci est basée sur le temps d'écho (temps entre le signal d'excitation et la réception de l'écho). L'utilisation d'un temps d'écho court rend le sang circulant plus brillant que les tissus statiques (Figure A.4-(b)).

*
* *

Nous n'avons cité que les techniques d'imagerie médicale mentionnées dans ce mémoire. Beaucoup d'autres techniques existent. Le choix entre ces techniques se fait selon la structure anatomique à visualiser, la pathologie recherchée, le coût de l'examen et le fait que celui-ci soit invasif ou pas.

INDEX

A

a priori, [7](#)

Algorithmes

algorithme d'étiquetage évolutif, [76](#)

algorithme Hongrois, [75](#)

CEASAR, [44](#)

construction de graphe, [60](#)

Détction d'une courbe fermée, [57](#)

Dijkstra, [44](#)

Extraction des branches, [50](#)

L'algorithme de *Generalized Pattern Search*,
[77](#)

TEASAR, [44](#)

anévrisme, [37](#)

collet, [56](#)

disséquant, [37](#)

fusiforme, [37](#)

sacciforme, [37](#)

Atlas, [8](#)

Atlas Classiques, [8](#)

Atlas informatisés, [9](#)

Atlas probabilistes, [10](#)

C

CARO, [23](#)

Clipping, [38](#)

D

Définitions, [42](#)

18-voisins, [42](#)

26-voisins, [42](#)

6-voisins, [42](#)

bifurcation, [60](#)

branche, [43](#)

extrémité, [60](#)

i-connexe, [43](#)

ligne principale, [43](#)

lignes de centre, [43](#)

maillage, [69](#)

mesure de vaisseau, [40](#)

mince, [43](#)

segment, [60](#)

longueur, [60](#)

rayon, [60](#)

segmentation du maillage, [69](#)

Méthodes, [71](#)

Types, [70](#)

squelette, [43](#)

voxel, [42](#)

E

Embolisation, [37](#)

F

FMA, [21](#)

G

GALEN, [20](#)

I

Imagerie

Angiographie, [109](#)

Angiographie de temps de vol, [110](#)

Angiographie numérisée avec soustrac-
tion, [109](#)

Imagerie par Resonance Magnétique, [107](#)

Tomodensitométrie, [108](#)

M

MyCorporisFabrica, [24](#)

O

Ontologie, [13](#)

R

Relations

d'emplacement, [16](#)

méréologiques, [15](#)

topologiques, [16](#)

V

Visible Human, [10](#)

BIBLIOGRAPHIE

- [Attene06] M. Attene, B. Falcidieno, and M. Spagnuolo. Hierarchical mesh segmentation based on fitting primitives. *The Visual Computer*, 22(3) :181–193, 2006.
- [Attene06a] M. Attene, S. Katz, M. Mortara, G. Patane, M. Spagnuolo, and A. Tal. Mesh segmentation - a comparative study. In *SMI '06 : Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006*, page 7, Washington, DC, USA, 2006. IEEE Computer Society.
- [Attene09] M. Attene, F. Robbiano, M. Spagnuolo, and B. Falcidieno. Characterization of 3D shape parts for semantic annotation. *Computer Aided Design*, 41(10) :756–763, 2009.
- [Audet02] C. Audet and J. E. Dennis, Jr. Analysis of generalized pattern searches. *SIAM J. on Optimization*, 13(3) :889–903, 2002.
- [Aylward02] S. R. Aylward and E. Bullitt. Initialization, noise, singularities, and scale in height ridge traversal for tubular object centerline extraction. *IEEE Transactions on Medical Imaging*, 21(2) :61–75, 2002.
- [Aylward96] S. Aylward, S. Pizer, D. Eberly, and E. Bullitt. Intensity ridge and widths for tubular object segmentation and description. *IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, 0 :01–31, 1996.
- [Beck03] R. Beck and S. Schulz. Logic-based remodeling of the digital anatomist foundational model. *AMIA Annual Symposium Proceedings*, pages 71–75, 2003.
- [Benhabiles09] H. Benhabiles, J.-P. Vandeborre, G. Lavoué, and M. Daoudi. A framework for the objective evaluation of segmentation algorithms using a ground-truth of human segmented 3D-models. In *IEEE International Conference on Shape Modeling and Applications (Shape Modeling International 2009)*, Beijing, China, June 26-28 2009. short paper.
- [Bertrand06] G. Bertrand and M. Couprie. A new 3D parallel thinning scheme based on critical kernels. In *Discrete Geometry for Computer Imagery*, pages 580–591. Springer, 2006.
- [Bertrand07] G. Bertrand. On critical kernels. *Comptes Rendus de l'Académie des Sciences, Série Math.*, I(345) :363–367, 2007.
- [Bertrand08] G. Bertrand and M. Couprie. Two-dimensional parallel thinning algorithms based on critical kernels. *Journal of Mathematical Imaging and Vision*, 31(1) :35–56, 2008.
- [Bertrand09] G. Bertrand and M. Couprie. On parallel thinning algorithms : Minimal non-simple sets, p-simple points and critical kernels. *Journal of Mathematical Imaging and Vision*, 35 :23–35, 2009.

