



**HAL**  
open science

# Structuration topologique pour le rendu et la visualisation 3D

Sébastien Horna

► **To cite this version:**

Sébastien Horna. Structuration topologique pour le rendu et la visualisation 3D. Informatique [cs]. université de Poitiers, 2022. tel-04037427

**HAL Id: tel-04037427**

**<https://hal.science/tel-04037427>**

Submitted on 20 Mar 2023

**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.

Public Domain



## UNIVERSITÉ DE POITIERS

### ÉCOLE DOCTORALE SISMI

SCIENCES ET INGENIERIE DES SYSTEMES, MATHEMATIQUES, INFORMATIQUE

Secteur de recherche : Informatique Graphique

## HABILITATION À DIRIGER DES RECHERCHES

Présentée à L'Université de Poitiers par

**Sébastien Horna**

---

### *Structuration topologique pour le rendu et la visualisation 3D.*

---

Préparée à l'institut XLIM, UMR CNRS 7252

Soutenue le 22 septembre 2022 devant la Commission d'Examen

**Jury :**

**Rapporteurs :**

Jean Luc Mari – Professeur..... Université d'Aix Marseille, LIS  
Céline Loscos –Professeur.....Université de Reims, CRESTIC  
Christophe Renaud – Professeur.....Université du Littoral, LISIC

**Examineurs :**

Guillaume Damiand – Directeur de Recherche, CNRS....., Université de Lyon, LIRIS  
Philippe Meseure – Professeur .....Université de Poitiers, XLIM  
Daniel Meneveau – Professeur..... Université de Poitiers, XLIM



---

*... lointaine, très lointaine...*

---



---

---

# REMERCIEMENTS

---

Nous y voilà... L'exercice aussi plaisant que périlleux d'écrire la page des remerciements. Sans aucun doute les lignes de ce mémoire qui seront les plus lues. Je me revois une quinzaine d'années en arrière, rédigeant les remerciements de mon mémoire de thèse et plaisantant avec mes compagnons doctorants sur une hypothétique *Habilitation à Diriger des Recherches*. Sans aucune prétention, je peux dire que depuis longtemps je réfléchis à ce que je pourrai écrire, à comment tourner les choses, à comment trouver les bons mots pour exprimer toute la gratitude que j'ai envers toutes les personnes qui m'ont accompagné jusqu'à aujourd'hui. Nous y voilà...

Je remercie l'ensemble des membres de mon jury. J'éprouve un profond respect pour leur travail et leur parcours. Je les ai choisis pour leur qualité scientifique, leur connaissance au plus proche de mes travaux de recherche, mais aussi et surtout pour leur qualité humaine. Les différents échanges que nous avons pu avoir tout au long de la préparation de cette habilitation m'ont conforté dans ce choix. J'ai vraiment ressenti une grande bienveillance et un soutien à chacune des étapes. C'est sans aucun doute l'une des particularités de la préparation de l'HDR, on se retrouve seul aux manettes administratives et organisationnelles. La réactivité et les petits mots gentils m'ont apporté beaucoup de réconfort dans ce timing de rédaction plutôt serré.

Le travail présenté lors d'une HDR n'est pas un travail individuel. C'est pour moi primordial de le souligner. Et je souhaite remercier toutes les personnes qui m'ont accompagné de près ou de loin dans mes différents projets scientifiques et pédagogiques. Les membres d'XLim Poitiers et plus particulièrement de l'équipe IG, mes collègues de la faculté de droit, l'ensemble des personnes que j'ai pu rencontrer au fil de ces années. Elles sont nombreuses et une grande difficulté tient dans le fait de n'oublier personne. C'est pourquoi je remercie par avance ceux dont le nom n'apparaît pas dans ces pages et qui m'ont aidé d'une manière ou d'une autre. Ils se reconnaîtront.

En premier lieu, un grand merci aux 5 doctorants que j'ai la chance d'encadrer (ou d'avoir encadré) : Maxime, Simon, Elsa, Sylvain et Hayssam. J'imagine qu'ils ont retrouvé les traces de leur travail à travers mon manuscrit et ma présentation. C'est une démonstration flagrante qu'ils sont les éléments moteurs de la recherche. J'ai pour ma part toujours éprouvé un immense plaisir à les suivre au jour le jour (un peu trop tous les jours peut-être), à les voir grandir et à les voir prendre certaines de mes bonnes ou mauvaises habitudes. "*Toujours par deux, ils vont*", cette relation encadrant / doctorant est très particulière, j'y accorde une grande importance, c'est une grande responsabilité. Finalement, c'est sans doute la seule activité de notre métier où nous sommes à la fois enseignant et chercheur.

Au fil des ans, on gagne en expérience, on apprend à se connaître. Pour ma part, je sais maintenant que j'ai besoin d'être entouré d'une équipe. J'accorde volontiers ma confiance à toutes les personnes avec qui je travaille. Ainsi, les barrières s'amenuisent, les relations professionnelles se transforment en liens d'amitié, et mon bureau devient officiellement la cafétéria du 3<sup>e</sup> étage... The place to be... Rita, Arthur, Hakim, Daniel, Philippe, Bruno, bien sur Mickaël, et tous les autres qui passent la porte pour boire un café, pour papoter, et même parfois pour bosser, surtout, faites que ça ne change pas!!! Mais je tiens également à tous vous remercier pour la confiance que vous m'avez donnée petit à petit. C'est sans aucun doute votre regard, votre reconnaissance, qui m'a m'y sur les rails de la rédaction de cette habilitation.

Et puis un grand merci à toute ma famille, mes parents, ma soeur Cyrielle, Thomas, et Lulu. Même si je me suis lancé dans ce travail d'HDR un peu en catimini, il était bien difficile de cacher ma joie et ma fierté de vous annoncer que c'était fait et que je vous attendais tous pour ma soutenance. Les moments que l'on passe tous ensemble sont sans aucun doute trop rares, cela les rend encore plus précieux. Il était pour moi inconcevable de passer cette étape sans vous avoir à mes côtés.

Enfin, comment trouver les mots... Que dire sur mes 3 princesses, mes 3 petits coeurs, mes 3 formules 1... Si mon boulot me donne l'impression d'avoir une vie à 100 à l'heure, alors à la maison on dépasse les 200. Nous formons une dream team prête à conquérir le monde. Ma petite Maëline, qui respire la joie de vivre à chaque seconde, pleine de fraîcheur, pleine de malice, tu me fais vivre un véritable feu d'artifice jour après jour. Ma grande Matilde, 10 ans déjà que tu as fait de moi un papa, j'ai à peine le temps de te voir grandir que tu passes aujourd'hui les portes du collège, ta petite vie est déjà tellement remplie et pourtant ta soif de tout connaître ne cesse de me surprendre. Et ma chérie, ma Magali, ma copilote depuis maintenant 26 ans... Tout un symbole pour nous. Il y a 26 ans, on se demandait où l'on serait en 2022, on en rigolait parce que c'était loin, c'était un autre monde. Certes, depuis le monde a beaucoup changé, mais nous, nous sommes restés ensemble. Une magnifique aventure qui a dépassé nos espérances les plus folles. Tu as toujours cru en moi, tu m'as toujours soutenu, tout est grâce à toi.

Merci,

*Les choses, on les fait ou on ne les fait pas...*





---

---

# SOMMAIRE

---

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Travaux réalisés . . . . .	5
1.2	Collaborations et encadrements . . . . .	7
1.3	Responsabilités . . . . .	8
1.4	Structure du document . . . . .	8
<b>2</b>	<b>Contexte et positionnement</b>	<b>9</b>
2.1	Gestion de scènes 3D en rendu réaliste par lancé de rayons . . . . .	10
2.1.1	Structures accélératrices classiques . . . . .	12
2.1.2	Partitionnement Spatial Contraint CSP . . . . .	13
2.1.3	Gestion de détails fins, les arrondis . . . . .	16
2.2	Modélisation à base topologique . . . . .	17
2.2.1	Modèles topologiques usuels . . . . .	18
2.2.2	Cartes généralisées concept et notations . . . . .	21
2.3	Reconstruction de modèles 3D à partir de données . . . . .	25
2.3.1	Reconstruction de maillages . . . . .	25
2.3.2	Reconstruction à partir de donnée médicales . . . . .	26
2.4	Positionnement des travaux . . . . .	28
2.4.1	Structure topologique pour la simulation d'éclairage . . . . .	28
2.4.2	Reconstruction topologique : application en imagerie médicale . . . . .	30
<b>3</b>	<b>Partition topologique contrainte pour le lancé de rayons</b>	<b>31</b>
3.1	Modèle topologique dédié aux bâtiments : CCSP . . . . .	32
3.1.1	Construction des volumes convexes . . . . .	34
3.1.2	Correspondance de GMapArchi à une CCSP . . . . .	34
3.2	Utilisation de la CCSP pour le rendu réaliste . . . . .	35
3.2.1	Traversée de la CCSP . . . . .	35
3.2.2	Utilisation étendue de la structure CCSP . . . . .	39
3.2.3	Gestion du mobilier . . . . .	40
3.3	Résultats et discussion . . . . .	41
3.3.1	Système d'évaluation des performances . . . . .	41
3.3.2	Etude de la complexité du calcul des rayons . . . . .	43
3.3.3	Analyse comparative des méthodes . . . . .	44
3.3.4	Bilan . . . . .	45
3.4	Visualisation topologique : généralisation . . . . .	45
3.4.1	Construction de la CDT . . . . .	46
3.4.2	Traversée de la CDT . . . . .	46
3.4.3	Résultats . . . . .	48

3.4.4	Bilan . . . . .	50
3.5	Conclusion . . . . .	50
<b>4</b>	<b>Visualisation d'arrondis réalistes</b>	<b>53</b>
4.1	Construction de la structure accélératrice SREC-RT . . . . .	55
4.1.1	Positionnement des surfaces arrondies . . . . .	56
4.1.2	Structure de détection . . . . .	57
4.1.3	Structure Topologique . . . . .	58
4.2	Utilisation de SREC-RT en lancé de rayons . . . . .	59
4.2.1	Processus détaillé . . . . .	59
4.2.2	Utilisation de la structure topologique de détection . . . . .	59
4.2.3	Cas particulier : sommets concaves . . . . .	61
4.3	Représentation des arrondis . . . . .	61
4.3.1	Representation et intersection des arêtes arrondies . . . . .	62
4.3.2	Representation et intersection des sommets arrondis . . . . .	62
4.4	Résultats . . . . .	64
4.4.1	Temps de rendu et impact mémoire . . . . .	64
4.4.2	Comparaison visuelle des arrondis . . . . .	66
4.4.3	Interpolation des normales . . . . .	67
4.5	Conclusion . . . . .	68
<b>5</b>	<b>Reconstruction topologique d'imagerie médicale</b>	<b>71</b>
5.1	Correction et reconstruction topologique . . . . .	73
5.1.1	Processus de reconstruction topologique . . . . .	74
5.1.2	Corafinement 3D . . . . .	75
5.1.3	Résultats et discussion . . . . .	76
5.2	Reconstruction 3D de données médicales à multi label sémantiques . . . . .	78
5.2.1	Méthodes de reconstruction étudiées . . . . .	79
5.2.2	Etude comparative . . . . .	80
5.2.3	Bilan . . . . .	82
5.2.4	Modèle topologique pour le médical . . . . .	84
5.2.5	Mise en oeuvre détaillée . . . . .	84
5.3	Résultats et comparaisons . . . . .	91
5.3.1	Statistiques et performances . . . . .	92
5.3.2	Comparaison avec des méthodes de reconstruction classiques . . . . .	93
5.3.3	Comparaison avec FreeSurfer . . . . .	95
5.4	Conclusion . . . . .	96
<b>6</b>	<b>Conclusion</b>	<b>99</b>
	<b>Annexes</b>	<b>103</b>
<b>A</b>	<b>Collaborations et encadrements</b>	<b>103</b>
<b>B</b>	<b>Publications Personnelles</b>	<b>107</b>
<b>C</b>	<b>Responsabilités</b>	<b>111</b>
	<b>Bibliographie</b>	<b>115</b>

---

# INTRODUCTION

---

Les applications de simulation numérique sont devenues incontournables dans de nombreux secteurs industriels : mécanique des solides, aérodynamique, écoulement de fluides, éclairage, etc. L'ensemble de ces applications nécessite de développer des modèles physiques et des processus de calculs lourds et coûteux en temps de calculs et en utilisation mémoire. Par exemple, dans l'étude des sous-sols géologiques, des méthodes de simulation d'écoulement sont utilisées pour détecter les réservoirs d'hydrocarbures (cf. figure 1.1.a). Dans le domaine de la synthèse d'image photoréaliste qui nous intéresse plus particulièrement, les calculs de simulation d'éclairage nécessitent une expression rigoureuse des phénomènes physiques (cf. figure 1.1.b). L'objectif est de calculer les échanges lumineux entre les surfaces d'un environnement en fonction du positionnement des objets, des types de sources lumineuses, des matériaux qui composent les surfaces. Ces travaux permettent la visualisation d'environnements virtuels utilisés dans de nombreux secteurs comme par exemple des études de conception (positionnement des éclairages dans un bâtiment, visualisation de carrosserie, etc.), le divertissement (films d'animation, jeux vidéo), la reconstitution de patrimoine (musées virtuels, études historiques) ou toute autre application de réalité virtuelle nécessitant la génération d'images photoréalistes.

Quel que soit le domaine d'application, avant de procéder aux processus de simulation, l'environnement virtuel doit être modélisé (cf. figure 1.2). La modélisation *géométrique* correspond à l'ensemble des outils mathématiques, numériques et informatiques qui permettent de construire la forme des objets virtuels. La géométrie est inévitablement fournie pour décrire la forme des objets. Elle est plus ou moins complexe, plus ou moins contrainte, et contient des informations spécifiques utilisées dans un second temps lors des algorithmes de simulation. Le processus de construction géométrique consiste à réaliser la construction et l'assemblage de formes élémentaires pour créer des objets de plus en plus complexes. Il peut également nécessiter de reconstruire des objets à partir de données existantes (modélisation architecturale à partir de plan, visualisation à partir d'imagerie médicale, photogrammétrie, etc.). Finalement, les traitements numériques et graphiques effectués sur des objets modélisés numériquement découlent du modèle de représentation choisi. Les résultats obtenus et la complexité des algorithmes tant en modélisation qu'en simulation sont grandement influencés par le choix de cette structure de données.

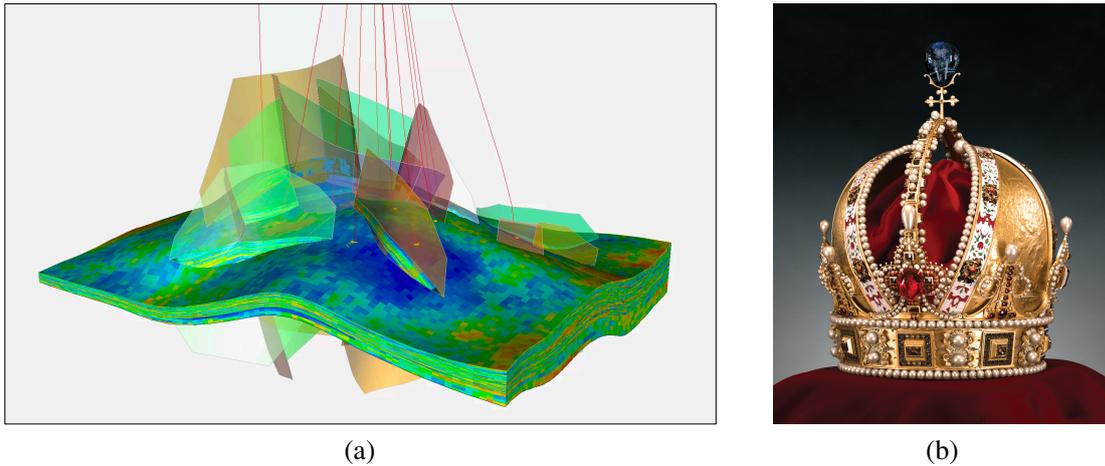


Figure 1.1 – Applications de simulation : (a) la suite logicielle *GOCAD Mining Suite* [1] est une plateforme pour la modélisation 3D qui intègre des données géologiques, géophysiques, géochimiques, structurales et géotechniques ; (b) la simulation d'éclairage par lancé de rayon permet de produire des images au plus proche de la réalité. Ici une image générée avec le logiciel *LuxRender*, inspiré du moteur de rendu open source *pbrt* [127]

Les *modeleurs géométriques* disposent de leur propre structure de données. Les modèles de description géométrique peuvent être de différentes natures en fonction des applications : un ensemble de triangles dépourvu de relation explicite (on parle de soupe de triangles), une liste de polygones, convexes ou non, liés afin de produire des surfaces, un maillage volumique structuré et agrémenté de relations de voisinage.

Pour mieux appréhender la structure de l'environnement, certains modeleurs disposent d'informations de voisinage entre les éléments, des informations dites "*topologiques*". Elles permettent de représenter des objets subdivisés, c'est-à-dire partitionnés en cellules de différentes dimensions (sommets, arêtes, faces, volumes). Cette structuration est primordiale pour de nombreux domaines d'application car elle fournit des informations qui permettent de manipuler efficacement les maillages. L'utilisation de ce type de représentation est appelée modélisation géométrique à base topologique. Elle permet la manipulation d'objets subdivisés, le contrôle de l'environnement produit (interpénétration d'objets, cohérence des voisinages), l'application d'opérations locales (intersections, opérations booléennes, chanfreinage, etc.), l'ajout d'informations sémantiques locales rattachées aux cellules. Par exemple, en géoscience, il est important de représenter les couches géologiques qui composent le sous-sol. Ces couches sont fracturées et composées de différents blocs correspondants géométriquement aux volumes, liés par leurs faces de contact. En architecture, pour des études de simulation thermique, acoustique, d'éclairage, il est important de représenter les différents volumes (pièces, murs, portes, etc.), de connaître les ouvertures ouvrant sur une pièce. Les cellules qui composent l'objet sont donc représentées, mais également leurs relations d'incidence et d'adjacence.

La géométrie seule n'est pas suffisante pour représenter un environnement virtuel. Divers attributs lui sont associés pour définir des propriétés sur les éléments géométriques. Dans le contexte du rendu, ces attributs peuvent par exemple correspondre aux données photométriques associées aux surfaces : couleur, texture, matériaux, réflectance, luminance, etc. Plus généralement, des informations sémantiques sont utilisées pour caractériser les objets et fournir des données utilisées dans des applications de simulation : composition d'un sous-sol géologique, concentration physiologique dans le médical, matériaux de construction utilisés dans un bâtiment.

Finalement, l'environnement 3D est produit à partir d'un processus qui reconstruit les informations à partir de données externes et/ou d'un ensemble d'opérations de construction et d'édition (cf. figure 1.3.a).

Parmi la multitude de contextes d'utilisation des environnements 3D, la génération d'images photoréalistes par simulation d'éclairage est un domaine de recherche à part entière, très exigeant quand au temps de traitement et à la qualité du résultat. L'objectif est de simuler le transport de la lumière dans des environnements aux interactions complexes entre lumière et matière (transparence, translucidité, brillance, anisotropie, surfaces multicouches, etc.). Au cours des premières années de l'infographie, dans les années 1970, les problèmes scientifiques concernaient des questions fondamentales comme les algorithmes de visibilité et les représentations géométriques. Les ressources matérielles étaient particulièrement limitées et les ordinateurs capables d'effectuer un million d'opérations en virgule flottante par seconde coûtaient des centaines de milliers de dollars. Tout calcul de simulation précis de la physique pour le rendu était alors impossible.

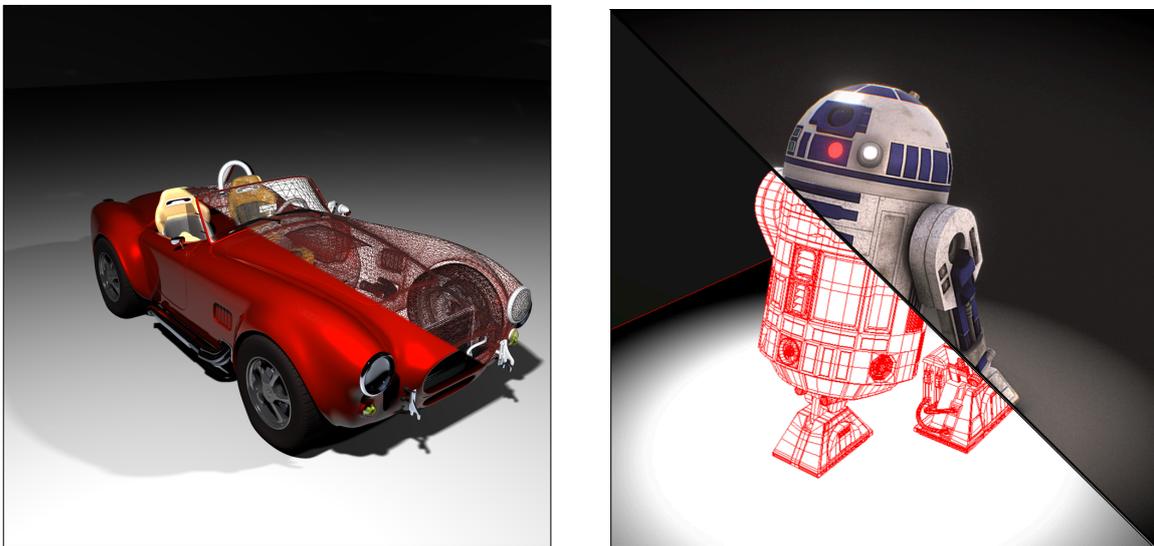


Figure 1.2 – Dans le contexte du rendu réaliste, les environnements 3D sont modélisés dans un premier temps à l'aide d'un logiciel de modélisation (un modelleur) qui permet de définir la forme géométrique des objets et l'ensemble de leur attributs (matériaux, propriétés physiques, etc.) Dans un second temps, un processus de rendu est utilisé pour générer l'image de synthèse.