- [Bitter00] I. Bitter, M. Sato, M. Bender, K. T. McDonnell, A. Kaufman, and M. Wan. CEASAR : a smooth, accurate and robust centerline extraction algorithm. In *VIS '00 : Proceedings of the conference on Visualization '00*, pages 45–52, Los Alamitos, CA, USA, 2000. IEEE Computer Society Press.
- [Bitter01] I. Bitter, A. E. Kaufman, and M. Sato. Penalized-distance volumetric skeleton algorithm. *IEEE Transactions on Visualization and Computer Graphics*, 7(3) :195–206, 2001.
- [Bittner08] T. Bittner, M. Donnelly, L. J. Goldberg, and F. Neuhaus. Modeling principles and methodologies : Spatial representation and reasoning. In Albert Burger, Duncan Davidson, and Richard Baldock, editors, *Anatomy Ontologies for Bioinformatics.*, Computational Biology, chapter 15, pages 307–326. Springer London, 2007.
- [Boltcheva07] D. Boltcheva. *Modélisation géométrique et topologique des images discrètes*. PhD thesis, Université Strasbourg1 - Louis Pasteur, Octobre 2007.
- [Bouix05] S. Bouix, K. Siddiqi, and A. Tannenbaum. Flux driven automatic centerline extraction. *Medical Image Analysis*, 9(3) :209–221, 2005.
- [Bray98] T. Bray, J. Paoli, and C. Sperberg. Exetnsible markup language (XML) 1.0 reference. W3C recommendation., February 1998. <http://www.w3.org/TR/REC-xml/>.
- [Brickly99] D. Brickly and R. V. Guha. Resource Description Framework (RDF) schema specification. W3C proposed recommendation., March 1999. <http://www.w3.org/TR/PR-rdf-schema>.
- [Brinkly99] James F. Brinkley, Benjamin A. Wong, Kevin P. Hinshaw, and Cornelius Rosse. Design of an anatomy information system. *IEEE Computer Graphics and Applications*, 19(3) :38–48, 1999.
- [Catalano09] C. Catalano, E. Camossi, R. Ferrandes, V. Cheutet, and N. Sevilmis. A product design ontology for enhancing shape processing in design workflows. *Journal of Intelligent Manufacturing*, 20 :553–567, 2009.
- [Cerveria00] P. Cerveria, M. Masserolia, F. Pincirolia, L. Portonib, and L. Vizzottoc. Anatomical knowledge representation : attempting querying integration on VHD via UMLS. In *The Third Visible Human Project Conference Proceedings*, 2000.
- [Charnoz04] A. Charnoz, V. Agnus, and L. Soler. Portal vein registration for the follow-up of hepatic tumours. In *MICCAI '04 : Proceedings of the 7th International Conference on Medical Image Computing and Computer-Assisted Intervention*, volume 3217 of *LNCS*, pages 878–886. Springer, september 2004.
- [Charnoz05] A. Charnoz, V. Agnus, G. Malandain, L. Soler, and M. Tajine. Tree matching applied to vascular system. In *GbrPR 2005*, volume 3434 of *LNCS*, pages 183–192, 2005.
- [Chen09] X. Chen, A. Golovinskiy, and T. Funkhouser. A benchmark for 3D mesh segmentation. *ACM Transactions on Graphics (Proc. SIGGRAPH)(2009)*, 28(3), August 2009.
- [Chillet03] D. Chillet, J. Jomier, D. Cool, and S. Aylward. Vascular atlas formation using a vessel-to-image affine registration method. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003*, volume 2878 of *Lecture Notes in Computer Science*, pages 335–342. Springer Berlin / Heidelberg, 2003.
- [Cohen-Steiner04] D. Cohen-Steiner, P. Alliez, and M. Desbrun. Variational shape approximation. In *SIGGRAPH '04 : ACM SIGGRAPH 2004 Papers*, pages 905–914, New York, NY, USA, 2004. ACM.

- [Corcho00] O. Corcho and A. Gómez-Pérez. A roadmap to ontology specification languages. In *Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling and Management, EKAW '00*, pages 80–96, London, UK, 2000. Springer-Verlag.
- [Corcho03] O. Corcho, M. Fernández-López, and A. Gómez-Pérez. Methodologies, tools and languages for building ontologies : where is their meeting point ? *Data & Knowledge Engineering*, 46 :41–64, July 2003.
- [Cordella98] L. P. Cordella, P. Foggia, C. Sansone, and M. Vento. Performance evaluation of the VF graph matching algorithm. In *Computing*, volume 12, pages 43–52, 1998.
- [Cordella99] L. P. Cordella, P. Foggia, C. Sansone, and M. Vento. Performance evaluation of the VF graph matching algorithm. In *ICIAP '99 : Proceedings of the 10th International Conference on Image Analysis and Processing*, page 1172, Washington, DC, USA, 1999. IEEE Computer Society.
- [Cornea07] N.D. Cornea, D. Silver, and P. Min. Curve-skeleton properties, applications, and algorithms. *Transactions on Visualization and Computer Graphics*, 13(3) :530–548, May-June 2007.
- [Cox94] T. Cox and M. Cox. *Multidimensional Scaling*. Chapman & Hall, London, 1994.
- [Dameron03] O. Dameron. *Modélisation, représentation et partage de connaissances anatomiques sur le cortex cérébral*. PhD thesis, Université de Rennes 1, Décembre 2003.
- [Dean02] M. Dean, D. Connolly, F. van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider, and L. A. Stein. OWL Web Ontology Language 1.0 reference. Technical report, July 2002.
- [Deschamps01] T. Deschamps and L.D. Cohen. Fast extraction of minimal paths in 3D images and applications to virtual endoscopy. *Medical Image Analysis*, 5(4), 2001.
- [Descoteaux08] M. Descoteaux, D. L. Collins, and K. Siddiqi. A geometric flow for segmenting vasculature in proton-density weighted MRI. *Medical Image Analysis*, 12(4) :497–513, 2008.
- [Dijkstra59] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1 :269–271, 1959.
- [Donnelly06] M. Donnelly, T. Bittner, and C. Rosse. A formal theory for spatial representation and reasoning in biomedical ontologies. *Artificial Intelligence in Medecine*, 36(1) :1–27, January 2006.
- [Ecker04] R.D. Ecker and L.N. Hopkins. Natural history of unruptured intracranial aneurysms. *Neurosurg Focus*, 17(5), 2004.
- [Farquhar96] A. Farquhar, R. Fikes, and J. Rice. The ontolingua server : a tool for collaborative ontology construction. In *International Journal of Human-Computer Studies*, 1996.
- [Firouzian10] A. Firouzian, R. Manniesing, Z. H. Flach, R. Risselada, F. Kooten, M. C.J.M. Sturkenboom, A. Lugt, and W.J. Niessen. Intracranial aneurysm segmentation in 3D CT angiography : Method and quantitative validation with and without prior noise filtering. *European Journal of Radiology*, In Press, Corrected Proof :to appear, 2010.
- [Fouard04] C. Fouard, E. Cassot, G. Malandain, C. Mazel, S. Prohaska, D. Asselot, M. Westerhoff, and J.P. Marc-Vergnes. Skeletonization by blocks for large 3D datasets : application to brain microcirculation. *IEEE International Symposium on Biomedical Imaging : Nano to Macro, 2004.*, 1 :89–92, 2004.