L'étude des approches de rendu basées sur la physique a débuté dans les années 1980 avec l'introduction du lancer de rayons par Whitted [167]. Parallèlement, les premiers modèles de réflexion microfacette ont été introduits pour améliorer l'apparence des matériaux métalliques [35]. Les premières méthodes de simulation d'éclairage ont été proposées dans les années 1980 pour les effets d'éclairage indirect avec la méthode de *radiosité* de Goral et al. en établissant le lien entre les transferts d'énergie et le rendu [63]. Mais bien que fortement basée sur les unités physiques et la conservation de l'énergie, la radiosité ne conduit pas à des algorithmes de rendu viables. La complexité asymptotique des calculs, ou la nécessité de tesseler les modèles géométriques pour obtenir de bons résultats ont démontré que la radiosité est limitée dans la pratique. En parallèle, les études concernant les approches physiques du rendu basées sur le lancer de rayons et l'intégration de Monte-Carlo ont été poursuivies. Dans un premier temps, ces méthodes produisaient des images contenant du bruit dû à la variance de l'intégration Monte-Carlo, donc de moindres qualités que les méthodes de radiosité. En 1986, Kajiya a établi une formulation rigoureuse du problème de rendu (l'équation intégrale du transport de la lumière) et a montré comment appliquer l'intégration de Monte-Carlo pour résoudre le calcul [89].

À partir des années 2000, les processeurs étant de plus en plus performants et les capacités mémoire augmentant, de nombreux articles ont présenté des algorithmes de simulation d'éclairage de plus en plus efficaces basés sur des propriétés physiques [160, 163]. Avec la résolution de ces verrous scientifiques et l'évolution technologique, il est devenu possible d'envisager des approches de plus en plus précises et efficaces, mais exigeantes en temps de calcul. L'objectif du rendu photoréaliste est de créer une image d'une scène 3D très complexe (plusieurs millions de faces) qui soit indiscernable<sup>1</sup> d'une photographie. Pour cela, le parcours de la lumière est simulé par le lancé de plusieurs milliards de rayons au travers de la scène. Pour répondre à cette problématique, une partie des recherches concerne le développement de structures accélératrices qui représentent l'environnement dans une organisation spatiale dédiée aux méthodes de simulation d'éclairage. Ces modèles de représentation ont ensuite été pensés et adaptés aux contraintes techniques des processeurs de cartes graphiques (GPU) et en exploitant uniquement les données géométriques de l'environnement.

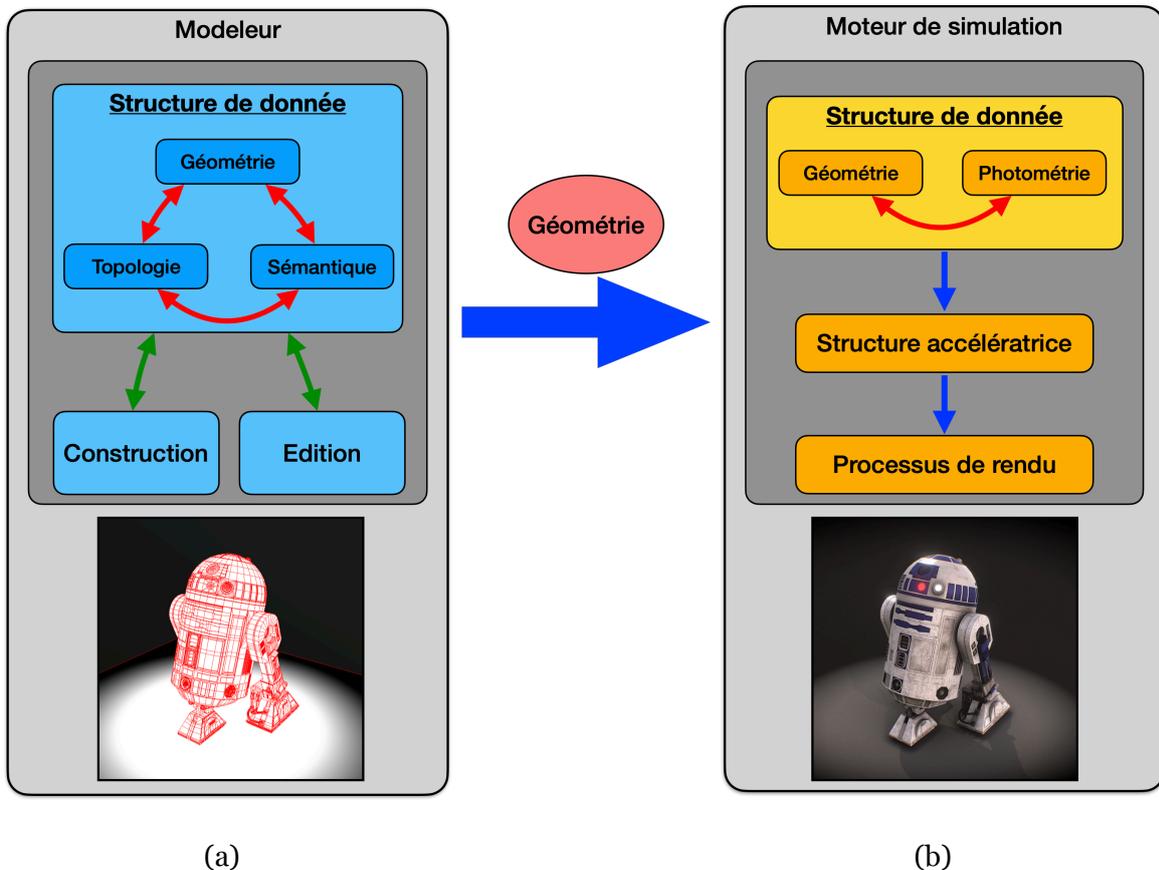


Figure 1.3 – Processus de création d'images de synthèse : (a) l'environnement est créé avec un modeleur qui dispose d'une structure de données contenant des informations géométriques, topologiques et sémantiques. La scène 3D est créée par des opérations de construction et d'édition ; (b) le moteur de simulation construit une structure de données à partir de la géométrie fournie par le modeleur, une structure accélératrice dédiée est produite pour réduire les temps de traitement.

La structure de données utilisée par le moteur de rendu est généralement construite uniquement à partir des informations géométriques des objets (cf. figure 1.3.b). Les informations sémantiques telles que les données photométriques sont conservées, mais non utilisées dans la construction de la structure

1. Notons que le terme "indiscernable" est imprécis car il implique un observateur humain, et différents observateurs peuvent percevoir la même image différemment.

accélétratrice. Ce procédé permet de s'abstraire du modèle de représentation utilisé lors de la phase de modélisation de l'environnement, rendant le processus de rendu indépendant des données d'entrée. La structure accélératrice produite est optimisée pour l'algorithme de simulation et peut être adaptée à la parallélisation sur GPU. Malheureusement le modèle de représentation utilisée au cours de l'étape de modélisation, aussi complet qu'il soit, est totalement édulcoré.

L'équipe de *IG*, spécialisée en informatique graphique au sein du laboratoire XLIM développe ses travaux de recherche dans le domaine de la synthèse d'images avec une partie importante concernant la modélisation géométrique, le rendu réaliste et la simulation d'éclairage. J'ai intégré cette équipe en décembre 2010 en tant que maître de conférence. Mon projet scientifique au sein de cette équipe est de mettre à contribution les structures topologiques et hiérarchiques pour améliorer les opérations de reconstruction 3D et optimiser les algorithmes de simulation d'éclairage par lancé de rayons. Au cours de ces années de recherche, j'ai participé à l'encadrement de 5 thèses de doctorat portant sur ces thématiques et j'ai contribué aux montages de plusieurs projets région et ANR. Je suis notamment le porteur du projet MEDIA-IRM 3D, un projet pluridisciplinaire impliquant des équipes de mathématiques, traitement d'images et informatique, des équipes du secteur biologie/santé et des médecins. La problématique centrale de ce projet réside dans la définition et l'utilisation d'un modèle de représentation globale et cohérente qui répond aux besoins de simulation statistique pour l'étude des données physiologiques cérébrales.

## 1.1 Travaux réalisés

Le projet de recherche que je propose est en partie issu d'un modèle de représentation formel construit sur la base d'un ensemble de contraintes topologiques et sémantiques dédié à la modélisation de complexes architecturaux 3D. Nous avons proposé d'utiliser ce modèle pour une application de reconstruction de bâtiment en 3D à partir de plan 2D et nous avons démontré que cette même structure était directement utilisable pour les applications de simulation d'éclairage par lancé de rayons. Les travaux présentés dans ce document se placent dans la continuité de ces recherches. Nous souhaitons étudier l'apport des modèles de représentation utilisés lors de la phase de modélisation pour améliorer les algorithmes de simulation. Cette étude a permis de définir et d'utiliser des modèles géométriques et topologiques pour représenter et éditer des environnements complexes (scènes architecturales, espace géologique, objets complexes). L'objectif principal est de mettre directement à contribution ces informations topologiques et d'utiliser des structures hiérarchiques pour améliorer les opérations de modélisation et optimiser les méthodes de simulation en utilisant des structures de voisinage (cf. figure 1.4).

Les paragraphes suivants déclinent les principaux projets engagés ces dernières années dans l'utilisation des modèles à base topologique pour les processus de rendu réaliste et la modélisation géométrique à base topologique.

### Processus de rendu et calculs d'éclairage

Les modèles topologiques étant plus riches que ceux habituellement employés dans les méthodes de simulation d'éclairage, nous avons étudié la mise à contribution des relations de voisinage souvent ignorées pour améliorer la rapidité des calculs en simulation d'éclairage. Nous avons proposé au cours de la thèse de Maxime Maria (2012-2016), une méthode permettant d'utiliser notre modèle topologique dédié à l'architecture comme structure accélératrice afin de calculer le parcours des rayons (correspondant aux photons) à travers une scène 3D. La méthode résultante nous a permis de développer une application de visualisation interactive de scènes d'intérieur de bâtiment composé de plusieurs centaines de millions de polygones.

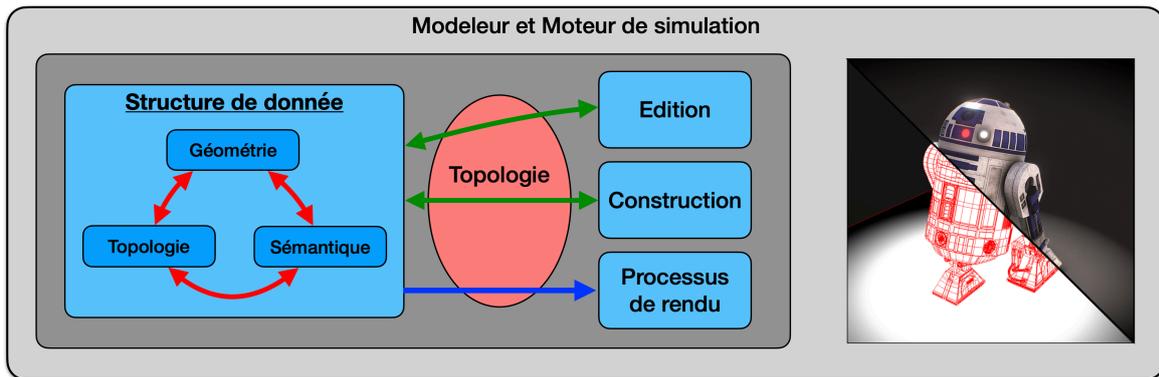


Figure 1.4 – Notre objectif est de mettre à contribution les données topologiques pour améliorer les opérations de reconstruction 3D et optimiser les algorithmes de simulation d'éclairage par lancé de rayons.

Nous avons ainsi mis en évidence que la répartition spatiale des primitives géométriques peut mettre en défaut certaines structures accélératrices usuelles (grandes faces occultantes, petits objets regroupés). Notre structure topologique et hiérarchique se trouvant plus efficace, nous avons étendu notre étude au rendu de détails fins présents dans les scènes complexes. L'objectif principal est d'aborder avec le plus d'exhaustivité possible l'ensemble des effets hautes fréquences produits par de petits objets dont la couverture sur l'image est inférieure à un pixel (par exemple des reflets lumineux sur le bord des objets). Ces effets sont difficiles à traiter de manière robuste et nous souhaitons mettre en évidence l'intérêt d'utiliser une structure hiérarchique et topologique dans ces contextes particuliers. Ce travail se place dans le cadre du projet Région SIMOREVA (2018) dont l'axe de recherche principal concerne la perception et le rendu réaliste de scènes virtuelles. Nos travaux sur le rendu de détails fins ont été réalisés en collaboration avec Pierre Poulin (Ligum de l'Université de Montréal), notamment avec le co-encadrement des thèses de Simon Courtin et Elsa Tamisier qui ont débuté respectivement en 2018 et 2019.

### Modélisation géométrique à base topologique

Pour mieux contrôler la production des scènes 3D, nous avons travaillé sur les méthodes de reconstruction d'environnement complexes 3D à partir de données numériques, et plus particulièrement dans les domaines de l'architecture et du médical. La reconstruction de bâtiments en 3D à partir de données vectorielles est le sujet principal de ma thèse de doctorat (2008). Par la suite, nous avons tout d'abord généralisé et étendu le processus proposé pour des problématiques de reconstruction et de visualisation de modèles 3D à partir des données médicales. Dans le cadre du projet région IMPROVED débuté en 2018, nous avons réalisé une première étude exploratoire pour comprendre et interpréter les données fournies par un procédé d'acquisition tel qu'une IRM. Nous avons ensuite approfondi ces travaux avec le projet région MEDIA-IRM 3D (MEDical Data Intelligent Analysis, Interpretation, Reconstruction and Manipulation in 3D - 2020) dont je suis porteur. Dans ce projet, l'objectif principal est de proposer une représentation formelle basée sur des contraintes géométriques, topologiques et sémantiques dédiées à la représentation d'un modèle médical 3D. Ce modèle doit être suffisamment riche pour : proposer une méthode de reconstruction automatique d'objets anatomiques 3D; réaliser des calculs statistiques de simulation de propagation de données métabolites (physiologiques); étudier la croissance de tumeurs cérébrales; visualiser les données multimodales. Dans le cadre de ce projet, la thèse de Sylvain Gerbaud traite de la reconstruction de modèles 3D topologiques à partir d'images, et de l'utilisation de ce modèle pour l'estimation des données métabolites au sein du cortex cérébral. La thèse de Hayssam Obeid qui vient de débuter traite de la visualisation multimodale des informations métabolites afin de proposer un outil d'analyse et de détection des tumeurs cérébrales.

## 1.2 Collaborations et encadrements

Depuis ma nomination en tant que maître de conférences à l'université de Poitiers, j'ai participé à l'encadrement des thèses et aux projets cités ci-dessous. Les travaux présentés dans ce manuscrit sont principalement associés aux projets [2 et 4] et aux encadrements [2, 3, 5 et 6]. Une liste plus complète des collaborations et encadrement est disponible en annexe A.

### Collaborations Nationales et Internationales

- [1] 2021 : Projet région ARTERIA  
Partenaires : CHU de Poitiers, Siemens Healthineers, l'université de Poitiers.
- [2] 2019 : Porteur du projet Région MEDIA IRM 3D  
Partenaires : CHU de Poitiers et Limoges, Siemens Healthineers, les universités de Poitiers, Limoges et La Rochelle, ainsi que les partenaires internationaux de l'université de Balamand (Liban) et de Heriot Watt University (Ecosse).  
Cotutelle de thèse : université de Balamand au Liban (Dr. Mohammad Khalil)
- [3] 2018 : Projet région IMPROVED  
Partenaires : CHU de Poitiers, les universités de Poitiers et La Rochelle.
- [4] 2018 : Projet région SIMOREVA  
Partenaires : universités de Poitiers (XLim), Montréal (LIGUM), l'INRIA, le L2PN de l'Institut d'Optique et le NTNU colorlab en Norvège  
Cotutelle de thèse : université de Montréal, laboratoire LIGUM (Dr : Pierre Poulin)

### Encadrements de thèse et post doctorat

- [1] Hayssam Obeid (thèse débutée en 2021 en cotutelle avec l'université Libanaise).  
Visualisation et interaction avec les informations anatomiques et métaboliques  
Directeurs : P. Meseure et M. Khalil, encadrants : R. Zrour et **S. Horna**.
- [2] Sylvain Gerbaud (thèse débutée en 2020)  
Développement de nouvelles méthodes de reconstructions tissulaires basées sur le métabolisme sous-jacent exploré en spectroscopie par résonance magnétique multi-noyaux,  
Directeur : P. Meseure, co-directeur : **S. Horna (ACT<sup>2</sup>)**, encadrant : R. Zrour.
- [3] Loubna Lecheleck (post doctorat de 18 mois 2020 - 2021).  
Reconstruction de modèles 3D à partir de données médicales.  
Encadrants : **S. Horna** et R. Zrour
- [4] Elsa Tamisier (thèse débutée en 2019 en cotutelle avec l'université de Montréal).  
Représentations géométriques de détails fins pour la simulation d'éclairage  
Directeurs : D. Meneveaux et P. Poulin, encadrants : M. Ribardière et **S. Horna**.
- [5] Simon Courtin (thèse débutée en 2018 - soutenance prévue en septembre 2022).  
Prise en compte des effets hautes fréquences en simulation d'éclairage,  
Directeur : D. Meneveaux, co-directeur : **S. Horna (ACT)**, encadrant : M. Ribardière.
- [6] Maxime Maria (thèse soutenue en 2016).  
Utilisation d'une partition spatiale contrainte et convexe pour la simulation basée rayons,  
Directeur : L. Aveneau, co-directeur : **S. Horna (ACT)**.

### 1.3 Responsabilités

En parallèle de mes activités de recherche et d'enseignement, je me suis investi dans les différentes responsabilités collectives tant au sein de mon équipe de recherche (XLIM - ASALI), qu'au sein de ma composante de rattachement (la faculté de Droit). Sur le plan scientifique, j'ai participé à plusieurs projets et collaborations industriels. J'ai assuré pendant 8 ans la gestion financière de l'équipe IG du laboratoire XLim-ASALI. Je suis également très investi dans la gestion scientifique globale de XLim-ASALI en raison de ma position de représentant du corps MC au conseil de département SIC (de 2013 à 2016), puis membre du conseil scientifique de l'axe ASALI depuis 2019.

Sur le plan pédagogique, j'effectue des enseignements en Licence et Master au département d'informatique de la faculté de Droit et Sciences Sociales de l'université de Poitiers. Au fil des années, j'ai progressivement pris en charge la responsabilité complète des unités d'enseignements transverses (S2, S3 et S6 de licence Droit et AES). Les effectifs importants de ces formations (900 L1, 600 L2) impliquent une logistique conséquente. Depuis 2019, j'ai pris en charge le pilotage de la commission pédagogique de numérique de l'université de Poitiers. Cette commission a pour rôle d'organiser, harmoniser la mise en place et favoriser les échanges dans le cadre des enseignements transverses (numérique et recherche documentaire) pour toutes les composantes de l'université de Poitiers.

Une présentation détaillée de mes différentes responsabilités est donnée en annexe C.

### 1.4 Structure du document

Ce mémoire d'Habilitation à Diriger des Recherches se poursuit suivant le plan suivant. Le chapitre 2 présente une étude bibliographique des différentes structures de données utilisées pour les applications de rendu photoréaliste par lancé de rayon, ainsi que les différents modèles topologiques proposés dans la littérature pour la modélisation et la construction de scènes complexes. Le chapitre 3 détaille notre approche pour définir l'utilisation d'une partition de l'espace contrainte et convexe (CCSP) comme structure accélératrice pour une application de lancé de rayons interactif. Le chapitre 4 décrit l'utilisation d'une structure topologique et hiérarchique non invasive pour le rendu réaliste des bords arrondis. Le chapitre 5 traite des problématiques de reconstruction de modèles 3D de manière générale, puis plus précisément à partir de donnée IRM dans le but de réaliser des analyses de données pour détecter des tumeurs cérébrales. Enfin, le chapitre 6 établit le bilan général de mes travaux ainsi que mes objectifs scientifiques à moyen et long terme.

---

# CONTEXTE ET POSITIONNEMENT

---

## Sommaire

<b>2.1</b>	<b>Gestion de scènes 3D en rendu réaliste par lancé de rayons</b>	<b>10</b>
<b>2.2</b>	<b>Modélisation à base topologique</b>	<b>17</b>
<b>2.3</b>	<b>Reconstruction de modèles 3D à partir de données</b>	<b>25</b>
<b>2.4</b>	<b>Positionnement des travaux</b>	<b>28</b>

Les environnements virtuels sont produits à l'aide de logiciels de modélisation géométrique adaptés à la création des objets 3D : construction de formes élémentaires, opération d'assemblage pour créer des objets de plus en plus complexes, reconstruction à partir de données numériques. Pour représenter l'environnement, les structures de données comportent des informations géométriques, topologiques et sémantiques. La topologie consiste notamment à connaître les relations de voisinage entre les éléments de différentes dimensions (sommets, arêtes, faces, volumes), permettant ainsi de structurer les objets et proposer des opérations complexes (intersection, opérations booléennes, chanfreinage, etc.). Malheureusement, la richesse des informations disponibles durant l'étape de modélisation n'est que très rarement utilisée par les applications de simulation. Les structures de données sont souvent plus synthétiques, optimisées pour des algorithmes dédiés afin de réduire les temps de traitement et les besoins mémoires. Par exemple, en simulation d'éclairage, la propagation de la lumière est calculée par des méthodes dites "basées rayons" pour respecter fidèlement le système physique et produire ainsi des résultats au plus proche de la réalité. Dans ces applications, l'objectif est de calculer l'éclairage au travers d'une scène 3D virtuelle. Pour cela, le chemin de la lumière est assimilé à un ensemble de rayons traversant la scène jusqu'à atteindre un objet (souvent une face). Finalement, la structure de rendu est construite uniquement à partir des primitives géométriques.