- [Frangi98] A. F. Frangi, W. J. Niessen, K. L. Vincken, and M. A. Viergever. Multiscale vessel enhancement filtering. 1998.
- [Garcia-Rojas05] A. Garcia-Rojas, D. Thalmann, F. Vexo, L. Moccozet, N. Magnenat-Thalmann, M. Mortara, M. Spagnuolo, and M. Gutierrez. An ontology of virtual humans : incorporating semantics into human shapes. *Integration of Knowledge, Semantics and Digital Media Technology, 2005. EWIMT 2005. The 2nd European Workshop on the (Ref. No. 2005/11099)*, pages 7–14, 2005.
- [Gibaud03] B. Gibaud, O. Dameron, and X. Morandi. Re-use of a multi-purpose knowledge corpus on cortex anatomy for educational purposes. In *MIE'03 Proceedings*, pages 439–444, 2003.
- [Golbreich06] C. Golbreich, S. Zhang, and O. Bodenreider. The foundational model of anatomy in OWL : Experience and perspectives. *Web Semantics : Science, Services and Agents on the World Wide Web*, 4(3) :181 – 195, 2006. Semantic Web for Life Sciences.
- [Grenon04] P. Grenon, B. Smith, and L. Goldberg. Biodynamic ontology : Applying BFO in the biomedical domain. In *Studies in Health Technology and Informatics*, pages 20–38. IOS Press, 2004.
- [Gruber93] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5 :199–220, June 1993.
- [Guarino95] Nicola Guarino. Formal ontology, conceptual analysis and knowledge representation. *International Journal of Humman-Computer Studies*, 43 :625–640, December 1995.
- [Haendel07] M. A. Haendel, F. Neuhas, D. Osumi-Sutherland, P. M. Mabee, J. L. V. Mejino Jr., C. J. Mungall, and B. Smith. CARO- the common anatomy reference ontology. In *Anatomy Ontologies for Bioinformatics*, chapter 16, pages 327–349. Springer, 2007.
- [Hohne01] K. H. Höhne, B. Pflesser, A. Pommert, M. Riemer, R. Schubert, T. Schiemann, U. Tiede, and U. Schumacher. A realistic model of human structure from the visible human data. *Methods Inf Med*, 40(2) :83–9, 2001.
- [Hohne96] K. h. Höhne, B. Pflesser, A. Pommert, M. Riemer, T. Schiemann, R. Schubert, and U. Tiede. A 'virtual body' model for surgical education and rehearsal. *Computer*, 29 :25–31, January 1996.
- [Irniger03] C. Irniger and H. Bunke. Theoretical analysis and experimental comparison of graph matching algorithms for database filtering. In *GbrPR'03 : Proceedings of the 4th IAPR international conference on Graph based representations in pattern recognition*, pages 118–129, Berlin, Heidelberg, 2003. Springer-Verlag.
- [Jiang05] G. Jiang and L. Gu. An automatic and fast centerline extraction algorithm for virtual colonoscopy. *Proceeding of the 2005 IEEE. Engineering in Medicine and Biology 27th Annual Conference*, pages 5194–5152, 2005.
- [Joubert98] M. Joubert, M. Fieschi, J.-J. Robert, F. Volot, and D. Fieschi. UMLS-based Conceptual Queries to Biomedical Information Databases. *Journal of the American Medical Informatics Association*, 5 :52–61, 1998.
- [Katz03] S. Katz and A. Tal. Hierarchical mesh decomposition using fuzzy clustering and cuts. In *SIGGRAPH '03 : ACM SIGGRAPH 2003 Papers*, pages 954–961, New York, NY, USA, 2003. ACM.
- [Katz05] S. Katz, G. Leifman, and A. Tal. Mesh segmentation using feature point and core extraction. *The Visual Computer*, 21(8-10) :649–658, 2005.

- [Kifer95] M. Kifer, G. Lausen, and J. Wu. Logical foundations of object-oriented and frame-based languages. *J. ACM*, 42 :741–843, July 1995.
- [Kuhn55] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2) :83–97, 1955.
- [Lamecker08] H. Lamecker. *Variational and statistical shape modeling for 3D geometry reconstruction*. PhD thesis, Freie Universität Berlin - Zuse Institute Berlin., Avril 2008.
- [Lassila99] O. Lassila and R. Swick. Resource Description Framework (RDF) Model and Syntax specification. W3C recommendation., January 1999. <http://www.w3.org/TR/PR-rdf-syntax/>.
- [Lavoue05] G. Lavoué, F. Dupont, and A. Baskurt. A new CAD mesh segmentation method, based on curvature tensor analysis. *Computer-Aided Design*, 37(10) :975–987, September 2005.
- [Lazarus99] F. Lazarus and A. Verroust. Level set diagrams of polyhedral objects. In *Proceedings of the fifth ACM symposium on Solid modeling and applications*, SMA '99, pages 130–140, New York, NY, USA, 1999. ACM.
- [Li09] B. Li, R. Schnabel, J. Shiyao, and R. Klein. Variational surface approximation and model selection. *Computer Graphics Forum (Proc. of Pacific Graphics)*, 28(7), October 2009.
- [Lindberg93] D.A. Lindberg, B.L. Humphreys, and A.T. McCray. The unified medical language system. In *Methods of information in medecine*, volume 32, pages 281–291, 1993.
- [Liu04] R. Liu and H. Zhang. Segmentation of 3D meshes through spectral clustering. *Pacific Conference on Computer Graphics and Applications*, 0 :298–305, 2004.
- [Lohou02] C. Lohou and G. Bertrand. A new 3D 6-subiteration thinning algorithm based on p-simple points. In *DGCI '02 : Proceedings of the 10th International Conference on Discrete Geometry for Computer Imagery*, pages 102–113, 2002.
- [Lohou04] C. Lohou and G. Bertrand. A 3D 12-subiteration thinning algorithm based on p-simple points. *Discrete Appl. Math.*, 139(1-3) :171–195, 2004.
- [Lorensen87] W. E. Lorensen and H. E. Cline. Marching cubes : A high resolution 3D surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21 :163–169, August 1987.
- [Luke00] S. Luke and J. Heflin. SHOE 1.01. proposed specification. SHOE project. <http://www.cs.umd.edu/projects/plus/SHOE/spec.html>, February 2000.
- [MacGregor91] R. M. MacGregor. Inside the loom description classifier. *SIGART Bulletin*, 2 :88–92, June 1991.
- [Mechouche07] A. Mechouche, C. Golbreich, and B. Gibaud. Vers un système hybride pour l'annotation sémantique d'images IRM du cerveau. In Monique N. Fraiture and Gilles Venturini, editors, *EGC*, volume RNTI-E-9 of *Revue des Nouvelles Technologies de l'Information*, pages 433–444. Cépaduès-Éditions, 2007.
- [Millan07] R. D. Millán, L. Dempere-Marco, J.M. Pozo, J.R. Cebral, and A.F. Frangi. Morphological characterization of intracranial aneurysms using 3-D moment invariants. 26(9) :1270–1282, 2007.
- [Mortara04] M. Mortara, G. Patanè, M. Spagnuolo, B. Falcidieno, and J. Rossignac. Plumber : a method for a multi-scale decomposition of 3D shapes into tubular primitives and bodies. In *SM '04 : Proceedings of the ninth ACM symposium on Solid modeling and applications*, pages 339–344, Aire-la-Ville, Switzerland, Switzerland, 2004. Eurographics Association.