La problématique scientifique abordée dans ce manuscrit traite de l'utilisation d'un même modèle de représentation pour des applications de modélisation et de rendu réaliste. Plus précisément, nous avons étudié comment les informations topologiques construites lors de la phase de modélisation peuvent être utilisées pour proposer des algorithmes de simulation. Dans ce chapitre, nous présentons les différents travaux existants pour identifier le contexte de cette thématique de recherche : les structures accélératrices dédiées aux algorithmes de lancé de rayons, les modèles géométriques à base topologique, la reconstruction de modèles 3D topologiques (en particulier dans le domaine médical).

Dans la suite de ce chapitre, nous étudions tout d'abord les différentes structurations de données proposées dans le cadre de la simulation d'éclairage par lancé de rayons (cf. section 2.1). Nous distinguons les structures classiques qui répartissent les primitives géométriques dans un partitionnement spatial. Celles-ci sont utilisées depuis de nombreuses années et ont été largement étudiées. Nous présentons des structures qui visent à utiliser certaines informations structurelles de l'environnement, les partitions spatiales contraintes, où l'espace est subdivisé en incluant les primitives géométriques du modèle. Nous discutons également de l'impact de la répartition spatiale des données qui peut mettre en défaut certaines structures, par exemple pour la gestion des détails fins comme les arrondis aux bords des objets.

Nous présentons ensuite dans la section 2.2 différents modèles existants dits "*topologiques*" utilisés en modélisation géométrique, dont le principe est de subdiviser l'espace en cellules de différentes dimensions : sommets, arêtes, faces, volumes, etc. Notre objectif étant d'utiliser un modèle pertinent dans le cadre de la représentation d'objets *réels* en 3D, nous expliquons notre choix d'utiliser le modèle de représentation des cartes généralisées. Ce modèle étant fortement mis à contribution dans les travaux présentés dans ce manuscrit, nous présentons les notions et définitions nécessaires à la compréhension des termes.

Pour étudier les différents processus de reconstruction de modèles 3D à partir de données numérisées, nous présentons dans la section 2.3.2 les différentes méthodes de reconstruction de modèles 3D et plus particulièrement dans le contexte du domaine médical. Dans ce secteur, les technologies d'imagerie telles que la tomographie par ordinateur (CT) et l'imagerie par résonance magnétique (IRM) se sont répandues et sont désormais couramment utilisées en médecine pour visualiser les organes internes. Habituellement, les praticiens basent essentiellement leur analyse sur des visualisations 2D, qui correspondent à des coupes des organes. Les applications de reconstruction 3D à partir d'acquisitions médicales se sont développées, car elles fournissent une représentation plus globale et permettent d'effectuer des analyses supplémentaires de l'activité fonctionnelle anatomique.

Enfin, la section 2.4 décrit et motive les projets que j'ai développés dans ce manuscrit.

## 2.1 Gestion de scènes 3D en rendu réaliste par lancé de rayons

Le lancé de rayons est une méthode largement utilisée en synthèse d'image, connue pour sa capacité à simuler la propagation de la lumière pour produire des images de haute qualité. Elle est apparue dans les années 80 avec les approches de rendu basées sur la physique pour les effets d'éclairage. Cette méthode consiste à suivre le parcours de la lumière entre la caméra et les sources de lumière tout en tenant compte des interactions lumineuses entre les objets (réflexion, diffraction, calcul d'ombres – figure 2.1). Elle est maintenant largement utilisée pour la réalisation de films d'animation produits par de grands studios bien connus tels que DreamWorks ou encore Pixar.

Toutefois, le calcul de la trajectoire de la lumière par ces algorithmes de lancé de rayons est reconnu comme chronophage et coûteux en mémoire. En effet, pour converger vers une simulation du parcours de la lumière le plus réaliste, il est nécessaire de multiplier les calculs d'interaction lumineuse entre les objets. Ces traitements correspondent au parcours de milliards de rayons au travers de la scène 3D : rayons primaires issus de la caméra, rayons secondaires d'ombrage, rayons secondaires pour calculer les interférences. Autrement dit, plus le nombre de rayons est important, plus l'image a un rendu photoréaliste satisfaisant. Mais le nombre d'opérations nécessaires au calcul d'une image en 4K d'un film d'animation peut rapidement dépasser plusieurs milliards. De plus, avec l'augmentation des capacités mémoire, il est possible de modéliser des scènes 3D de plus en plus complexes, composées de plusieurs millions de faces. L'efficacité d'une telle technique réside donc principalement dans le calcul rapide du parcours des rayons et de leur interaction avec l'environnement, tout en minimisant l'utilisation mémoire pour gérer de grandes quantités de données.

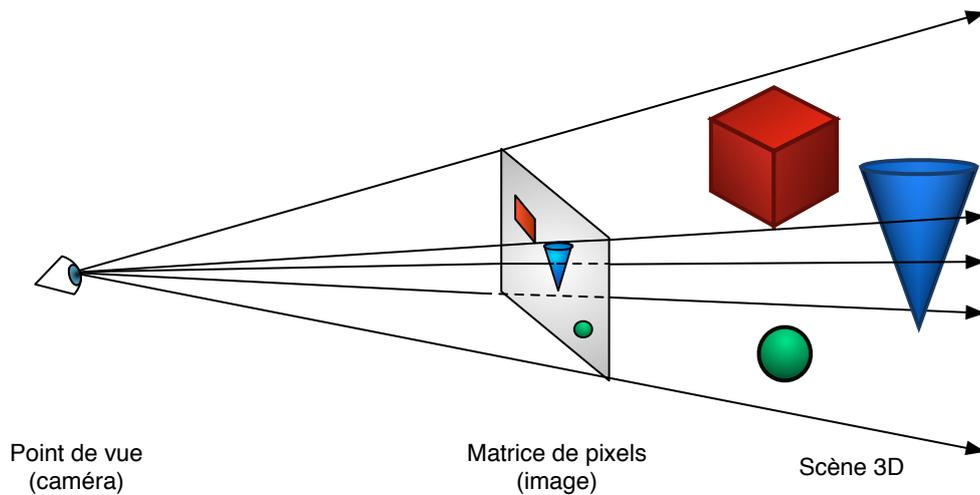


Figure 2.1 – Le lancé de rayons consiste à lancer des rayons depuis le point de vue (la caméra) au travers de la scène 3D. Si un rayon atteint une face, le pixel correspondant est associé à un triplet  $RVB$  dépendant de la réflectance de l'objet. C'est le principe du *ray-casting*.

Aujourd'hui, l'objectif principal reste la qualité visuelle des images photoréalistes (film d'animation, jeu vidéo, réalité virtuelle) et la rapidité de production. Les nouvelles technologies mises à disposition permettent de répondre au besoin en matière de ressources des algorithmes basés sur le lancé de rayons (grande capacité mémoire, ferme de calcul, GPU, etc.) De plus, les architectures parallèles qui se sont fortement démocratisées ces 10 dernières années, sont parfaitement adaptées pour ce genre de processus (cf. figure 2.2). Malgré ces progrès matériels, les exigences concernant le temps de calcul et qualité de rendu restent un sujet de recherche pleinement d'actualité. L'efficacité d'un moteur de rendu par lancé de rayons réside principalement sur l'utilisation de structures de données spatialement organisées et adaptées pour réduire les temps de calcul : on parle de structures accélératrices.



Figure 2.2 – "*Reflections*" : la première démonstration de lancé de rayon en temps réel dans Unreal Engine 4 utilisant le framework DXR de Microsoft et la technologie NVIDIA RTX pour les GPU Volta. La démo *Reflections* illustre les techniques expérimentales d'éclairage et de rendu de nouvelle génération dans Unreal Engine, développées en collaboration avec NVIDIA et ILMxLAB.

### 2.1.1 Structures accélératrices classiques

Les structures accélératrices les plus couramment utilisées sont basées soit sur une partition de l'espace utilisant un schéma de grille, soit sur une organisation hiérarchique de l'espace/des objets, soit par un mixte de ces deux procédés. Ces approches bien connues ont été étudiées en profondeur depuis leur apparition et plusieurs structures permettent d'atteindre l'interactivité en exploitant la cohérence des rayons [113, 137, 162] ou la parallélisation GPU [2, 48, 74, 90, 134].

Nous proposons de classer les structures accélératrices les plus courantes selon le tableau 2.1. Pour éviter toute confusion, nous définissons qu'une partition spatiale correspond à une partition de l'espace euclidien de la scène  $\mathbb{E}^3$ , et qu'une partition objet est une partition de l'ensemble des primitives de la scène  $\partial$ . Nous partons du principe que le lecteur est familiarisé avec les structures accélératrices classiques, pour plus d'informations, il peut se référer aux travaux de V. Havran [80]. D'autres méthodes se basent sur une approche différente notamment en incluant la géométrie de la scène dans la structure accélératrice, nous reviendrons sur ces structures dans la section 2.1.2.

	Type de partition	
	Partition spatiale	Partition objet
Structure linéaire	Grille uniforme [56] Grille non uniforme [79]	Volumes englobants [78]
Structure hiérarchique	BSP-tree [55] Kd-tree [15] Octree [61] Grilles hiérarchiques [79]	BVH [91]

TABLE 2.1 – Classification des structures accélératrices avec quelques exemples des plus répandues

#### Définition 1 (Partition spatiale)

Une partition spatiale  $P = \{R_1, \dots, R_n\}$  subdivise l'espace euclidien  $\mathbb{E}^3$ . Une primitive  $\partial$  représentée dans  $\mathbb{E}^3$  par un ensemble de points  $X \subset \mathbb{E}^3$  est associée à une région  $R$  si et seulement si  $\exists x \in X | x \in R$ .

Une partition spatiale correspond à une partition de l'espace euclidien de la scène  $\mathbb{E}^3$  (cf. Définition 1). Quelques exemples classiques de structures accélératrices basées sur une partition spatiale sont représentés (en 2D) sur la figure 2.3.

La grille uniforme [56] est probablement la structure accélératrice de partitionnement spatial la plus simple. Elle vise à partitionner l'espace en volumes identiques alignés sur des axes. Chaque objet partiellement ou totalement contenu dans un volume est associé à celui-ci. L'algorithme de traversée est un algorithme de type Bresenham, traversant la grille le long du rayon, un volume à la fois. Seuls les objets contenus par les volumes traversés sont testés. Les principaux inconvénients de cette approche apparaissent avec les scènes à densité hétérogène. Pour éviter ce problème, il est possible d'utiliser une grille hiérarchique qui raffine la subdivision en fonction de la distribution de la géométrie (cf. figure 2.3.a). En revanche, son algorithme de traversée est plus coûteux [79].

Les structures d'accélération telles que les Kd-tree [15], les arbres BSP (Binary Space Partitioning [55]), sont basées sur une organisation spatiale et hiérarchique de la géométrie dans un arbre binaire (cf. figure 2.3.b et c). Ces méthodes présentent une complexité de traversée qui dépend de la profondeur de l'arbre. En effet, leur traversée nécessite de commencer par la racine de la hiérarchie. Certains travaux se sont attachés à réduire cette restriction en l'enrichissant par des liens supplémentaires entre les feuilles de l'arbre [133] ou en incluant des niveaux de boîtes englobantes [81]. En revanche, le parcours de ces structures améliorées est plus coûteux puisque l'utilisation de ces extensions nécessite de calculer le point de sortie dans le noeud pour poursuivre la propagation du rayon au travers de la scène 3D.

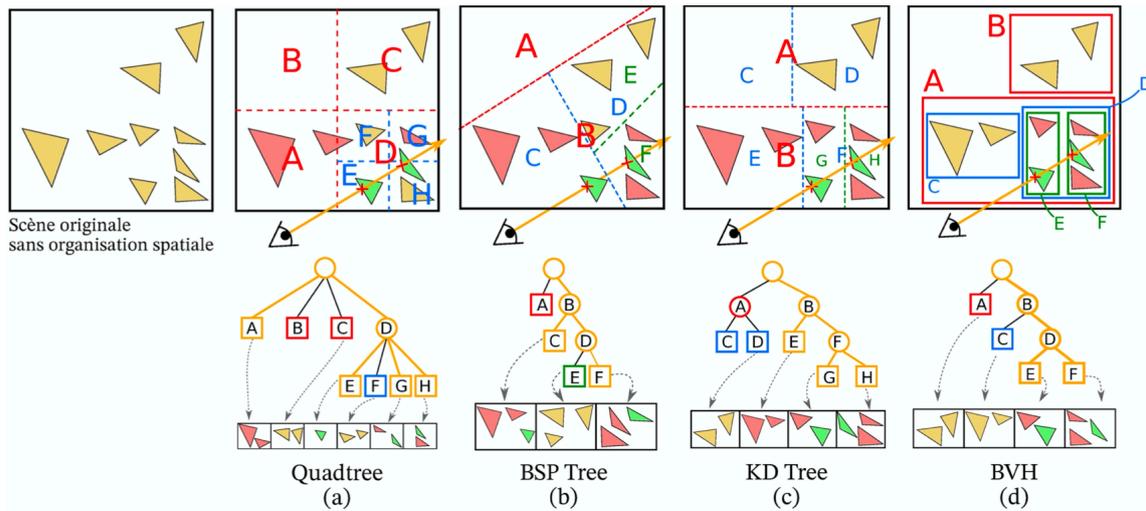


Figure 2.3 – Illustration du parcours d’un rayon au travers de différentes structures accélératrices : (a) Quad-tree (b) BSP tree (c) Kd-tree (d) BVH. En haut, le partitionnement spatial correspondant. En bas, l’organisation de l’arbre avec en orange les noeuds visités.

### Définition 2 (Partition objet)

Une partition objet  $P = \{R_1, \dots, R_n\}$  subdivise l’espace des primitives géométriques de la scène  $\partial^3$ . Un point  $x \in \mathbb{E}^3$  est associé à une région  $R$  si et seulement si  $\exists G \in R | x \in G$ .

Une partition objet correspond à une partition de l’espace des primitives géométriques de la scène  $\partial^3$  (cf. Définition 2). Dans une partition objet, la plus petite région correspond à une unique primitive (un élément de  $\partial^3$ ). Le plus souvent, un volume fermé, dit englobant, est associé à chaque région contenant un ensemble non vide de primitives. Lorsqu’un rayon ne traverse pas un tel volume englobant, il ne traverse aucune primitive.

Un exemple typique de structure accélératrice basée sur une partition objet est la hiérarchie de volumes englobants, le BVH (Bounding Volumes Hierarchy [91, 140]). Le principe consiste à regrouper les primitives en volumes englobants organisés de manière hiérarchique sous la forme d’un arbre. La racine de l’arbre correspond au volume englobant de toute la scène. Chaque noeud interne représente un ensemble (i.e. un volume englobant contenant des primitives), lui même subdivisé par ses fils. Chaque feuille est une région de la partition, associée à une liste (non vide) de primitives entièrement contenues. La figure 2.3.d présente en 2D un exemple de BVH. Le parcours de l’arbre se fait alors de manière récursive depuis sa racine. Si le rayon ne traverse pas le volume d’un noeud donné, alors le sous-arbre associé n’est pas traité.

### 2.1.2 Partitionnement Spatial Contraint CSP

À l’instar d’une partition spatiale classique, une partition spatiale contrainte (CSP) subdivise l’espace euclidien de la scène  $\mathbb{E}^3$  en un ensemble de régions disjointes et non vides mais les primitives géométriques sont obligatoirement confondues avec les bords (i.e. les faces) des volumes composant la partition.

On distingue alors deux types de faces : les faces appartenant à la géométrie de la scène et les faces non occultantes (i.e. qui permettent la propagation de la lumière sans l’interférer) créées pendant la construction de la partition. Fortune a introduit ce concept en proposant un traçage topologique de faisceau utilisant une subdivision convexe acyclique respectant les obstacles de la scène [49].

La recherche de l'intersection la plus proche entre un rayon et la géométrie consiste à parcourir la partition le long du rayon, volume par volume, jusqu'à rencontrer une face appartenant à la géométrie de la scène. Une seule opération est nécessaire : rechercher la face de sortie d'un rayon depuis l'intérieur d'un volume. Si cette face fait partie de la géométrie, alors l'intersection la plus proche est trouvée. Sinon, si la face est non occultante, alors le rayon est propagé dans le volume voisin. La relation de voisinage entre les volumes est utilisée (i.e. relation topologique de dimension 3). La complexité de l'algorithme est donc linéaire en nombre de volumes traversés par le rayon.

La stratégie de subdivision employée a donc logiquement un impact sur l'efficacité du parcours de la CSP, i.e. sur le nombre moyen de volumes traversés par les rayons. L'autre point influençant cette efficacité est l'algorithme de recherche de la face de sortie d'un rayon depuis l'intérieur d'un volume. Afin de faciliter cette recherche, une contrainte supplémentaire peut être ajoutée à la construction de la CSP : les volumes doivent être convexes. Ainsi, un rayon ne peut sortir d'un volume que par une et une seule face. On parle alors de partition spatiale contrainte et convexe (CCSP - Constrained Convex Space Partition).

Lagae propose une CCSP basée sur une tétrahédration qui respecte la géométrie de la scène [96]. Dans ce travail, plusieurs méthodes sont proposées afin de trouver la face de sortie d'un rayon à l'intérieur d'un tétraèdre : utilisation des intersections rayon/plan [60], classification de demi-espace en utilisant les coordonnées de Plücker [131, 147]. L'algorithme se révélant être le plus efficace utilise de 3 à 6 produits scalaires triples. Malheureusement, ces méthodes n'exploitent pas le fait qu'un rayon ne peut pas sortir d'un tétraèdre par sa face d'entrée.

Dans le contexte du rendu volumique, G. Marmitt [117] résout ce problème en exploitant les relations de voisinage entre les faces pour écarter automatiquement la face d'entrée. Le principal inconvénient de cette méthode est le nombre d'opérations nécessaires, qui varie avec les données d'entrée, ce qui entraîne une certaine divergence de calcul dans un environnement parallèle. En effet, la précision arithmétique insuffisante avec des flottants peut provoquer des échecs dans la traversée de la structure conduisant à des boucles infinies.

En comparaison avec les structures accélératrices classiques, les structures accélératrices de type CSP possèdent les propriétés suivantes :

- contrairement aux partitions spatiales, une CSP évite les problèmes de référencement multiple des primitives géométriques ; de plus, il est impossible de tester plusieurs fois la même primitive lors du parcours de la structure accélératrice ;
- contrairement aux partitions objets, il n'y a pas de chevauchement spatial ;
- contrairement aux structures linéaires telle qu'une grille uniforme, une CSP s'adapte parfaitement à la géométrie de la scène ;
- contrairement aux structures hiérarchiques, son parcours est totalement itératif et exploite implicitement la connectivité entre les rayons.

Malgré ces avantages apparents, lorsque nous nous sommes intéressés à ce type de partition, elle était encore peu utilisée dans le cadre des simulations d'éclairage basées rayons. À notre connaissance, lors du lancement de la thèse de M. Maria, il n'existait que deux exemples d'utilisation de CSP dans ce contexte : le graphe cellules et passages et la tétrahédration de Delaunay contrainte.

### **CSP dédiée à l'architecture**

Les environnements architecturaux sont composés de grandes surfaces occultantes telles que des murs, des sols ou des plafonds. Ainsi, à partir d'un point de vue donné, seule une petite partie de la scène est visible. Par conséquent, les structures générales d'accélération souffrent de deux problèmes principaux. D'une part, les scènes meublées contiennent de petits objets très détaillés dans de grands

espaces vides, ce qui ralentit la traversée des structures régulières. D'autre part, étant donné que les murs des bâtiments ne sont pas nécessairement alignés, les structures d'accélération basées sur des plans de fractionnement ne sont pas adaptées; elles ne respectent pas la position des murs, ce qui entraîne un schéma de fractionnement complexe et donc de mauvaises performances.

Le calcul d'images de synthèse pour ces scènes est donc souvent traité à l'aide de structures particulières permettant de localiser les traitements pour des méthodes de simulation d'éclairage [4, 50, 51, 118, 155]. Les auteurs utilisent la notion de "*cells and portals*" (nommée cellules et passages dans la suite de ce chapitre). Pour cela, les cellules sont associées aux pièces du bâtiment et les passages correspondent aux faces incidentes à deux cellules (cf. figures 2.4.b et 2.4.c). Cette représentation correspond à une représentation topologique 3D du bâtiment : des pièces (des volumes 3D) et des relations de voisinage par les ouvertures (des faces 2D reliant les volumes). Le calcul de traversée d'un rayon au travers de la scène utilise les relations de voisinage entre les cellules, limitant de ce fait l'ensemble de volumes potentiellement parcourus. Comme tout autre CSP, pour un rayon émis à partir d'un volume  $V$ , l'ensemble des faces pouvant être atteintes correspond aux faces incidentes à  $V$ . Si la face est un passage, alors le rayon est propagé récursivement dans la cellule suivante, sinon il est stoppé (cf. figure 2.4.d). Par ailleurs, ces informations permettent de construire facilement et directement le PVS<sup>1</sup> (pour *Potentially Visible Set* [157]) de chaque pièce. La consultation de ce graphe de voisinage localisé réduit d'autant plus les temps de traitement.

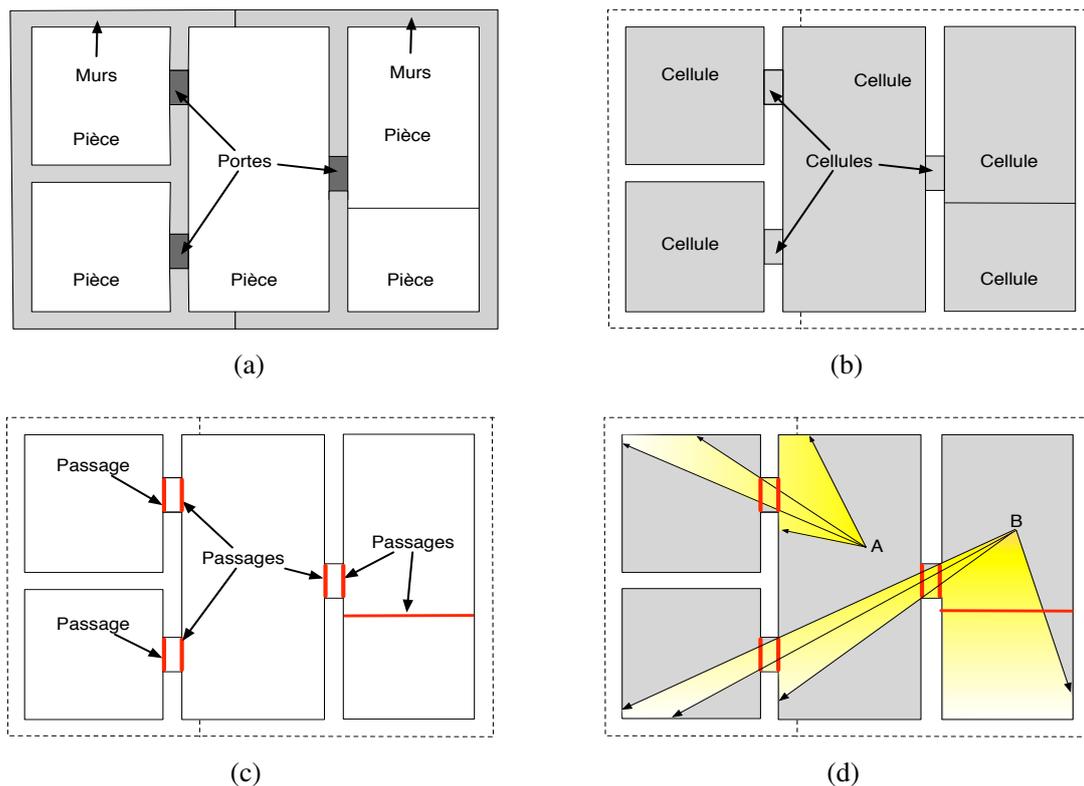


Figure 2.4 – Représentation et utilisation en 2D d'une structure cellules et passages : (a) représentation 2D d'un bâtiment; (b) les volumes de la scène correspondent aux cellules; (c) les faces incidentes à deux cellules correspondent aux passages; (d) propagation des rayons au travers de la scène. Les rayons entrants en collision avec un passage sont propagés dans la cellule voisine, les autres sont stoppés sur le bord des cellules.

1. PVS : ensemble des cellules potentiellement visibles à partir d'une cellule principale.

Dans la littérature, les structures de données type cellules et passages sont construites soit à partir d'un ensemble de polygones 3D représentant une scène [4, 118, 155], soit lors de la construction du bâtiment [50, 52]. Dans le premier cas, les informations topologiques sont déduites par un découpage de l'espace 3D. Les données calculées correspondent uniquement aux relations d'adjacence entre les cellules (i.e. des volumes). Aucune information topologique supplémentaire n'est définie. Dans le deuxième cas, la structure topologique est construite lors de la création du modèle. Cette structure topologique fournit des renseignements supplémentaires : adjacence entre éléments de dimensions 1 et 2 (arêtes et faces), volumes des murs, volume de la façade. Mais bien que le modèle soit plus complet (des cartes généralisées dans ces travaux), seules les relations d'adjacence entre les volumes sont employées lors du processus de rendu [51]. Les informations de dimensions inférieures ne sont pas utilisées pour l'application de simulation d'éclairage proposée.

Pour une scène architecturale 3D, la répartition spatiale des primitives géométriques peut mettre en défaut les structure accélératrices classiques (grandes faces occultantes, les murs et petites faces concentrées, le mobilier). Ce constat peut être généralisé pour des scènes quelconques composées d'objets aux détails fins : objets de très petite taille (fils, grillages, toiles d'araignée), bords arrondis des objets, etc.

### 2.1.3 Gestion de détails fins, les arrondis

Les objets *réels* qui nous entourent ne présentent presque jamais de bords parfaitement nets. Sur les bordures apparaissent des zones plus claires ou plus foncées en fonction des interactions lumineuses avec le reste de la scène. Comme l'illustre la figure 2.5, un élément clé du réalisme concerne la description de ces très petits détails qui ont un impact évident sur l'apparence globale de l'image. Même s'ils correspondent à des effets subpixels [41, 107, 135, 169], leur représentation est essentielle au réalisme, car leur apparence (reflets spéculaires, ombres propres et variations des normales) influence profondément la perception des formes [142]. Il est donc très important d'avoir une représentation géométrique fidèle à l'objet réel pour ces éléments de détails fins, et que le traitement appliqué lors du processus de rendu permette une visualisation photoréaliste.

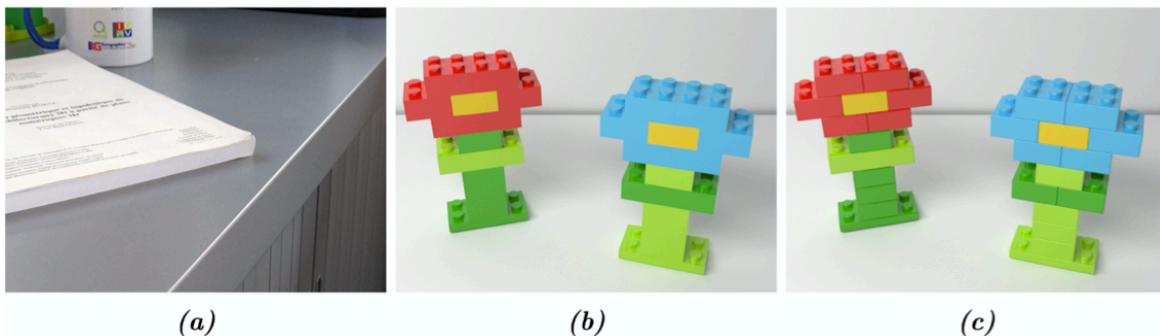


Figure 2.5 – Exemples d'effets lumineux sur des bords arrondis (a) photographie d'un reflet spéculaire sur l'arête chanfreinée d'un meuble métallique (b) et (c) deux objets identiques sans arrondis de bords (b) et avec arrondis (c). Les arrondis permettent de mieux percevoir les délimitations des différentes pièces constituant l'objet.

La modélisation des arrondis repose généralement sur un chanfreinage explicite [34, 153] ou sur des surfaces de subdivision [19, 30, 151], qui génèrent *in fine* des surfaces maillées complexes. Un maillage reste une représentation discrète d'une surface lisse, inévitablement sujette à des artefacts observables lorsque l'observateur et/ou la lumière se déplace(nt) autour de l'objet.

Malheureusement, la gestion de tels détails complexifie la géométrie et augmente les besoins en mémoire et en temps de calcul pour la génération d'une image. Par ailleurs, l'augmentation du nombre de petites faces sur des zones localisées met en défaut la plupart des structures accélératrices classiques. Il devient nécessaire de définir des heuristiques comme le SAH (*Surface Area Heuristic*) pour estimer le coût de recherche d'intersection pour un rayon [62, 111]. En partant du principe que les rayons sont uniformément distribués et qu'ils ne touchent aucune primitive de la scène, l'objectif est de choisir un plan de partitionnement parmi plusieurs possibilités dans le but de minimiser ce coût. Une structure accélératrice est alors considérée comme optimale lorsque son coût SAH est minimal [161].

Pour modéliser un arrondi continu avec une visualisation des effets de lumière plus précise, il est possible d'utiliser des surfaces paramétriques [88]. Ce procédé permet également de conserver une continuité entre les faces de l'objet et le chanfrein (une continuité  $G1$  représente la tangence des courbes ou la continuité de la tangence entre deux courbes). Néanmoins, le coût d'intersection d'une telle représentation est supérieur à un ensemble polygonal fin approchant la même surface. Les éléments paramétriques sont également difficiles à intégrer dans les structures accélératrices et dans le processus de rendu. C'est pourquoi, le rendu des bords et des coins arrondis est souvent traité à l'aide d'une subdivision géométrique ou de shaders de biseau, évitant ainsi le processus d'arrondi et minimisant la complexité de la géométrie du maillage généré [18, 31, 142, 154, 164].

Finalement, les différentes solutions proposées consistent à réaliser un prétraitement au processus de simulation afin d'étudier la répartition spatiale des primitives géométriques de l'environnement (statistique de SAH), l'objectif étant de construire une structure accélératrice adaptée au mieux à cette répartition spatiale. Or, ces informations sont omniprésentes dans le domaine de la modélisation à base topologique qui permet de connaître la structure de l'environnement. Il serait donc judicieux d'utiliser la richesse du modèle topologique plutôt que d'opérer une étape supplémentaire dans le traitement.

## 2.2 Modélisation à base topologique

Dans le domaine de l'informatique graphique, la modélisation géométrique concerne la structuration d'un environnement 3D : la forme géométrique des objets ; les relations de voisinages entre les éléments ; les propriétés (physiques ou visuelles) qui caractérisent les objets. Ces informations sont utilisées pour structurer et manipuler les informations géométriques, appliquer des opérations complexes et améliorer les temps de traitement. Plus particulièrement, la modélisation à base topologique a pour objectif de modéliser des objets géométriques partitionnés en cellules de différentes dimensions (0 pour les sommets, 1 pour les arêtes, 2 pour les faces, 3 pour les volumes, etc.). Les cellules sont liées entre elles par des relations d'adjacence si elles sont de mêmes dimensions, d'incidence si elles sont de dimensions différentes [114]. Cette structuration facilite la manipulation des informations de voisinage entre objets. Elle permet par exemple de directement connaître la liste des faces qui composent d'un volume, ou l'ensemble des arêtes incidentes à un même point. Ces informations sont importantes pour diverses applications telles que la modélisation de bâtiments ou l'étude des sous sols géologiques [25, 52, 99]. Le principe, issu de la représentation par les bords (cf. figure 2.6), est de représenter la topologie (structure) de la subdivision par un modèle combinatoire (par exemple : la structure d'arêtes ailées [12], les cartes combinatoires [42] ou les cartes généralisées [103]).

Dans la suite de cette section, nous décrivons les modèles topologiques de subdivision cellulaire usuels. Notre objectif étant d'utiliser un modèle pertinent dans le cadre de la représentation d'objets réels en 3D, nous expliquons notre choix d'utiliser le modèle de représentation des cartes généralisées et nous présenterons les notions de base à sa manipulation.

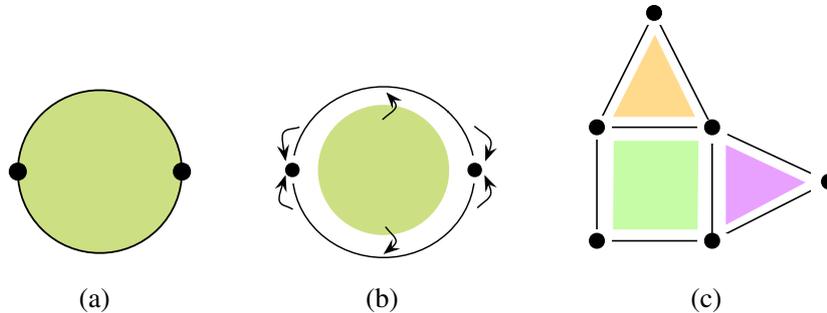


Figure 2.6 – Représentation par les bords : les objets sont décomposés en cellules de différentes dimensions : sommets, arêtes, faces et volumes. Chaque cellule est en relation avec son bord. (a) représentation par les bords d'un disque, la face (dimension 2) est associée à 2 arêtes (dimension 1), elles-mêmes associées à 2 sommets (dimension 0) ; (b) vue éclatée du même disque où les flèches représentent les relations de bord entre les cellules ; (c) modèle cellulaire composé de 3 faces, 8 arêtes, et 6 sommets.

### 2.2.1 Modèles topologiques usuels

De nombreux modèles topologiques et leurs opérations associées ont été proposés afin de représenter et manipuler des objets subdivisés. Historiquement, les premiers travaux ont porté sur la représentation de subdivision de surfaces (variétés de dimension 2), puis sur les complexes (non-variétés) de dimension 2 (cf. figure 2.7). Il existe une grande diversité de modèles permettant de représenter des objets de dimensions quelconques, depuis des objets très généraux (complexes de dimension  $n$ ) jusqu'à des sous-classes de ces derniers (variétés, variétés orientables sans bord, quasi-variétés, etc.).

#### Complexes et variétés

Les modèles de représentation des objets réels sont tous des variétés de dimension 3, ils sont constitués de volumes. Par exemple, un bâtiment est composé de volumes (murs, pièces, sol, etc.). Plus formellement, une variété subdivisée de dimension  $n$  est un assemblage de cellules de dimension  $n$  le long de cellules de dimension  $n - 1$ . En dimension 2, les faces sont assemblées le long de leurs arêtes et une arête est partagée par au plus deux faces.

Un complexe (non-variété) est un objet plus général ne répondant pas au critère précédent. Il peut être construit par assemblage de cellules de dimension quelconque le long de cellules de dimensions inférieures et non de dimension  $n - 1$  seulement. La figure 2.7 illustre les différences entre les assemblages complexes et variétés.

#### Modèles simpliciaux et modèles cellulaires

Les modèles utilisant des cellules régulières (cubiques, simplex, etc.) sont couramment utilisés pour la représentation de maillages : la subdivision n'est pas liée à une sémantique particulière et la régularité de structuration des cellules peut simplifier des calculs complexes (simulation mécanique par exemple). Les différents modèles existants sont distingués suivant le type de cellules composant l'objet. Les modèles simpliciaux sont utilisés pour représenter des objets composés de simplexes (sommets, arêtes, faces triangulaires, volumes tétraédriques, etc.). Les modèles cellulaires permettent de représenter des subdivisions d'objets en cellules quelconques.

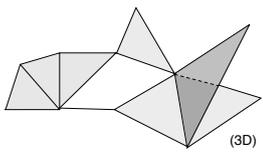
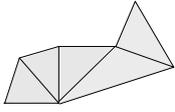
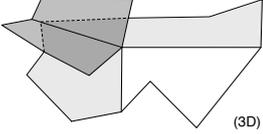
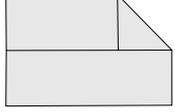
	Assemblages complexes	Assemblages variétés
Cellules régulières		
Cellules "quelconques"		

Figure 2.7 – Classification des subdivisions. Type de cellules : régulières (simplexes, simplexes cubiques ou *quelconques*). Type d'assemblage : quasivariété (caractérisé par des propriétés topologiques comme la multiincidence ou l'orientation) ou complexe (composé de cellules quelconques).

Une partie importante des travaux en modélisation géométrique porte sur la définition des modèles cellulaires afin d'associer une sémantique et un plongement aux cellules. Cette sémantique correspond généralement à des informations supplémentaires comme la nature des objets ou le matériau utilisé pour leur construction. Par exemple, un volume de sémantique *MUR* dans un bâtiment, ou une face aux propriétés photométriques particulières utilisées dans le processus de rendu (couleur, texture, modèles d'apparence, etc.).

### Les graphes d'incidences et les modèles ordonnés

Pour définir les relations entre différents éléments, les graphes d'incidence et les modèles ordonnés sont utilisés. Les graphes d'incidence permettent de modéliser des objets très généraux, mais sans spécifier les relations de multi-incidences. En revanche, les modèles ordonnés ne présentent pas ces inconvénients. Les modèles ordonnés sont basés sur un élément plus "atomique" que les cellules et sur des opérateurs portant sur ces éléments.

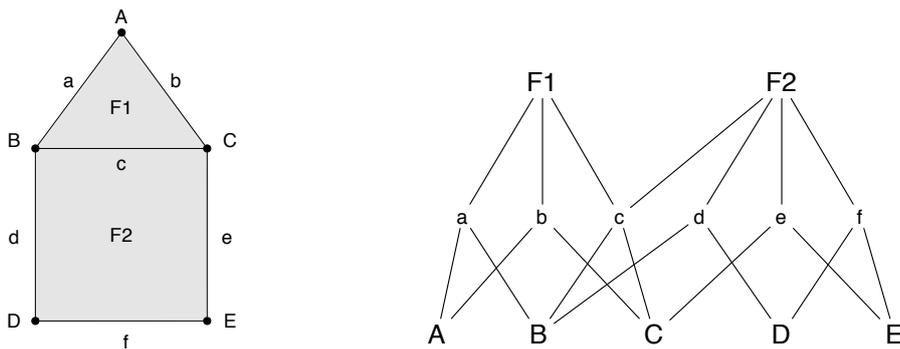


Figure 2.8 – Exemple de graphes d'incidences. Ici le sommet  $A$  est incident aux arêtes  $a$  et  $b$ , l'arête  $c$  est incidente aux faces  $F_1$  et  $F_2$ .

Les graphes d'incidence représentent les différentes cellules d'une subdivision par les noeuds d'un graphe, et les relations de bord entre cellules (relations d'incidence) par les arêtes de ce graphe (cf. figure 2.8). Bien qu'ils permettent d'identifier rapidement les relations d'adjacence et d'incidence entre les cellules, nous pouvons citer certains problèmes fondamentaux liés à la définition même des graphes d'incidence [105, 165] :

- il est difficile de choisir les contraintes de cohérence permettant de garantir la représentation d'un objet valide (un graphe d'incidence peut représenter une arête incidente à trois sommets) ;
- ils ne permettent pas de représenter sans ambiguïté, en particulier la multi-incidence ;
- les relations d'ordre entre cellules (par exemple, l'ordre des arêtes autour d'un sommet) ne sont pas explicitement représentées.

Pour pallier ces différents problèmes, les modèles ordonnés ont été introduits. Ils ont pour avantage principal d'être spécifiés de manière algébrique à l'aide d'un unique type d'élément de base sur lequel des fonctions sont définies. Les modèles ordonnés représentent une relation d'ordre au sein des cellules incidentes. La détection de l'orientabilité de l'objet est facilitée. Les entités de base varient en fonction des modèles. L'un des premiers à avoir été proposé est celui des arêtes ailées [13, 72, 166]. Ce modèle est principalement utilisé pour la représentation de subdivisions de surfaces orientables. Les cartes combinatoires [42] sont définies sur des entités élémentaires appelées *brin*, correspondant intuitivement en 2D à des demi-arêtes. Ceci permet de représenter des subdivisions de surfaces orientables sans bord. Ce modèle a été étendu par Lienhardt [103] en dimension quelconque pour la représentation de variétés orientables ou non, avec ou sans bord. Monsieur Pascal Lienhardt a montré que tous ces modèles sont équivalents, au domaine de modélisation près [104].

De manière informelle, les quasi-variétés constituent une classe d'objets intermédiaires entre les pseudo-variétés et les variétés (cf. figure 2.9) [64]. De manière intuitive, une quasi-variété de dimension  $n$  est construite par assemblage de cellules de dimension  $n$  le long de cellules de dimension  $n - 1$ , de façon à ce qu'une  $(n-1)$ -cellule soit incidente à au plus deux  $n$ -cellules.

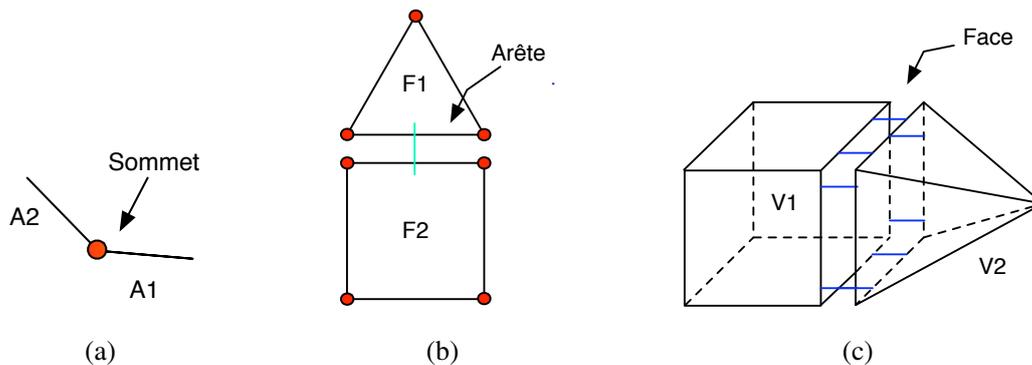


Figure 2.9 – Représentation de quasi-variété : (a) liaison de deux arêtes au niveau d'un sommet ; (b) liaison de deux faces au niveau d'une arête ; (c) liaison de deux volumes au niveau d'une face.

Dans le contexte du rendu de scènes 3D, nous visualisons des objets réels et travaillons dans l'espace  $\mathbb{R}^3$  subdivisé en cellules de dimension 0, 1, 2 et 3 (sommets, arêtes, faces et volumes). Ces modèles représentent des objets orientables 3D (les volumes), délimités par leurs frontières (leurs faces). De plus, dans la mesure où nous souhaitons avoir une représentation volumique, l'intégralité de l'espace doit être modélisé. Ainsi, deux objets ne peuvent pas occuper le même emplacement et l'espace entier est rempli, ce qui conduit à une partition de l'espace. C'est pourquoi le modèle doit être contraint afin de correspondre obligatoirement à une subdivision de l'espace fermée et composée de volumes, de faces et d'arêtes orientées. Par conséquent et topologiquement parlant, une scène doit correspondre à un maillage 3D orienté et fermé.