- [Mortara06] M. Mortara, G. Patané, and M. Spagnuolo. From geometric to semantic human body models. *Comput. Graph.*, 30(2) :185–196, 2006.
- [Naidich09] Th.P. Naidich, H.M. Duvernoy, B.N. Delman, A.G. Sorensen, S.S. Kollias, and E.M. Haacke. *Duvernoy's atlas of the human brain stem and cerebellum*. 2009.
- [Neches91] R. Neches, R. Fikes, T. Finin, T. Gruber, R. Patil, T. Senator, and W. R. Swartout. Enabling technology for knowledge sharing. *AI Mag.*, 12 :36–56, September 1991.
- [Neuhaus08] F. Neuhaus and B. Smith. Modelling principles and methodologies : Relations in anatomical ontologies. In Albert Burger, Duncan Davidson, and Richard Baldock, editors, *Anatomy Ontologies for Bioinformatics*, volume 6 of *Computational Biology*, chapter 14, pages 289–306. Springer London, 2008.
- [Novotni04] M. Novotni and R. Klein. Shape retrieval using 3D zernike descriptors. *Computer-Aided Design*, 36(11) :1047–1062, 2004. Solid Modeling Theory and Applications.
- [Orte06] E. M. Orte, M. Rizo, D. Buss, and J. Graell. *Grand atlas d'anatomie humaine - Anatomie, Histologie, Pathologies*. 2006.
- [Palagyi01] K. Palagyi, E. Sorantin, E. Balogh, A. Kuba, C. Halmai, B. Erdohelyi, and K. Haussegger. A sequential 3D thinning algorithm and its medical applications. In *IPMI '01 : Proceedings of the 17th International Conference on Information Processing in Medical Imaging*, pages 409–415, 2001.
- [Palombi09] O. Palombi, G. Bousquet, D. Jospin, S. Hassan, L. Revéret, and F. Faure. My corporis fabrica : a unified ontological, geometrical and mechanical view of human anatomy. In *2nd workshop on 3D Physiological Human, 3DPH2009, December, 2009*, volume 5903 of *Lecture Notes in Computer Science*, pages 207–219, Zermatt, Suisse, December 2009. Springer.
- [Passat05] N. Passat. *Contribution à la segmentation des réseaux vasculaires cérébraux obtenus en IRM. Intégration de connaissance anatomique pour le guidage d'outils de morphologie mathématique*. PhD thesis, Université Louis Pasteur de Strasbourg, Septembre 2005.
- [Pisupati96] C. Pisupati, L. Wolff, W. Mitzner, and E. Zerhouni. Tracking 3-d pulmonary tree structures. In *Mathematical Methods in Biomedical Image Analysis, 1996., Proceedings of the Workshop on*, pages 160–169, 1996.
- [Pommert01] A. Pommert, K. H. Höhne, B. Pflessner, E. Richter, M. Riemer, T. Schiemann, R. Schubert, U. Schumacher, and U. Tiede. Creating a high-resolution spatial/symbolic model of the inner organs based on the visible human. *Medical Image Analysis*, 5(3) :221–228, 2001.
- [Pommert94] A. Pommert, R. Schubert, M. Riemer, U. Tiede, K. H. Höhne, and T. Schiemann. Symbolic modeling of human anatomy for visualization and simulation. In *Visualization in Biomedical Computing*, volume 2359, 1994.
- [Pudney98] C. Pudney. Distance-ordered homotopic thinning : A skeletonization algorithm for 3D digital images. *CVIU*, 72(3) :404–413, December 1998.
- [Rector93] A. Rector, W. Nowlan, and A. Glowinski. Goals for concept representation in the GALEN project. In *17th Annual Symposium on Computer Applications in Medical Care SCAMC93*, pages 414–418, 1993.
- [Rector94] A.L. Rector, A. Gangemi, E. Galeazzi, A.J. Glowinski, and A. Rossi-Mori. The GALEN CORE model schemata for anatomy : Towards a re-usable application-