Plusieurs modèles topologiques ont été proposés dans la littérature pour composer différentes classes de subdivision (surfaces orientées, maillage, etc.) en toutes dimensions. Parmi ces structures nous trouvons : les graphes d'adjacence [139], les modèles ordonnés (comme défini dans [27]), modèle 2D et 3D d'arêtes ailées [12, 166] ou des modèles de dimension supérieure [27, 105].

Les graphes d'adjacence ne permettent pas les multi-incidences (voir [105]) et la validité topologique du modèle est difficile à exprimer [165], c'est pourquoi nous avons choisi un modèle ordonné pour les travaux présentés dans ce manuscrit. Les études ont montré que les maillages 2D et 3D (subdivision de surface et de volume), sont coûteux pour représenter un modèle ordonné tout comme les graphes d'incidence [53]. C'est encore plus vrai lorsqu'il s'agit de géométrie, photométrie et autres attributs pris en compte.

Les modèles définis pour représenter les maillages 3D sont comparables aux cartes 3D (pour les cartes orientables 3D sans bord) ou aux cartes généralisées 3D (pour les maillages orientables ou non et avec ou sans bord) [104]. Les G-cartes offrent une description implicite des cellules (sommets, arêtes, faces, volumes) avec des opérations locales efficaces automatiquement propagées aux cellules incidentes [9, 40]. Elles sont homogènes en toutes dimensions ce qui facilite la définition des opérations en 2D et 3D. L'utilisation d'un tel modèle associé à un plongement linéaire<sup>2</sup> permet de mettre en place facilement des applications de construction et/ou reconstruction d'environnement 3D tout en contrôlant la structuration topologique du modèle.

L'ensemble de ces propriétés nous ont conduit à utiliser ce modèle de représentation pour les différents travaux de modélisation géométrique à base topologique décrits dans ce manuscrit. Nous détaillons dans la section suivante les définitions et notations utilisées dans nos travaux.

### 2.2.2 Cartes généralisées concept et notations

Le modèle des cartes généralisées (ou G-cartes) est une extension des cartes combinatoires [103]. Ce modèle est défini mathématiquement comme des algèbres, à l'instar des algèbres d'arêtes [73] et des structures cell-tuples [27]. Elles permettent de représenter la topologie de quasi-variétés subdivisées orientables ou non, avec ou sans bord [105]. Leur principal avantage est d'être définies de façon homogène en toutes dimensions, contrairement aux cartes combinatoires, ce qui simplifie les définitions et l'écriture des algorithmes. Une carte généralisée de dimension 3 - ou 3-G-carte représente la structure (i.e. la topologie) d'un objet en définissant ses frontières dans chaque dimension : un volume est représenté par ses faces, une face par ses arêtes, et une arête est définie par ses sommets.

À partir d'un élément simple de base (appelé le *brin*) et d'applications bijectives (appelée  $\alpha$ ) définies sur ces brins, les cartes généralisées représentent des cellules et leurs relations d'adjacences et d'incidences. Chaque involution<sup>3</sup>  $\alpha_i$ , avec  $0 \leq i \leq n$  ( $n$  est la dimension de l'espace considéré), représente les relations d'adjacence entre les cellules de dimension  $i$ .  $\alpha_0$  représente les liens entre deux sommets,  $\alpha_1$  lie deux arêtes,  $\alpha_2$  lie deux faces et  $\alpha_3$  lie deux volumes (cf. Définition 3). Dans la suite de ce manuscrit, les relations topologiques en dimension  $i$  sont notées  $\alpha_i$ . La figure 2.10 montre comment cette topologie est visualisée (pour plus de clarté, les sommets liés en dimension 0 pour constituer des arêtes ne sont pas représentés).

2. Représentation vectorielle des éléments, i. e. coordonnées des points selon 3 composantes  $\vec{x}$ ,  $\vec{y}$ ,  $\vec{z}$  représentant chacune une dimension en 3D.

3. Une involution  $f$  est une bijection tel que :  $f^2 = Id$ .

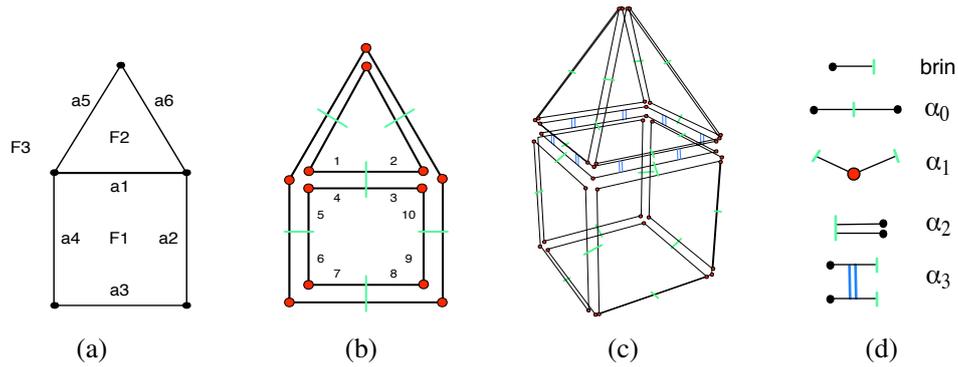


Figure 2.10 – Représentation des G-cartes : (a) un objet 2D composé de 3 faces, 6 arêtes et 5 sommets ; (b) G-carte correspondante : l'ensemble des brins  $\{1,2,3,4\}$  représente l'arête  $a_1$ , l'ensemble des brins  $\{3,4,5,6,7,8,9,10\}$  représente la face  $F_1$  ; (c) objet 3D composé de deux volumes représentés avec les 3-G-cartes, les volumes sont liés par  $\alpha_3$  ; (d) représentation symbolique couramment utilisée des involutions. Dans la plupart des schémas, les liens  $\alpha_0$  ne sont pas représentés pour faciliter la lecture.

### Définition 3 (Cartes Généralisées [105])

Une carte généralisée de dimension  $n \geq 0$  (ou  $n$ -G-carte) est une algèbre  $G = (B, \alpha_0, \dots, \alpha_n)$ , où :

- $B$  est un ensemble fini de brins ;
- $\alpha_0, \dots, \alpha_n$  sont des involutions sur  $B$  ;
- $\alpha_i \alpha_j$  est une involution,  $\forall i, j$  tel que  $0 \leq i < i + 2 \leq j \leq n$ .

Deux brins  $b$  et  $b'$  sont dits  $i$ -cousus si  $\alpha_i(b) = b'$  avec  $b \neq b'$ . Le brin  $b$  est dit  $i$ -libre si  $\alpha_i(b) = b$ . L'orbite associée à un brin  $b$  et à un ensemble d'involutions  $\langle f_1, \dots, f_k \rangle (b)$  est composé de tous les brins obtenus à partir de  $b$  en utilisant toutes compositions d'involutions de l'orbite.

Une  $i$ -cellule est une orbite particulière composée de toutes les involutions à l'exception de  $\alpha_i$ . Deux cellules sont disjointes quand leur intersection est vide, i. e. quand aucun brin n'est partagé par les cellules. Dans la figure 2.11, les  $i$ -cellules qui composent une 3-G-carte sont représentées avec leurs orbites correspondantes.

### Définition 4 (i-cellule)

Soit  $G$  une  $n$ -G-carte,  $b$  un brin de cette carte, et  $i \in \{0, \dots, n\}$ . La  $i$ -cellule incidente à  $b$  est  $\langle \alpha_0, \dots, \alpha_{i-1}, \alpha_{i+1}, \dots, \alpha_n \rangle (b)$ .

Le degré d'une  $i$ -cellule  $C$  est le nombre de  $(i + 1)$ -cellules distinctes incidentes à  $C$  (par exemple, le degré d'un sommet est le nombre d'arêtes qui lui sont incidentes).

### Définition 5 (Éléments pendants)

$\forall i \in [1, 2]$  une  $i$ -cellule  $C_i$  est pendante si et seulement si le degré de  $C$  est 1 et si l'une de ses  $(i-1)$ -cellules incidentes  $C_{i-1}$  est également de degré 1.

Dans la figure 2.12.a,  $A_1$  est une arête pendante. Son degré est 1 et elle est incidente à un sommet de degré 1. Au contraire,  $A_2$  est de degré 1, mais n'est pas pendante, car elle n'est pas incidente à un sommet de degré 1 cf. figure 2.12.c.

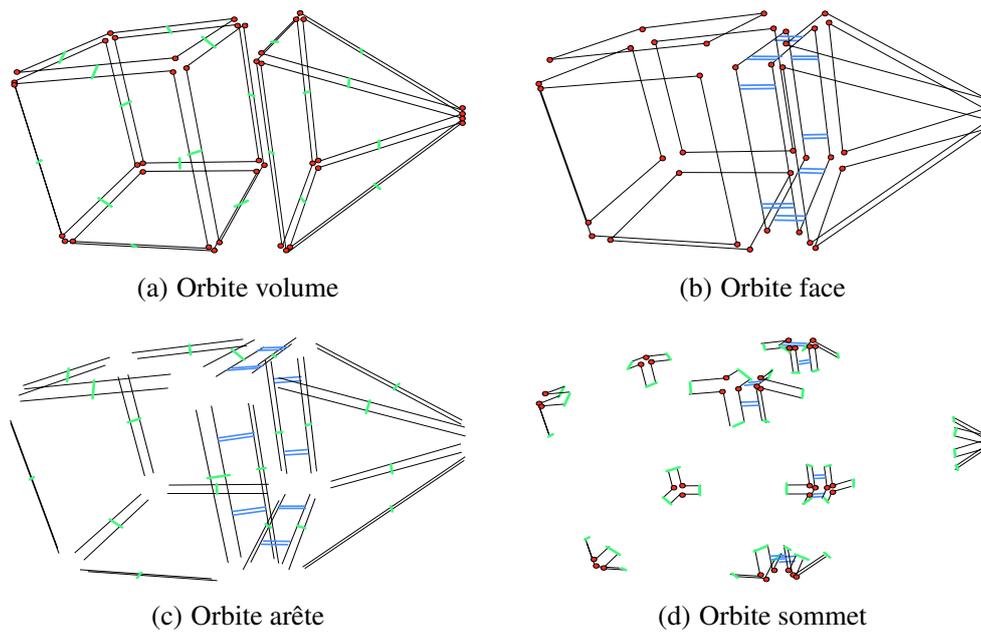


Figure 2.11 – Chaque cellule est obtenue à partir d'un brin et d'une orbite : (a)  $\langle \alpha_0, \alpha_1, \alpha_2 \rangle$  pour les volumes ; (b)  $\langle \alpha_0, \alpha_1, \alpha_3 \rangle$  pour les faces ; (c)  $\langle \alpha_0, \alpha_2, \alpha_3 \rangle$  pour les arêtes ; (d)  $\langle \alpha_1, \alpha_2, \alpha_3 \rangle$  pour les sommets.

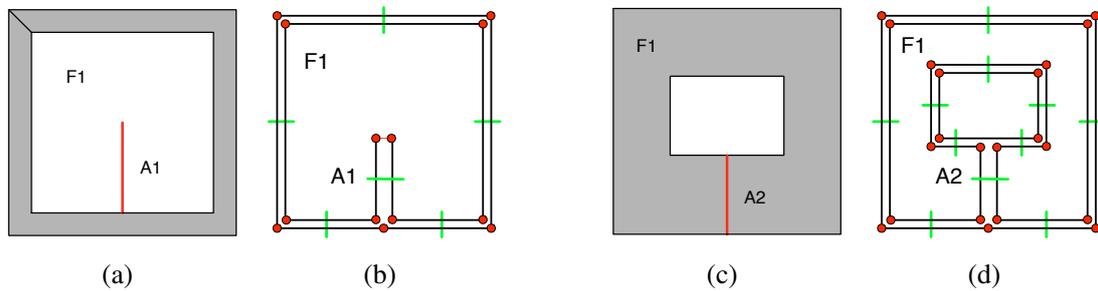


Figure 2.12 – Éléments pendants : (a)  $A_1$  est une arête pendante dans la face  $F_1$ . Son degré est 1 et elle est incidente à un sommet de degré 1 ; (b) représentation de  $A_1$  en terme de G-carte ; (c)  $A_2$  est une arête de degré 1 non-pendante ; (d) représentation de  $A_2$  en terme de G-carte.

### Représentation d'objets réels

Dans la représentation virtuelle d'objets réels, les éléments occupent chacun leur espace sans superposition et correspondent à une subdivision de l'espace en cellules (volumes, faces, arêtes et sommets) définies par leurs bords en toutes dimensions. Les G-cartes permettent de représenter des objets beaucoup plus génériques, notamment orientés et non orientés<sup>4</sup>. Nous devons définir un ensemble de propriétés pour identifier le modèle de représentation d'un objet réel. Nous utilisons notamment une subdivision de l'espace 3D qui nous aide à définir une partition 3D formellement fermée et orientée.

4. Les cellules orientables sont des cellules pour lesquelles nous pouvons définir un parcours en "sens horaire" de manière cohérente

Le critère de partition de l'espace repose sur un mélange de géométrie et de topologie, où nous définissons  $C_i$  comme l'espace géométrique d'une  $i$ -cellule (par exemple,  $C_3$  définit l'espace d'un volume,  $C_2$  définit une face,  $C_1$  une arête,  $C_0$  un sommet). Formellement, un objet 3D est bien formé si et seulement si une partition de l'espace  $P$  satisfait la définition 6. En d'autres termes, deux cellules  $i$  ne doivent pas occuper le même espace et/ou doivent être reliées par des cellules de dimensions inférieures.

**Définition 6 (Partition de l'espace)**

$\forall i \in \{0, 1, 2, 3\}, \forall C_i^1, C_i^2 \in P, C_i^1 \cap C_i^2 = \emptyset \vee \exists j \in [0; i[, C_i^1 \cap C_i^2 = C_j$  où  $C_j \in P$ .

La propriété de fermeture des objets est bien connue, mais avec des 3-G-map, cette propriété diffère et devient une fermeture topologique définie par la définition 7. De manière moins formelle, cela signifie qu'il n'y a pas d'auto-boucle dans le graphe et que les faces ouvertes sont interdites.

**Définition 7 (Fermeture)**

$\forall b \in G, \forall i \in [0; 3], b.\alpha_i \neq b$ .

La dernière propriété est l'orientation : Les G-cartes peuvent représenter des objets orientés (ou non). Pour représenter l'orientation, nous incluons un booléen (souvent nommé *marque d'orientation*). Formellement, la propriété d'orientation est définie dans la définition 8 où deux brins adjacents ont une valeur opposée sur cette marque d'orientation pour toute dimension.

**Définition 8 (Orientation)**

$\forall b \in G, \forall i \in [0; 3], b.\text{orient} \neq b.\alpha_i.\text{orient}$ .

**Mise en oeuvre**

À partir d'un noyau basé sur les G-cartes, le modelleur topologique 3D *Moka* a été développé au laboratoire XLim. Ce modelleur comprend plusieurs opérations comme la couture de deux cellules le long d'une face, ou d'autres applications plus complexes comme l'extrusion ou le corafinement [159]. *Moka* a été fortement mis à contribution dans les divers travaux présentés dans ce manuscrit (cf. figure 2.13).

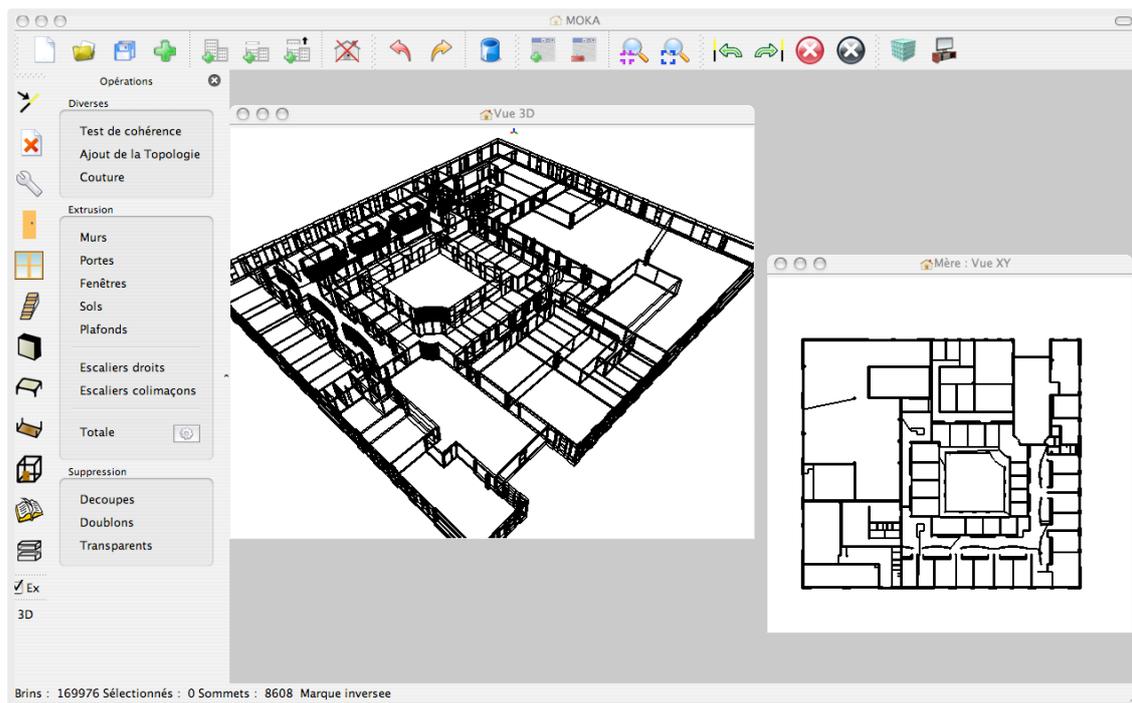


Figure 2.13 – Interface du module de Moka [159] dédié à la modélisation de bâtiments.

## 2.3 Reconstruction de modèles 3D à partir de données

La section précédente décrit les différents modèles de représentation utilisés en modélisation géométrique à base topologique. Lorsque les environnements 3D sont construits directement à l'aide des modélisateurs géométriques par un assemblage d'éléments géométriques de base, la structuration topologique de la scène est directement définie au fil des opérations de construction. Il est donc facile de contrôler les cohérences topologiques du processus de construction. Mais dans divers domaines d'application, il est nécessaire de produire une scène virtuelle à partir de données numériques existantes : nuage de points issus de la numérisation d'un bâtiment, plan numérique d'architecte, sondage sismique pour la reconstruction de sous sol géologique, imagerie médicale pour la modélisation des organes internes, etc. La principale difficulté des méthodes de reconstruction réside dans la génération de modèles virtuels identiques (ou au plus proche) géométriquement de la donnée d'origine (réelle). Il est par ailleurs nécessaire de procéder à un calcul spécifique pour reconstruire les informations topologiques et sémantiques. Les résultats de ces posts-traitements sont fortement dépendants des données d'entrée qui sont généralement très bruitées.

### 2.3.1 Reconstruction de maillages

Basée sur des recherches antérieures [16, 17, 70, 120], une large étude des méthodes de reconstruction et de réparation de modèles est proposée dans l'ouvrage rédigé par M. Botsch & al. [23]. Les auteurs distinguent deux grandes catégories de méthodes : les approches volumiques et les approches surfaciques. Les approches volumétriques consistent à recalculer un nouveau maillage valide après transformation du maillage existant en un maillage intermédiaire. Ces méthodes sont souvent très chronophages et leurs performances reposent sur la structure intermédiaire choisie (variété orientée [70], arbre de partition d'espace binaire [120], polyèdres de Nef [75]). Les algorithmes sont robustes et produisent des modèles

cohérents. Mais la phase de reconstruction entraîne une déconnexion avec les données de départ et une perte des informations sémantiques difficiles à recalculer [23, 149]. Les approches surfaciques visent à modifier directement le maillage d'entrée pour réparer localement les intersections *une par une* [16, 149]. Ainsi, les modifications sont limitées aux parties incohérentes. Cependant, la résolution de la géométrie d'intersection est numériquement instable, ce qui peut produire des artefacts (faces non planes, arêtes nulles, etc.) et la reconstruction ne garantit pas la génération d'un modèle volumique.

Les procédés mentionnés ci-dessus ne peuvent pas respecter à la fois les propriétés de robustesse, de précision et d'efficacité. Des post-traitements sont proposés pour supprimer efficacement les incohérences simples (arêtes complexes ou sommets singuliers), mais ils ne sont pas entièrement automatiques [70, 120]. Finalement, seules les contraintes géométriques sont contrôlées, les notions topologiques de voisinages sont rarement utilisées et vérifiées par un modèle de subdivision formel.

### 2.3.2 Reconstruction à partir de données médicales

Dans le contexte de l'imagerie médicale, la reconstruction de surfaces 3D à partir de données joue un rôle important dans la production de modèles virtuels anatomiques. Le champ d'application de tels modèles est vaste : diagnostic, visualisation et planification de thérapies, biopsie virtuelle, formation, etc. Les technologies d'acquisition d'imagerie médicale, telles que la tomographie assistée par ordinateur (CT) et l'imagerie par résonance magnétique (IRM), fournissent des images de coupes transversales en 2D qui sont espacées uniformément sur des tranches individuelles. Ces données permettent la reconstruction d'un modèle 3D discret (on parle de modèle digital) qui est assimilé à une grille uniforme.

Cette grille contient des informations sur un espace 3D, et est composée d'un certain nombre de voxels dont la taille correspond à la résolution spatiale de l'acquisition. Pour identifier les différents éléments anatomiques - *les substances* - il est courant d'utiliser une méthode de segmentation permettant répartir les voxels de la grille dans des *segments*, c'est à dire des régions d'intérêt (ROI pour Region Of Interest) correspondant aux différentes substances. Les méthodes de reconstruction visent ensuite à construire une surface 3D correspondant aux limites d'une ROI [123]. Le maillage surfacique créé s'adapte à la limite d'un segment de la segmentation, mais pour répondre mieux aux besoins cliniques, il est nécessaire d'avoir une représentation volumique et de connaître les relations de voisinage entre les éléments.