- independent model of medical concepts. In *Twelfth International Congress of the European Federation for Medical Informatics, MIE-94*, pages 229–233, Lisbon, Portugal, 1994.
- [Rector96] A. L. Rector, J. E. Rogers, and P. A. Pole. The GALEN High Level Ontology. pages 174–178. IOS Press, January 1996.
- [Rogers01] J. Rogers, A. Roberts, D. Solomon, E. Haring, C. Wroe, P. Zanstra, and A. Rector. GALEN Ten Years On : Tasks and Supporting Tools. In R. Rogers, R. Haux, and V. Patel, editors, *MedInfo 2001 : Proceedings of the 10th World Congress on Health and Medical Informatics*, pages 256–260, Amsterdam, 2001. International Medical Informatics Association, IOS Press.
- [Rosse07] C. Rosse and J. L. V. Mejino. The foundational model of anatomy. In *Anatomy Ontologies for Bioinformatics*, chapter 4, pages 59–117. Springer, 2007.
- [Rosse95] M. Ben-Said K.R. Eno J.F. Brinkley Rosse, C. Enhancements of anatomical information in umls knowledge sources. In *Proc 19th Annu Symp Comput Appl Med Care (SCAMC 95)*, pages 873–877. Philadelphia : Hanley & Belfus, 1995.
- [Rubin04] D.L. Rubin, D. Bashir, Y.and Grossman, P. Dev, and M.A. Musen. Linking ontologies with three-dimensional models of anatomy to predict the effects of penetrating injuries. volume 2, pages 3128–3131, 2004.
- [Rubin06] D. L. Rubin, O. Dameron, Y. Bashir, D. Grossman, P. Dev, and M. A. Musen. Using ontologies linked with geometric models to reason about penetrating injuries. *Artificial intelligence in medicine*, 37(3) :167–176, July 2006.
- [Sander01] P. V. Sander, J. Snyder, S. J. Gortler, and H. Hoppe. Texture mapping progressive meshes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques, SIGGRAPH '01*, pages 409–416, New York, NY, USA, 2001. ACM.
- [Sato00] M. Sato, I. Bitter, M. A. Bender, A. E. Kaufman, and M. Nakajima. Teasar : Tree-structure extraction algorithm for accurate and robust skeletons. In *PG '00 : Proceedings of the 8th Pacific Conference on Computer Graphics and Applications*, page 281, Washington, DC, USA, 2000. IEEE Computer Society.
- [Shamir04] A. Shamir. A formulation of boundary mesh segmentation. In *Proceedings of the 3D Data Processing, Visualization, and Transmission, 2nd International Symposium, 3DPVT '04*, pages 82–89, Washington, DC, USA, 2004. IEEE Computer Society.
- [Shamir08] A. Shamir. A survey on mesh segmentation techniques. *Computer Graphics Forum*, 27(6) :1539–1556, 2008.
- [Simari09] P. Simari, D. Nowrouzezahrai, E. Kalogerakis, and K. Singh. Multi-objective shape segmentation and labeling. In *Computer Graphics Forum : Eurographics Symposium on Geometry Processing*, Aire-la-Ville, Switzerland, Switzerland, March 2009. Eurographics Association.
- [Simons87] Peter Simons. *Parts : A Study in Ontology*. Routledge, 1987.
- [Smith04] B. Smith. Beyond Concepts : Ontology as Reality Representation. In A. Varzi and L. Vieu, editors, *Proceedings of FOIS*, 2004.
- [Smith05] B. Smith, W. Ceusters, B. Klagges, J. Kohler, A. Kumar, J. Lomax, C. Mungall, F. Neuhaus, A. Rector, and C. Rosse. Relations in biomedical ontologies. *Genome Biology*, 6(5) :R46+, 2005.