De nombreuses méthodes de reconstruction 3D ont été proposées, qui peuvent être regroupées en plusieurs catégories. Les modèles déformables ont été largement utilisés dans les images médicales. Il existe essentiellement deux types de modèles déformables : les modèles paramétriques et les modèles géométriques. L'utilisation de modèles paramétriques [33, 110] génère un maillage initial continu qui est ensuite transformé géométriquement pour correspondre aux frontières de la ROI. La précision obtenue est inférieure au voxel initial, mais le temps de calcul nécessaire est souvent important notamment pour éviter les auto-intersections. L'utilisation des modèles géométriques offre des solutions pour pallier ces limites. Ils sont basés sur la théorie de l'évolution des courbes [95] et des méthodes de *level set* [26, 94, 108], représentant les courbes et les surfaces de manière implicite. Ces modèles peuvent facilement empêcher l'auto-intersection [76, 172]. Cependant, ils ne peuvent être utilisés que dans le cas d'une topologie initialement connue (par exemple le nombre de volumes à reconstruire), ce qui limite leur application physiologique [170]. En particulier, les méthodes de *level set* sont utilisées pour la segmentation d'images médicales dans la reconstruction d'un volume unique de tumeur [92, 93]. Néanmoins, le volume reste discrétisé si une représentation de surface est nécessaire, un algorithme de construction d'isosurface est requis.

Historiquement, les techniques de reconstruction ont été introduites avec le célèbre algorithme d'extraction d'isosurfaces *Marching Cubes* (MC) [106]. La simplicité, la robustesse et la rapidité du MC en ont fait l'algorithme le plus utilisé pour la visualisation interactive de données médicales. Cependant,

un problème bien connu du MC réside dans la génération d'un maillage continu (sans discontinuité topologique). L'association à des données volumiques est ainsi compromise. De nombreuses méthodes avec décideur asymptotique ont été proposées pour traiter ce problème [38, 68, 121]. En outre, de nouvelles approches sont apparues, telles que Dual Marching Cubes [122] qui remplace les triangles par des quadrilatères tendant à éliminer les triangles mal formés existants dans les surfaces MC. La continuité n'étant toujours pas entièrement résolue, des versions plus élaborées ont été proposées [69]. Malgré ces versions récentes, le MC ne fournit qu'un maillage de surface de la ROI, chaque surface produite est lissée indépendamment, ce qui peut conduire à une perte de connectivité ou à des superpositions lorsque plusieurs segments sont reconstruits et regroupés dans le même espace. Il est donc indispensable de gérer un maillage volumique contraint sur les relations de voisinages des volumes.

Outre les évolutions proposées du MC, de nombreuses techniques de reconstruction de surface basées sur les contours ont été développées, et peuvent être classées en deux grandes catégories : l'assemblage de contours qui est basé soit sur une représentation explicite de la surface [10, 22, 57, 59], soit sur une représentation volumétrique qui utilise des représentations implicites [26]. L'inconvénient de ces méthodes est qu'elles ne traitent pas de cas particuliers tels que les cavités. La deuxième classe de méthodes peut encore être subdivisée en deux classes. Les méthodes d'interpolation dans lesquelles les contours transversaux sont exprimés par des champs de distance 2D lisses et continus [58, 87, 100, 116, 173]. Les méthodes d'approximation qui tendent à ajuster une fonction implicite aux données d'entrée en utilisant des normales fournies telles que les modèles implicites MPU (Multi-level Partition of Unity) proposés par Braude et *et al.* [26].

Plus récemment, les recherches sur la reconstruction 3D à partir d'images médicales se sont concentrées sur la croissance et l'amélioration des approches d'apprentissage profond et de réseaux neuronaux, avec des méthodes implicites et explicites. Les premières utilisent un réseau de neurones profond (DNN - Deep Neural Network) pour apprendre une représentation implicite de la surface [37, 82, 124], puis trianguler le maillage final à l'aide d'une méthode d'extraction des isosurfaces. Les méthodes explicites font appel à un DNN pour déformer un maillage initial vers une cible [98, 109], afin de produire directement un maillage explicite. Les approches d'apprentissage profond sont limitées dans leurs applications, car elles doivent s'entraîner sur une topologie connue. Par exemple, le réseau de neurones utilisé pour des méthodes de reconstruction corticale est entraîné à reproduire des mailles homéomorphes à une sphère, ce qui est une hypothèse erronée en présence d'une tumeur. Des recherches supplémentaires ont été menées afin de raffiner le maillage produit par les MC avec des réseaux de neurones. Ces approches restent limitées, car la phase d'apprentissage doit s'appuyer sur une topologie fixe [32, 102]. De plus, chaque maillage volumique produit est lissé indépendamment, ce qui conduit à une perte de connectivité ou à des superpositions lorsque plusieurs volumes sont reconstruits.

Par ailleurs, de nouvelles méthodes de reconstruction sont apparues, prenant en compte la connectivité et la topologie globale pour reconstruire des données avec des segments multiples. Elles ne sont pas limitées à la reconstruction corticale, comme les approches d'apprentissage profond [37, 82]. Ce type de méthodes de reconstruction permet de contrôler la topologie de la surface reconstruite en définissant la propriété exacte à respecter au cours du processus. Les méthodes de reconstruction à partir de nuages de points sont conçues pour produire, à partir d'une entrée partielle ou complète, un maillage multisegment qui nécessite de connaître la topologie résultante avant le traitement [97]. Sans utiliser de telles contraintes, Bruel et al. ont développé une méthode basée sur l'homologie persistante pour raffiner la fonction implicite à partir de laquelle la surface finale sera extraite [28]. Cette approche peut garantir le fonctionnement avec des modèles multisegment uniquement s'ils partagent les mêmes caractéristiques topologiques. Des tentatives récentes ont été faites pour effectuer une reconstruction de surface à partir d'une segmentation multilabels qui préserve la connectivité entre chaque élément. Basé sur le double contourage, le méthode proposée récemment par Frisken [54] conserve des frontières nettes pour produire des mailles lisses sans discontinuités, en préservant la topologie de la segmentation. Cependant, il n'applique aucun type de correction, et ne prend pas en compte les connaissances anatomiques.

## 2.4 Positionnement des travaux

Les travaux présentés dans cette habilitation à diriger des recherches visent à mettre à contribution les structures topologiques et hiérarchiques pour améliorer les opérations de reconstruction 3D et optimiser les algorithmes de simulation d'éclairage par lancé de rayons.

Mon projet de recherche s'articule autour de trois axes principaux. Tout d'abord, l'utilisation d'une structure topologique basée sur les cartes généralisées comme structure accélératrice pour un algorithme de lancé de rayons dans le contexte de la visualisation de scènes architecturales. Puis nous avons proposé une nouvelle structure hiérarchique et topologique permettant une meilleure gestion des détails fins tels que les arrondis dans le rendu de scènes 3D. Enfin, toujours dans l'idée d'utiliser un modèle homogène pour les phases de modélisation et de simulation, nous avons étudié les processus de reconstruction 3D avec pour objectif de proposer un processus basé sur la reconstruction automatique de la topologie à partir d'un modèle quelconque. Ce travail expérimental a ensuite été utilisé dans une application de reconstruction de modèles 3D à partir d'images IRM. L'objectif est de produire un modèle 3D continu complet du cortex cérébral directement utilisable pour des calculs de simulations physiologiques permettant de détecter la présence de tumeurs.

### 2.4.1 Structure topologique pour la simulation d'éclairage

Dans un premier axe de recherche, nous avons étudié l'utilisation des informations topologiques issues de la phase de modélisation comme structure accélératrice pour une méthode de lancé de rayons. Pour cela, nous avons exploré une voie peu étudiée jusqu'alors, en utilisant le modèle de représentation issu de la phase de modélisation de scènes architecturales pour produire directement une partition de l'espace contrainte et convexe (CCSP) (cf. figure 2.14). Cette CCSP est ensuite directement utilisée comme structure accélératrice lors du processus de rendu. Ce travail a été réalisé au cours de la thèse de Maxime Maria, l'idée principale consistant à placer toute la géométrie sur les frontières des volumes convexes vides composant une telle partition. En d'autres termes, la structure d'accélération est fusionnée avec la géométrie. Cette structure ressemble à la tétraédrisation contrainte de Lagae [96], mais en utilisant des volumes convexes généraux. Ce type de partition se distingue des structures classiques par plusieurs concepts apportant des propriétés uniques et intéressantes. Notre CCSP est composé d'un plus petit nombre de volumes généraux, mais convexes. De plus, la CCSP est directement déduite de l'étape de modélisation. Et surtout, nous accélérons l'analyse des CCSP en utilisant des informations topologiques de toutes les dimensions (de 0 à 3). Ce processus nous a permis de développer une application de rendu interactif de scènes architecturales complexes (plusieurs centaines de millions de polygones).

Nous utilisons ces résultats pour généraliser cette approche à des scènes quelconques. Nous nous concentrons notamment sur l'utilisation d'une tétraédrisation de Delaunay contrainte comme structure accélératrice et proposons un nouvel algorithme de parcours. Comme la méthode proposée par G. Marmitt [117], notre méthode exploite les relations de voisinage entre les faces. L'originalité est la manière spécifique de représenter un tétraèdre qui permet de fixer à strictement 2 le nombre d'opérations nécessaires au calcul de sa traversée. Ce nouvel algorithme est naturellement plus efficace puisqu'il nécessite moins d'opérations arithmétiques que les précédents. De plus, il est plus adapté au calcul parallèle puisqu'il ne pose pas de problème de divergence d'exécution supplémentaire.

Nos travaux concernant le rendu de scènes architecturales ont mis en évidence l'intérêt de partitionner l'espace et de hiérarchiser son contenu notamment pour gérer des primitives géométriques de tailles très hétérogènes (grandes faces de murs, petit mobilier). Nous nous sommes donc intéressés dans un deuxième axe de recherche au rendu réaliste de détails fins. Dans ce contexte, nous avons proposé la thèse de Simon Courtin dont le sujet concerne le rendu des arrondis sur les bords des objets. Contrairement aux surfaces maillées qui discrétisent la courbure des arrondis, notre objectif est de garantir une apparence

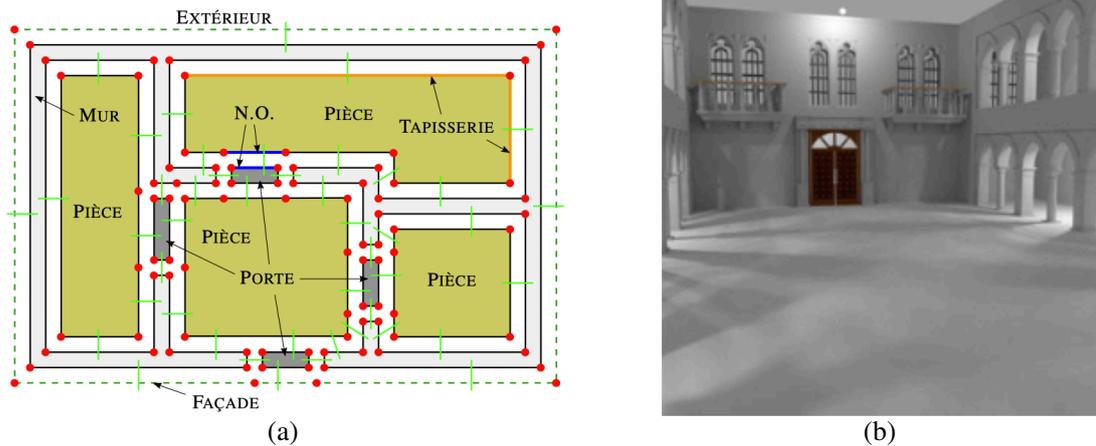


Figure 2.14 – (a) Modèle topologique basé sur les G-Cartes, (b) image de résultat.

arrondie et lisse, quel que soit le point de vue et la distance de l'observateur, ainsi que l'éclairage incident. Au cours de ces recherches nous avons proposé une méthode non invasive (c'est-à-dire qui contrairement aux méthodes de subdivision ne modifie pas la géométrie d'origine) pour la modélisation et le rendu de chanfreins. La figure 2.15 illustre quelques exemples de configurations d'arrondis d'arêtes et de sommets quelconques.

Au coeur de notre contribution se trouve une structure hiérarchique et topologique qui définit automatiquement et avec précision la géométrie des zones arrondies des bords et des sommets. Cette structure, appelée SREC-RT (pour Structure for Ray Tracing Rounded Edges and Corners), est construite automatiquement et est intégrée dans une structure d'accélération dédiée au lancé de rayons qui permet de déterminer la région d'intérêt de chaque chanfrein et d'y appliquer le traitement correspondant. Elle permet de visualiser les arrondis des bords et sommets sans augmenter la complexité géométrique de la scène 3D. L'intersection des rayons avec les surfaces arrondies est accélérée grâce aux liens topologiques définis par les arêtes et les sommets. SREC-RT est compatible avec tout type de représentation de géométrie arrondie. Nous proposons une solution avec des surfaces paramétriques. Notre méthode a été intégrée dans le modèleur *Blender* [20] avec les moteurs de rendu *Cycles* [21] et *pbrt* [127].

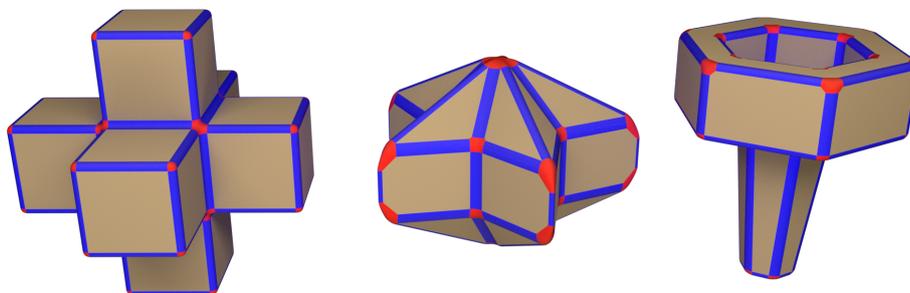


Figure 2.15 – Exemples de configurations avec différents cas d'arrondis gérés par notre méthode. Les arrondis d'arêtes convexes et concaves (en bleu) et d'angles (en rouge).

### 2.4.2 Reconstruction topologique : application en imagerie médicale

Les différentes études que nous avons menées concernant l'utilisation de structures topologiques pour des algorithmes de simulation nous ont démontré que la meilleure stratégie consiste à construire directement la structure au cours de la phase de modélisation. Les objets 3D peuvent être entièrement produits par des opérations de construction, mais plus généralement, il est nécessaire de construire une structure à partir de données existantes (un ensemble de polygones désordonnés). C'est pourquoi, nous nous sommes intéressés au processus de reconstruction d'informations topologiques à partir de données géométriques. La problématique principale de ce type de processus réside dans le calcul automatique de l'ensemble des relations de voisinages afin de reconstruire un ensemble de volumes sans superposition spatiale à partir d'une soupe de polygones non contraints. Nous proposons un algorithme général pour reconstruire des objets 3D avec des informations topologiques 3D et sans incohérence géométrique. Tout d'abord, une subdivision cellulaire est effectuée en s'appuyant sur des données d'entrée géométriques, puis les relations topologiques entre les cellules sont explicitement définies. Le modèle des cartes généralisées est utilisé pour représenter la topologie de la scène 3D. Dans un second temps, les faces sont scindées le long de leurs intersections afin de satisfaire une contrainte de partition d'espace pour construire un ensemble de volumes sans se superposer. Une de nos contributions notables consiste à résoudre le problème d'intersection 3D en utilisant des informations topologiques pour améliorer les performances en vitesse et gérer l'imprécision géométrique.

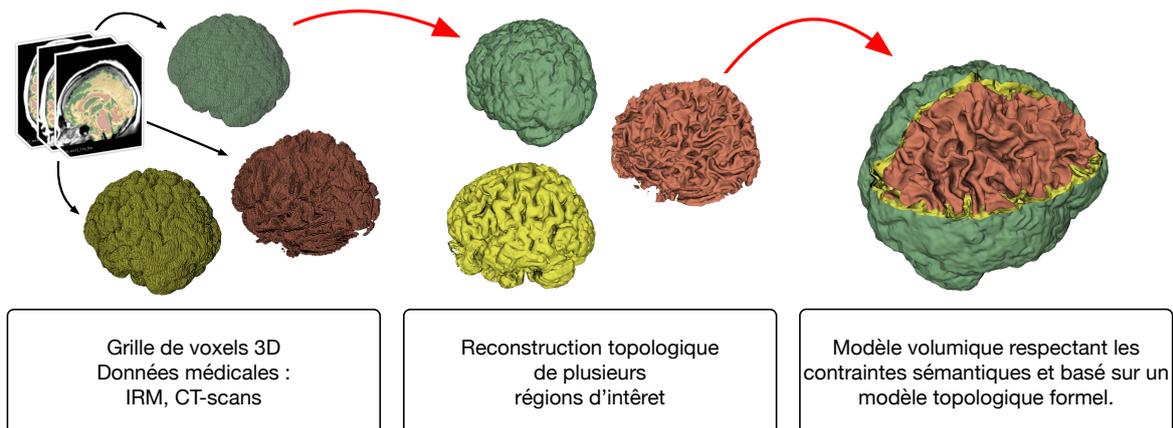


Figure 2.16 – Processus de reconstruction topologique de modèle 3D à partir de données médicales.

Nous avons ensuite intégré cet algorithme de reconstruction topologique dans un processus pour la reconstruction de maillage volumétrique à partir d'acquisitions médicales 3D discrètes telles que des IRM (cf. figure 2.16). Ce processus permet la reconstruction automatique d'objets composés de plusieurs substances anatomiques, afin d'assurer une visualisation globale, unique et cohérente d'un cortex cérébral. Notre processus donne, sans aucune intervention de l'utilisateur, un modèle 3D topologique robuste et propre avec des relations d'adjacence explicites entre les volumes reconstruits. Le modèle résultant respecte les données sémantiques, sans les défauts topologiques et géométriques habituellement rencontrés dans les autres approches de reconstruction 3D. Il est ainsi directement utilisable pour des calculs d'estimation des données métaboliques/physiologiques au sein des différentes parties du cerveau dans le cadre d'outils d'aide à la détection des tumeurs.

---

# PARTITION TOPOLOGIQUE CONTRAINTE POUR LE LANCÉ DE RAYONS

---

## Sommaire

---

<b>3.1</b>	<b>Modèle topologique dédié aux bâtiments : CCSP</b>	<b>32</b>
<b>3.2</b>	<b>Utilisation de la CCSP pour le rendu réaliste</b>	<b>35</b>
<b>3.3</b>	<b>Résultats et discussion</b>	<b>41</b>
<b>3.4</b>	<b>Visualisation topologique : généralisation</b>	<b>45</b>
<b>3.5</b>	<b>Conclusion</b>	<b>50</b>

---

Nous avons vu dans le chapitre 2 le fonctionnement d'un algorithme de lancé de rayon. Cette technique est utilisée pour simuler le parcours des ondes lumineuses en les assimilant à des rayons qui traversent la scène 3D. Dans les environnements dits complexes (i.e. composés de plusieurs millions de polygones), la mise en place de structures accélératrices (grille uniformes, Octree, Kd-tree, etc.) est indispensable pour réduire les temps de calcul des applications de simulations.

L'utilisation de structures accélératrices pour les applications de lancé de rayons est un domaine très largement étudié depuis plusieurs décennies (cf. section 2.1.1). Le principe généralement appliqué consiste à travailler à partir d'un ensemble de polygones pour construire la structure la plus adaptée à la scène 3D en cours de visualisation. Finalement, la structure obtenue est créée de manière complètement ad-hoc sans utiliser les informations disponibles dans les applications de modélisation. L'étape de rendu est ainsi décorrélée de l'étape de construction de l'environnement or des informations utilisées dans la modélisation seraient très intéressantes à prendre en compte pour guider le processus de rendu. C'est le cas par exemple des relations de voisinage entre les éléments, les informations topologiques utilisées dans les modeleurs géométriques.

Pour ce premier axe de recherche, nous avons souhaité approfondir l'utilisation d'informations topologiques dans le cadre des méthodes de simulation basées rayons. Pour cela, nous proposons un modèle topologique formel représentant une partition spatiale contrainte et convexe (CCSP Constrained Convex Space Partition). Une CCSP est un graphe représentant une partition spatiale pour laquelle la géométrie de la scène est située à la limite des volumes convexes (cf. section 2.1.2). Par conséquent, un volume est composé de deux types de faces : les faces occultantes, qui appartiennent à la géométrie de la scène et sont associées à un matériau ou une texture spécifique, et les faces non occultantes (notées  $N.O$ ), qui n'existent que pour construire une partition convexe. Cette structure de données correspond à un graphe  $G = \langle V, E \rangle$ ,  $V$  étant un ensemble de volumes convexes vides et  $E$  l'ensemble des faces non occultantes partagées par deux volumes. Le lancé de rayon est effectué en parcourant ce graphe, un volume à la fois, jusqu'à ce qu'il rencontre une face occultante.