- [Solomon11] J. Solomon, M. Ben-Chen, A. Butscher, and L. Guibas. Discovery of intrinsic primitives on triangle meshes. *Computer Graphics Forum*, 2011.
- [Studer98] R. Studer, V. R. Benjamins, and D. Fensel. Knowledge engineering : principles and methods. *Data Knowledge Engineering*, 25 :161–197, March 1998.
- [Tang06] W. H. Tang and A. C. S. Chung. Cerebral vascular tree matching of 3D-RA data based on tree edit distance. In *International Workshop on Medical Imaging and Augmented Reality*, 2006.
- [Tang07] P.H. Tang, F. Hui, and Y.Y. Sitoh. Intracranial aneurysm detection with 3T magnetic resonance angiography. *Annals Academy of Medicine*, 36 :388–393, 2007.
- [Tierny07] J. Tierny, J.-P. Vandeboorde, and M. Daoudi. Topology driven 3D mesh hierarchical segmentation. In *SMI '07 : Proceedings of the IEEE International Conference on Shape Modeling and Applications 2007*, pages 215–220, Washington, DC, USA, 2007. IEEE Computer Society.
- [Torczon97] V. Torczon. On the convergence of pattern search algorithms. *SIAM J. on Optimization*, 7(1) :1–25, 1997.
- [Tschirren02] J. Tschirren, K. Palágyi, J. M. Reinhardt, E. A. Hoffman, and M. Sonka. Segmentation, skeletonization, and branchpoint matching - a fully automated quantitative evaluation of human intrathoracic airway trees. In *MICCAI '02 : Proceedings of the 5th International Conference on Medical Image Computing and Computer-Assisted Intervention-Part II*, pages 12–19, London, UK, 2002. Springer-Verlag.
- [Ujiie99] H. Ujiie, H. Tachibana, O. Hiramatsu, A. Hazel, T. Matsumoto, Y. Ogasawara, H. Nakajima, T. Hori, K. Takakura, and F. Kajiya. Effectes of size and shape (aspect ratio) on the heomdynamics of saccular aneurysms : a possible index for surgical treatment of intracranial aneurysms. *Neurosurgery*, 45 :119–130, 1999.
- [Ullmann76] J. R. Ullmann. An algorithm for subgraph isomorphism. *J. ACM*, 23(1) :31–42, 1976.
- [Wan02] M. Wan, Z. Liang, Q. Ke, L. Hong, I. Bitter, and A. E. Kaufman. Automatic centerline extraction for virtual colonoscopy. *IEEE Transactions on Medical Imaging*, 21 :1450–1460, 2002.
- [Wier02] B. Weir. Unruptured intracranial aneurysms : a review. *J Neurosurgery*, 96 :3–42, 2002.
- [Wilson97] D. L. Wilson and J. A. Noble. Segmentation of cerebral vessels and aneurysms from mr angiography data. In *IPMI '97 : Proceedings of the 15th International Conference on Information Processing in Medical Imaging*, pages 423–428, London, UK, 1997. Springer-Verlag.
- [Wink04] O. Wink, W.J. Niessen, and M.A. Viergever. *Medical Image Analysis*, 23(1) :130–133, January 2004.
- [Wu05] J. Wu and L. Kobbelt. Structure recovery via hybrid variational surface approximation. In *Eurigraphics 2005*, volume 24, pages 277–284, 2005.
- [Yan06] D.M. Yan, Y. Liu, and W.P. Wang. Quadric surface extraction by variational shape approximation. In *GMP06*, pages 73–86, 2006.
- [nominaAnatmica] Churchill Livingstone, editor. *Nomina Anatomica*. Edimburgh, 6 edition, 1989.
- [schnabel2007] R. Schnabel, R. Wahl, and R. Klein. Efficient RANSAC for point-cloud shape detection. *Computer Graphics Forum*, 26(2) :214–226, June 2007.

- [vanKaick11] O. van Kaick, A. Tagliasacchi, O. Sidi, H. Zhang, D. Cohen-Or, L. Wolf, and G. Hamarneh. Prior knowledge for shape correspondence. *Computer Graphics Forum (Special Issue of Eurographics)*, 30(2) :1–10, 2011.