Les travaux présentés dans ce chapitre ont été réalisés au cours de la thèse de Maxime Maria où l'idée d'une CCSP est appliquée aux scènes architecturales en suivant le modèle topologique dédié aux environnements architecturaux proposés par Horna et al. dans [84]. Le modèle associé fournit un ensemble d'opérations topologiques et sémantiques permettant de créer une scène 3D de bâtiment et de générer directement la CCSP correspondante (cf. section 3.1). Après avoir expliqué de manière détaillée comment les informations topologiques sont utilisées pour guider le lancé de rayons (cf. section 3.2), nous montrons que la CCSP fonctionne pour différentes stratégies de lancé de rayons ; elle permet de lancer plusieurs centaines de millions de rayons par seconde, spatialement cohérents ou non, pour de grands bâtiments meublés composés de centaines de millions de polygones et contenant des milliers de sources lumineuses (cf. section 3.3). Enfin, dans la section 3.4, nous proposons une généralisation de l'utilisation de la CCSP pour la visualisation d'objets quelconques (i.e. des scènes non-architecturales).

### 3.1 Modèle topologique dédié aux bâtiments : CCSP

Les travaux présentés dans ce chapitre se basent sur le modèle topologique dédié aux environnements architecturaux proposé au cours de ma thèse de doctorat [83, 84]. Ce modèle que nous appelons *GMapArchi* étend les cartes généralisées (cf. section 2.2.2) par un ensemble de contraintes topologiques, géométriques et sémantiques. La principale contribution de ce travail consiste à définir un ensemble de contraintes de cohérences permettant de contrôler la structure d'une scène architecturale 3D. Ainsi, le modèle correspond à une subdivision de l'espace 3D orientée et fermée composée de volumes d'une épaisseur significative et disjoints deux à deux. L'ensemble de l'espace est modélisé : les zones de « vide » (e.g. les pièces ou l'extérieur) correspondent aussi à des volumes. Chaque volume est identifié par une sémantique en fonction de sa nature (un plongement sur les cellules de dimension 3), telle que PIÈCE, MUR, PLAFOND, etc. L'ensemble de ces contraintes de cohérence, formellement défini à partir du modèle de cartes généralisées, permet de définir le modèle *GMapArchi* suivant :

#### Définition 9 ( $n$ -*GMapArchi*)

Une  $n$ -*G*-carte  $G = (B, \alpha_0, \dots, \alpha_n)$  est une  $n$ -*GMapArchi* si :

- $G$  représente une partition :  $\forall C_i^1$  et  $C_i^2$  deux  $i$ -cellules de  $G$  avec  $i \in \{0, 1, 2, 3\}$  ;  $C_i^1 \cap C_i^2 = \emptyset$  ou  $C_i^1 \cap C_i^2 = \{C_j \mid j \in \{0, 1, 2\} \text{ et } j < i\}$  ;  $C_j \in P$  ;  $C_j$  est incidente à  $C_i^1$  et à  $C_i^2$
- $G$  est fermée (i.e.  $G$  ne contient aucun brin libre et deux  $i$ -cellules adjacentes ont obligatoirement une  $(i-1)$ -cellule en commun)  $\forall b \in G \alpha_i(b) \neq b, i \in \{0, \dots, n\}$
- $G$  ne représente qu'une seule composante connexe :  $\forall b \in B, B = (\alpha_0, \alpha_1, \alpha_2, \alpha_3)(b)$
- $G$  ne contient pas d'éléments pendants  $\forall b \in B, \alpha_1(b) \neq \alpha_2(b)$  et  $\alpha_1(b) \neq \alpha_2(b)$
- Chaque volume est associé à une et une seule sémantique :  $\text{sémantique}(b) = \text{sémantique}(\alpha_0, \alpha_1 \alpha_2)(b)$

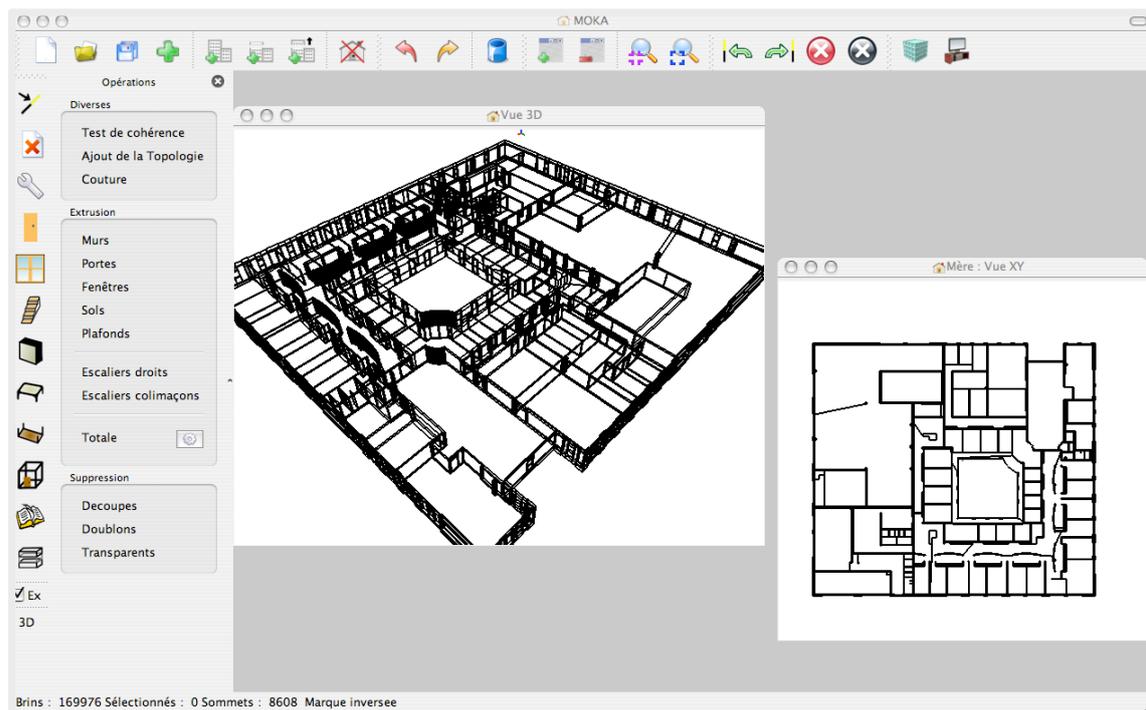


Figure 3.1 – Présentation de l’interface du module de Moka [159] dédié à la modélisation de bâtiments. Communément appelé *MokaArchi*.

Une application de modélisation architecturale, développée au sein du département ASALI du laboratoire XLIM, implémente ce modèle en se basant sur le noyau de manipulation de G-cartes Moka [159]. Cette application, que nous nommons *MokaArchi*, propose un ensemble d’opérations de modélisation basée sur le modèle GMapArchi. Elle permet la génération semi-automatique d’un bâtiment en 3D à partir de son plan d’architecte 2D [83] (cf. figure 3.1). Elle a été employée pour modéliser l’ensemble des scènes architecturales utilisées pour les expérimentations de la thèse de Maxime Maria. La génération se déroule en deux étapes. Dans un premier temps, la topologie 2D du plan est reconstruite en respectant les contraintes du modèle. Dans un second temps, le bâtiment 3D est construit par extrusion en ajoutant le sol et le plafond. Le résultat contient toutes les informations nécessaires à la génération du CCSP : il représente une partition d’espace orientée et fermée respectant parfaitement la géométrie, contient toutes les informations topologiques nécessaires, et identifie la nature de tous les volumes (et celle des faces par le voisinage des volumes).

Le processus de reconstruction produit des volumes composés d’un couple de faces horizontales inférieures et supérieures, et d’un ensemble de faces verticales (murs, ouvertures, *etc.*). Chaque face est également identifiée par une sémantique : les faces occultantes, dont la sémantique fait référence à un matériau ou à une texture spécifique (TAPISSERIE, SOL, PLAFOND, *etc.*); les faces non occultantes (N.O.) correspondent aux ouvertures (par exemple, une face incidente à un volume de pièce et un volume de porte). La figure 3.2 illustre la représentation topologique et sémantique d’un bâtiment en 2D respectant ces propriétés.

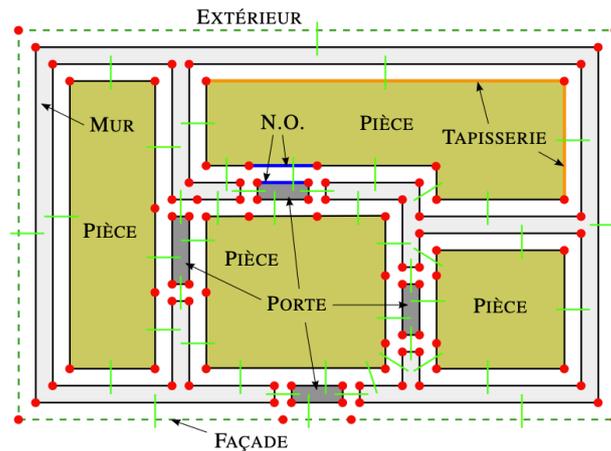


Figure 3.2 – Représentation topologique d'un bâtiment 2D par une GMapArchi de dimension 2. Dans cette partition les faces et les arêtes sont identifiées par une sémantique distincte (volumes et faces en 3D).

### 3.1.1 Construction des volumes convexes

Par définition, les volumes composant une CCSP sont convexes (cf. section 2.1.2). Cependant, dans le modèle architectural utilisé, il n'y a a priori aucune raison pour que les volumes respectent cette contrainte. Pour respecter les contrainte d'une CCSP, il est donc nécessaire de subdiviser chaque volume concave en volumes convexes. Pour simplifier le problème, la décomposition en volumes convexes est effectuée en 2D, avant l'extrusion du modèle vers la 3D. Le polygone concave correspondant à la face inférieure du volume est subdivisé en régions convexes en utilisant la méthode présentée par Fernández et al. [44]. Les arêtes de subdivision créées sont par la suite extrudées verticalement. Les faces résultantes sont associées à une sémantique *N.O.* car elle n'appartiennent pas à la géométrie initiale de la scène et doivent donc être traversées par les rayons sans les dévier.

Cette méthode permet une décomposition convexe de polygones à la fois robuste et efficace. Cependant, elle peut générer de nombreux nouveaux polygones, notamment en présence de murs concaves ou de piliers (cf. figure 3.3.a). Un rayon traversant une telle zone va parcourir un grand nombre de volumes, réduisant ainsi les performances des simulations. Pour diminuer l'impact de ce type de configuration, nous restreignons l'espace occupé par la sur-découpe en ajoutant préalablement des plans virtuels autour des courbes concaves (cf. figure 3.3.b). S'il en résulte un plus grand nombre de volumes, la probabilité qu'un rayon traverse un nombre de volumes important est réduite.

### 3.1.2 Correspondance de GMapArchi à une CCSP

Le modèle GMapArchi décrit entièrement la CCSP et pourrait être directement utilisé pour le lancé de rayons. La CCSP est déduite en balayant la GMapArchi. Chaque volume devient un noeud de la CCSP : il correspond à un sous-graphe décrivant la topologie du volume, c'est-à-dire qu'un volume est décrit par ses faces reliées par  $\alpha_2$ , et une face est décrite par ses arêtes reliées par  $\alpha_0\alpha_1$ . Les sommets ne sont pas explicitement représentés, puisqu'ils sont inutiles lors du lancé de rayons. Lors de la conversion, chaque face ayant une sémantique *N.O.* devient une arête reliant le volume courant à un voisin (connu grâce à la relation  $\alpha_3$ ). Selon leur sémantique, les autres faces sont associées à un matériau spécifique.



Figure 3.3 – Convexification d’une pièce aux parois courbes : (a) convexification basée sur la méthode présentée par Fernández et al. [44]; (b) application d’un prétraitement permettant une première découpe autour des éléments courbes.

## 3.2 Utilisation de la CCSP pour le rendu réaliste

Cette section présente en détail l’utilisation de notre CCSP et des informations topologiques qu’elle contient pour calculer le parcours des rayons dans la scène 3D. Ce procédé permet de gérer le calcul de l’éclairage à partir de sources lumineuses ponctuelles. Pour évaluer la robustesse de notre structure, notamment son comportement en cas d’utilisation d’un grand nombre de rayons incohérents à travers des scènes architecturales meublées complexes, nous proposons un traitement spécifique permettant d’inclure du mobilier dans la scène.

### 3.2.1 Traversée de la CCSP

La recherche de la plus proche intersection entre un rayon et la géométrie de la scène s’effectue en deux étapes principales. Tout d’abord, la localisation du volume contenant l’origine du rayon. Ensuite, le rayon traverse la partition en parcourant un volume à la fois jusqu’à ce qu’il rencontre une face occultante.

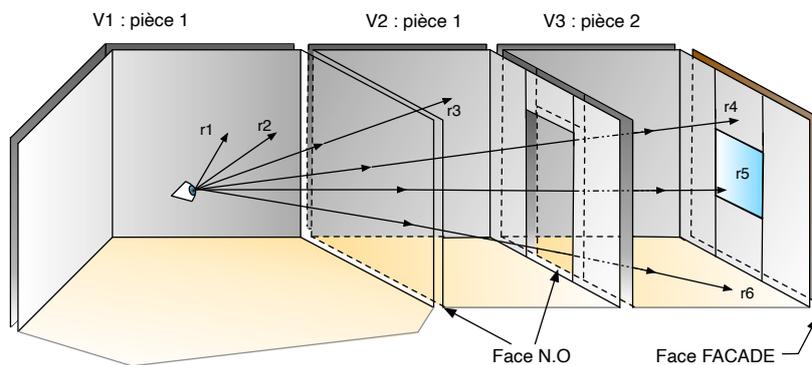


Figure 3.4 – Traversée de la CCSP : tous les rayons primaires partent du même volume ( $V_1$ ) contenant la caméra ; ils sont propagés s’ils touchent une face non occultante (N.O.) ou arrêtés sinon. Par exemple, le rayon  $r_1$  frappe directement une face de  $V_1$  ayant la sémantique WALLPAPER, tandis que  $r_4$  passe par deux faces N.O. avant d’être propagé dans  $V_3$ .

Traverser la structure et tester les intersections de rayons aboutit à la même opération : déterminer la face de sortie d'un rayon à l'intérieur d'un volume donné. Les volumes étant convexes, la face de sortie est unique. Si cette face est occultante, alors le rayon est arrêté et l'intersection est retournée. Si cette face est non occultante, le rayon se propage à travers le volume voisin, connu par  $\alpha_3$  (information topologique de dimension 3). La figure 3.4 illustre le parcours de plusieurs rayons au travers d'une scène 3D. L'algorithme de traversée a une complexité temporelle qui est linéaire dans le nombre de volumes traversés par le rayon.

Par sa simplicité, cet algorithme cache en fait deux points essentiels. Le premier concerne la recherche du volume de départ du rayon. Le second problème, le plus important, réside dans la recherche de la face de sortie d'un rayon à l'intérieur d'un volume.

### Localisation du point d'origine des rayons

Dans un moteur de lancé de rayons, nous distinguons deux types de rayons distincts. Les rayons primaires issus de la caméra vers la scène 3D, et les rayons secondaires qui permettent de calculer les interactions lumineuses entre les objets (ombrage, réflexion, réfraction, etc.). Pour les rayons primaires, en supposant que la caméra se trouve dans le domaine de partition, plusieurs solutions sont envisageables pour localiser le volume contenant l'origine de la caméra (et donc l'origine de tous les rayons primaires). Une méthode naïve consiste à tester un à un tous les volumes de la partition. Cette méthode peut être accélérée en utilisant une grille de subdivision ou un Kd tree. Dans tous les cas, la localisation de l'origine du rayon dépend du nombre global de volumes composant la partition. Dans le cas d'un modèle de caméra avec des origines de rayon perturbées ou lorsque la caméra est déplacée, le mouvement correspond à une translation. Ainsi, une variante de l'algorithme de traversée CCSP résout efficacement ce problème.

Finalement, pour une application interactive, le temps de calcul pour la localisation du volume d'origine reste négligeable, d'autant plus que ce calcul n'est réalisé que pour la première image. Par ailleurs, que ce soit sur le CPU ou le GPU, les expériences montrent que les performances ne sont pas affectées lorsque la caméra est déplacée.

Pour les rayons non primaires, la localisation de l'origine du rayon est évitée en exploitant la connectivité implicite des rayons à l'intérieur de la CCSP. En effet, le point de départ et le volume correspondent tous deux à l'arrivée des rayons précédents. Comme indiqué dans la section 2.1.1, les structures d'accélération hiérarchiques nécessitent de recommencer la traversée depuis la racine, et l'exploitation de la connectivité des rayons reste une tâche difficile. Au contraire, à l'aide d'une CCSP, une fois l'origine des rayons primaires déterminée, le calcul d'un chemin de lumière ne consiste qu'à rechercher sa face de sortie à l'intérieur d'un volume.

### Recherche de la face de sortie

Les volumes qui composent la CCSP sont convexes, un rayon traverse au maximum une face orientée d'un volume donné. L'approche proposée est divisée en deux étapes (cf. Algorithme 3.1). Le volume est considéré comme un polyèdre infini, en ignorant les faces supérieures et inférieures. La première étape détermine la section infinie de sortie (ligne 2-5). Cette recherche balaie les arêtes incidentes à la face inférieure en utilisant les liaisons topologiques  $\alpha_0$  et  $\alpha_1$ . Ainsi, les arêtes de la face inférieures sont balayées, chacune correspondant à une section verticale infinie délimitée par deux lignes orientées (cf. figure 3.5.a). Ces arêtes sont balayées linéairement, compte tenu du faible nombre de sections par volume, un balayage plus complexe (comme la recherche binaire) n'est pas nécessaire.

**Algorithme 3.1** Recherche de la face de sortie pour un rayon dans un volume

---

**ENTRÉE:**  $V$  : volume ;  $r$  : rayon ;

- 1: {1<sup>re</sup> étape : section infinie de sortie}
- 2:  $i \leftarrow 0$  ;
- 3: **Tant que** (  $\text{side}(r, V.\text{sec}[i].\text{left}) \leq 0$   
 $\parallel \text{side}(r, V.\text{sec}[i].\text{right}) > 0$  ) **faire**
- 4:  $i \leftarrow i + 1$  ;
- 5: **Fin Tant que**
- 6: {2<sup>me</sup> étape : face de sortie de la section infinie  $i$ }
- 7:  $f \leftarrow V.\text{sec}[i].\text{faces}$  ;
- 8: **Si**  $\text{side}(r, f[0].\text{line}) < 0$  **alors**
- 9: **return**  $f[0]$  ; {traverse le sol}
- 10: **Fin si**
- 11: **pour**  $k = 1$  to  $V.\text{sec}[i].\text{nb\_faces1}$  **faire**
- 12: **Si**  $\text{side}(r, f[k].\text{line}) < 0$  **alors**
- 13: **return**  $f[k]$  ;
- 14: **Fin si**
- 15: **fin pour**
- 16: {Sortie par le plafond}
- 17: **return**  $V.\text{plafond}$  ;

---

La deuxième et dernière étape de l'algorithme consiste à trouver la face de sortie parmi celles qui composent la section verticale infinie de sortie (incluant le sol et le plafond). Chacune de ces faces est délimitée par deux lignes orientées de droite à gauche (direction trigonométrique depuis l'intérieur du volume). Encore une fois, une recherche non linéaire n'est pas nécessaire grâce au faible nombre de faces composant une section verticale. Ainsi, cette recherche se fait linéairement de bas en haut en utilisant les relations topologiques de dimension 2 qui connectent les faces, les  $\alpha_2$ , (cf. figure 3.5.b). La face de sortie est trouvée lorsque le rayon passe sous les lignes de délimitation supérieures.

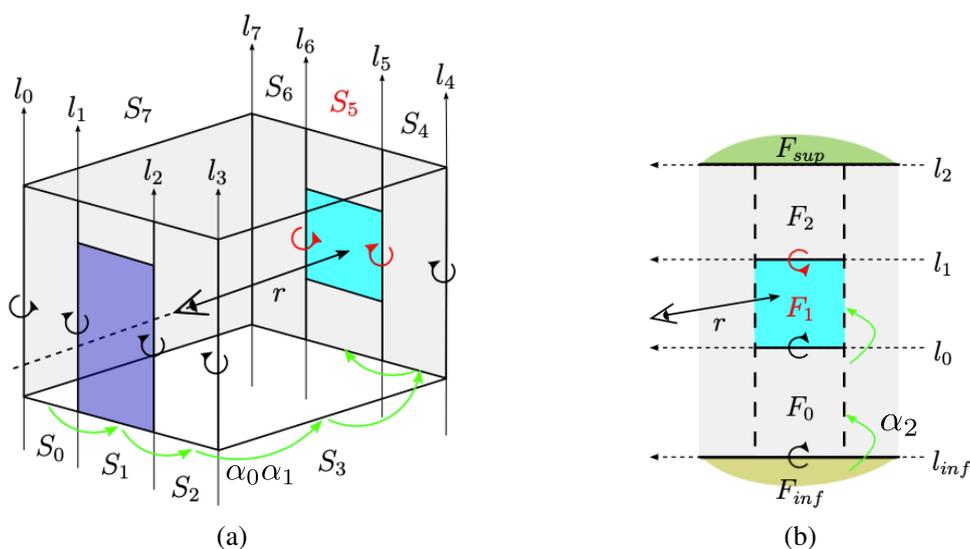


Figure 3.5 – Recherche de la face de sortie d'un volume pour un rayon donné (a) recherche de la section verticale infinie contenant la face de sortie ; (b) Recherche de la face de sortie à l'intérieur de la section.

En résumé, le parcours de la structure est optimisé en utilisant les relations topologiques dans toutes les dimensions :

- $\alpha_0$  et  $\alpha_1$  pour balayer les arêtes incidentes à la face inférieure d'un volume et trouver la section infinie de sortie ;
- $\alpha_2$  pour balayer les faces à l'intérieur d'une section infinie et déterminer la face de sortie ;
- $\alpha_3$  pour propager un rayon d'un volume à son voisin lors de la traversée d'une face non occultante.

Pour réduire le nombre d'opérations nécessaire au calcul de la face de sortie, nous utilisons l'espace de Plücker, présenté par Julius Plücker au 19<sup>ème</sup> siècle [132]. Les coordonnées de Plücker sont une paramétrisation analytique des droites de l'espace tridimensionnel dans un espace projectif  $\mathbb{P}^5$ , appelé espace des droites ou encore espace de Plücker par abus de langage. Elles procurent un cadre analytique pour réaliser des calculs sur des droites et des ensembles de droites.

L'espace de Plücker est couramment utilisé en informatique graphique, car il permet notamment de spécifier facilement la position de vecteurs les uns par rapport aux autres dans l'espace 3D. Par exemple, dans [112, 156], cet espace est utilisé pour définir les intersections entre les rayons et des polygones convexes ou des boîtes englobantes d'objets. Nous souhaitons utiliser la paramétrisation de Plücker pour définir la position relative de deux droites dans l'espace 3D via l'opérateur *side* (cf. figure 3.6). Cet opérateur est défini comme le produit interne permuté des deux lignes de coordonnées de Plücker [147]. Les lignes délimitant une section infinie sont orientées de bas en haut. En utilisant les coordonnées de Plücker, le test d'intersection rayon/section est très rapide puisqu'il ne nécessite que deux tests d'orientation : un rayon passe par une section si et seulement s'il passe par la ligne de délimitation droite (*resp.* la gauche) ou gauche (*resp.* la droite) (cf. figure 3.5.a).

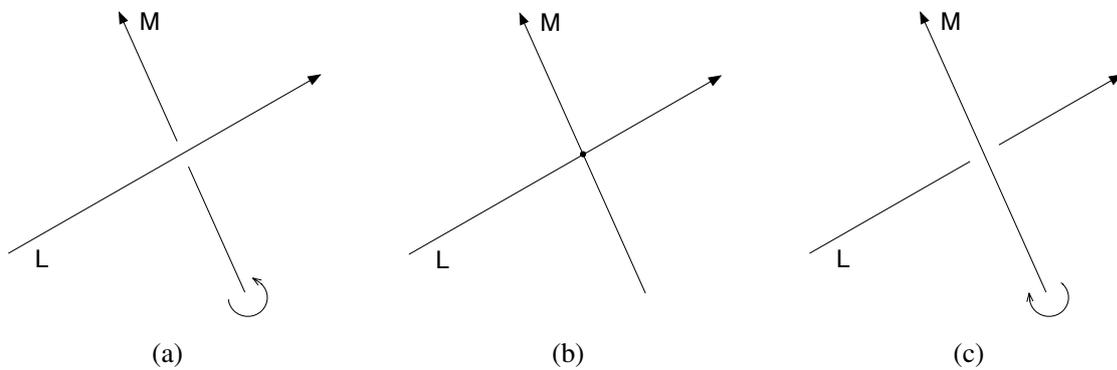


Figure 3.6 – Calcul de l'orientation relative de deux droites à partir de l'opérateur *side* et de leurs coordonnées de Plücker : (a)  $\Pi_L \odot \Pi_M > 0$ ; (b)  $\Pi_L \odot \Pi_M = 0$ ; (c)  $\Pi_L \odot \Pi_M < 0$ .

Pour un cube simple composé de 6 faces, une méthode naïve effectuerait au pire 24 opérations *side* ( $4 \times 6$  faces) pour détecter la face de sortie d'un rayon donné. Avec notre stratégie, le pire cas est atteint en effectuant 10 opérations *side* ( $2 \times 4$  sections infinies + 2 lignes). L'ordonnancement de nos calculs, guidés par les relations topologiques contenu dans la CCSP, permet de réduire fortement le nombre d'opérations Plücker *side* nécessaire au calcul de la face de sortie.

### 3.2.2 Utilisation étendue de la structure CCSP

La gestion de rayons secondaires avec notre structure d'accélération est un problème simple. Chaque rayon suit le même algorithme de traversée (cf. section 3.2.1). Le coût de la traversée d'un rayon secondaire est identique au lancé d'un rayon primaire. La seule différence concerne le volume de départ qui est directement connu grâce à la connectivité implicite des rayons. À partir de ce constat, nous pouvons utiliser notre structure pour calculer l'éclairage direct à partir de sources lumineuses ponctuelles.

Nous proposons également d'ajouter du mobilier dans les scènes 3D. Pour cela, nous utilisons notre CCSP avec un niveau de hiérarchie pour gérer l'inclusion des objets. L'objectif principal de ces extensions est d'évaluer la robustesse de notre structure d'accélération, notamment son comportement en cas d'utilisation d'un grand nombre de rayons incohérents à travers des scènes architecturales meublées complexes.

#### Eclairage direct

Une approche naïve de l'éclairage direct implique de tracer autant de rayons d'ombre que de sources lumineuses ponctuelles dans la scène. Pour les environnements architecturaux, une telle approche conduit à des taux de rafraîchissement médiocres étant donné le nombre important de sources lumineuses. Néanmoins, étant donné qu'un bâtiment est composé de grandes surfaces occultantes, la plupart des tests de visibilité effectués sont inutiles. Pour éviter ce problème, l'ensemble potentiellement visible (PVS) des sources lumineuses de chaque volume est calculé [157].

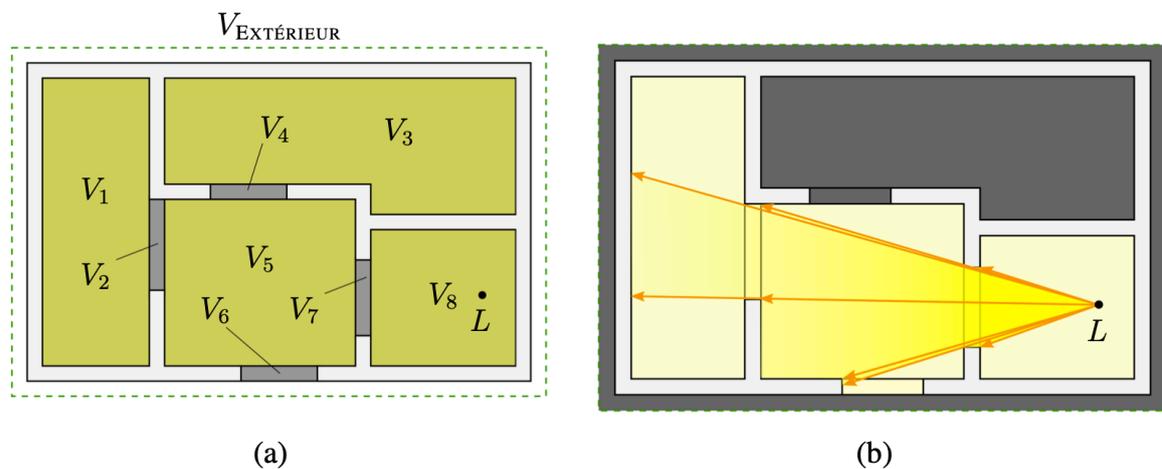


Figure 3.7 – Calcul de visibilité et zones visibles (PVS) : (a) un bâtiment composé de neuf volumes et d'une source lumineuse L; (b) calcul du PVS : la source lumineuse est ajoutée au PVS des volumes traversés par le faisceau issu de L; le PVS des volumes  $V_3$  et  $V_4$  ne contient pas L.

Le PVS des sources lumineuses doit être calculé pour chaque volume de la partition spatiale. Pour cette étape, nous avons résolu le calcul du PVS en inversant le problème pour réduire le temps de calcul. Ainsi, pour une source lumineuse donnée, un faisceau continu est projeté à travers chaque ouverture du volume qui la contient; puis, ce faisceau est propagé dans la scène pour ajouter la source lumineuse au PVS de chaque volume traversé (cf. figure 3.7). Cette amélioration permet de connaître directement l'ensemble des sources lumineuses visible pour chacune des pièces.

### 3.2.3 Gestion du mobilier

Pour ajouter du mobilier dans les scènes architecturales, une méthode d'ameublement automatique a été ajoutée à *MokaArchi*. Les objets sont assimilés à leur boîte englobante et sont placés de manière cohérente dans les pièces. Par exemple : les étagères sont placées le long des murs, les chaises autour des tables, etc. La boîte englobante de l'objet est ajoutée dans le modèle comme un volume à part entière avec une sémantique distincte (OBJECT).

Pour respecter la contrainte de convexité, le volume initial (contenant la boîte englobante de l'objet) est subdivisé. Les volumes résultants conservent leur sémantique originale et sont séparés par des faces non occultantes pour produire la CCSP correspondante (cf. figure 3.8).

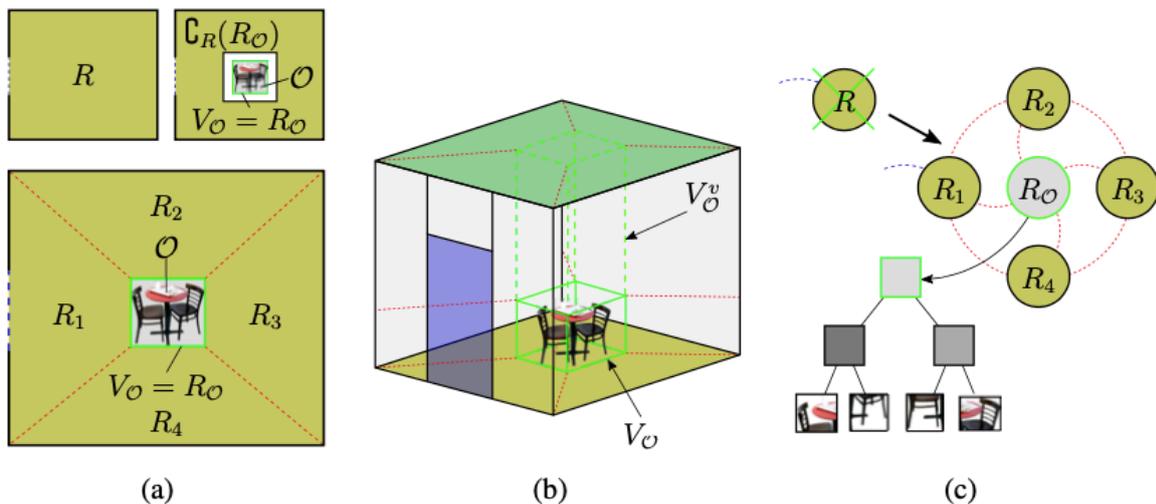


Figure 3.8 – Inclusion d'objets dans une CCSP : (a) illustration en 2D : le volume englobant  $V_O$  de l'objet  $O$  est inclus dans la région  $R$ ;  $R$  est subdivisée en régions convexes  $R_1, R_2, R_3$  et  $R_4$ ; (b) illustration en 3D : un volume vide ( $V_O^v$ ) est créé au-dessus de  $V_O$ ; (c) graphe associé : le noeud correspondant à la région  $R$  est subdivisé;  $R_O$  (représentant  $V_O$ ) est associée à une structure accélératrice classique construite à partir des primitives de  $O$ .

Pour un rendu rapide, la traversée des objets peut être optimisée en utilisant n'importe quelle structure accélératrice secondaire. Ainsi, une CCSP devient une structure accélératrice à plusieurs niveaux. Lorsqu'un rayon arrive dans un volume de sémantique OBJECT, nous effectuons un changement de système de coordonnées (selon la matrice de transformation de l'instance de l'objet) avant de traverser la structure accélératrice de l'objet. Si le rayon manque l'objet, la traversée continue jusqu'à ce qu'il atteigne un polygone de la scène.

L'utilisation d'un BVH comme une structure accélératrice secondaire est une solution suffisante pour avoir une application interactive de parcours de bâtiments meublés. Nous proposons dans la section 3.4 une méthode permettant d'associer une CCSP à un objet quelconque.

### 3.3 Résultats et discussion

Les scènes utilisées pour évaluer ce travail correspondent à des bâtiments réels conçus à partir de leurs plans architecturaux 2D. Ces environnements virtuels 3D sont directement générés à partir de notre modèleur topologique dédié aux environnements architecturaux (cf. section 3.1.2). Le tableau 3.1 résume leurs principales caractéristiques, notamment les temps de génération et l'occupation mémoire. Ce panel présente des scènes de test plus ou moins complexes en termes de nombre de polygones, de sources de lumière et d'objets (cf. figures 3.9 et 3.10).

Scènes	# Pièces	# Volumes		# Polygones		# Lumières	Construction en ms	# Mémoire	
		(vide)	(meublé)	(vide)	(meublé)			(vide)	(meublé)
ATRIUM	5	343	343	2,022	525,710	1	7.7	0.7	20.5
HOUSE	8	510	577	1,166	1,005,454	8	7.46	0.45	72.67
LAB	46	1,923	2,510	8,557	8,956,708	55	51.43	2.03	20.8
ADMIN	451	7,724	13,752	59,682	58,660,932	649	116.36	7.36	66.3
TOWER	2,209	94,345	120,534	496,057	356,242,576	3211	1929.9	108.9	173.6

TABLE 3.1 – Caractéristiques des scènes de tests : nombre de pièces, de volumes, de polygones, de lumières, temps de construction de la structure CCSP à partir de la GMapArch (single core CPU) et occupation mémoire de la CCSP en Mocketts.

Comme expliqué dans la section 3.1.2, la CCSP est directement déduite par un simple parcours d'une GMapArch. Néanmoins, étant donné qu'elle contient beaucoup d'informations inutiles pour notre méthode, elle est allégée pour réduire le coût mémoire en conservant uniquement les informations nécessaires. La colonne "construction" de tableau 3.1 indique le temps correspondant à ce traitement.

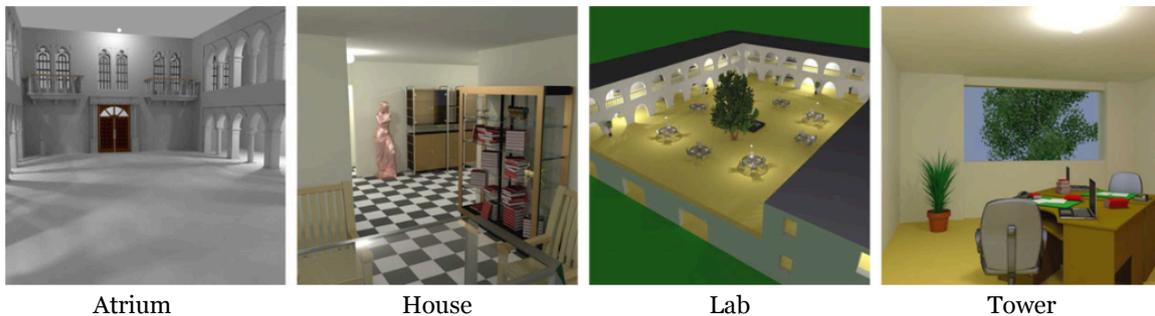


Figure 3.9 – Exemple de rendu de nos scènes de test.

#### 3.3.1 Système d'évaluation des performances

Pour évaluer les performances et la robustesse de notre structure, nous l'avons utilisée pour huit méthodes de rendu soit sur CPU, soit sur GPU :

- Trois méthodes sur le CPU, (un seul coeur (S) ou parallélisées avec OpenMP (P) avec implémentation simple : *Ray casting* est un ray casting primaire. *Packets* exploite les instructions Intel SSE et regroupe les rayons primaires en paquets pour profiter de la cohérence spatiale. *Packets Lighting* utilise également des paquets pour les rayons primaires et les rayons d'ombre dure.
- Cinq méthodes sur GPU, parallélisées avec CUDA (en utilisant la technique des threads persistants [2]) : *Ray casting* est un Ray casting primaire. *Lighting* utilise des rayons secondaires pour les ombres dures. *Traçage de chemin* qui implémente un traceur de chemin avec 5 rebonds par chemin (plus de détails dans section 3.3.3). *Objects* et *Objects Lighting* étendent respectivement *Ray casting* et *Lighting* pour les scènes meublées.

Les performances ont été évaluées sur une machine équipée d'un processeur Intel® Core™ i7-4930K CPU @ 3.40Ghz, 32 Go de RAM et une carte graphique NVidia® GeForce® GTX 680. Les images rendues ont une définition de 1024 × 1024 pixels sans anticrénelage (un seul rayon est lancé par pixel). Pour disposer de statistiques pertinentes, nous avons utilisé un système de way-point afin de conserver les mêmes points de vue pour chaque algorithme testé. Ces points de vue résultent d'un parcours dans chaque scène. Les performances sont mesurées en termes de :

- FPS : nombre d'images calculées par seconde comprenant la génération de rayons, le traçage de rayons, l'ombrage et le remplissage du tampon d'image (le temps d'affichage n'est pas inclus).
- Mrays/s : nombre de millions de rayons traités par seconde; correspond au nombre de rayons émis par image multiplié par FPS.

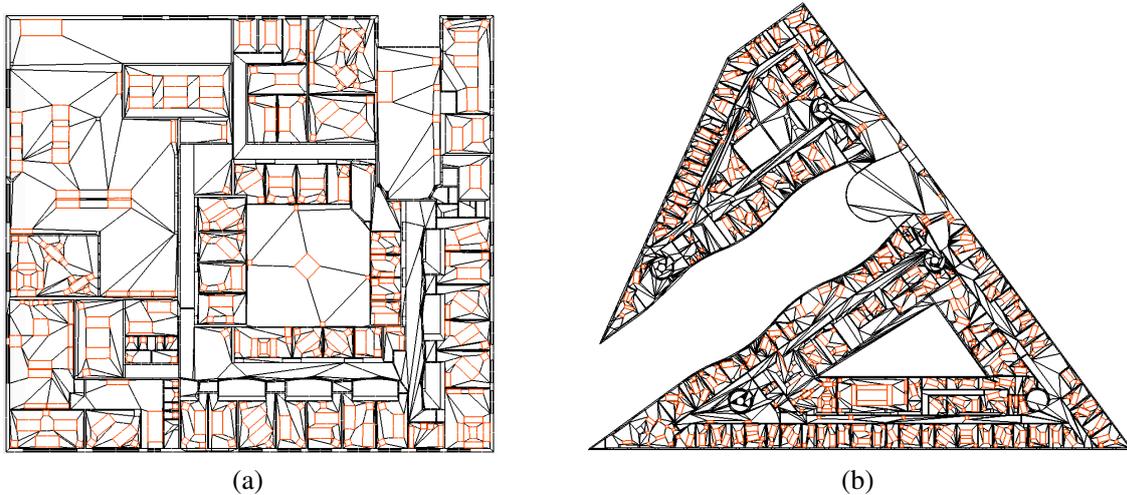


Figure 3.10 – Plans 2D correspondant à des étages de scènes de test ; les meubles sont représentés ici par des boites englobantes en rouge : (a) un étage de la scène ADMIN ; (b) un étage de la scène TOWER.

Méthodes			HOUSE		LAB		ADMIN		TOWER	
			FPS	Mrays/s	FPS	Mrays/s	FPS	Mrays/s	FPS	Mrays/s
CPU	Ray casting	(S)	21.57	22.62	24.11	25.28	21.09	22.11	17.66	18.51
		(P)	123.06	129.04	131.03	137.39	117.16	122.85	92.75	97.25
	Packet casting	(S)	135.94	142.54	136.51	143.14	128.13	130.16	132.64	139.08
		(P)	435.89	457.06	454.05	476.11	423.27	443.83	394.55	413.71
	Packet & Lighting	(S)	47.21	154.56	24.53	135.09	8.63	146.68	21.52	154.07
		(P)	180.36	590.46	100.08	551.22	38.91	661.09	76.19	545.42
GPU	Ray casting		1,389.16	1,445.66	1,064.71	1,116.43	1,219.92	1,279.18	1,096.61	1,149.87
	Lighting		439.47	1,438.59	179.54	988.88	64.99	1,104.61	145.49	1,041.96
	Path tracing		18.57	218.87	9.79	206.34	12.01	533.58	9.08	211.11
	Objects		191.39	200.69	245.77	257.7	456.57	478.75	237.32	248.85
	Objects & Lighting		58.28	163.48	28.13	183.12	19.06	289.85	25.63	163.21

TABLE 3.2 – Résultats : nombre moyen d'images par seconde (FPS) et moyenne en million du nombre de rayons traités par seconde (Mrays/s).

### 3.3.2 Etude de la complexité du calcul des rayons

Une particularité des méthodes de lancé de rayons dans une CCSP est que la complexité du calcul de parcours est linéaire par rapport au nombre de volumes traversés par le rayon. Comparer la complexité de traversée de CCSP avec des structures accélératrices habituelles telles qu'un Kd-tree ou un BVH n'est pas une tâche triviale. En effet, la recherche de la plus proche intersection dans de telles structures hiérarchiques nécessite de traverser l'arbre depuis la racine avant de tester l'intersection des rayons pour chaque feuille visitée. Le nombre d'étapes de traversée est au mieux logarithmique à la taille de l'arbre. En revanche, avec un CCSP, aucun test d'intersection rayon/polygone n'est réellement effectué et la recherche de la face de sortie est suffisante pour déterminer la plus proche intersection. Cela suppose que le temps de localisation de l'origine du rayon est négligeable (cf. section 3.2.1). La complexité de la traversée est linéaire par rapport au nombre global de volumes composant la partition. Ainsi, dans le meilleur des cas, un seul volume doit être traversé. À l'inverse, une scène composée d'un long couloir fortement divisé à un temps de traitement un peu plus long. En pratique, le nombre de volumes traversés par rayon reste assez faible.

La figure 3.11 montre la relation entre le FPS et le nombre moyen de volumes traversés par les rayons (noté  $\Phi$ ). Les statistiques présentées sont mesurées sur GPU et extraites d'un parcours dans la scène ADMIN. La courbe de tendance révèle clairement que les performances sont inversement proportionnelles au nombre de volumes traversés. Il est donc primordial d'utiliser une stratégie bien adaptée pour convexifier la scène tout en minimisant le nombre moyen de volumes traversés par un rayon.

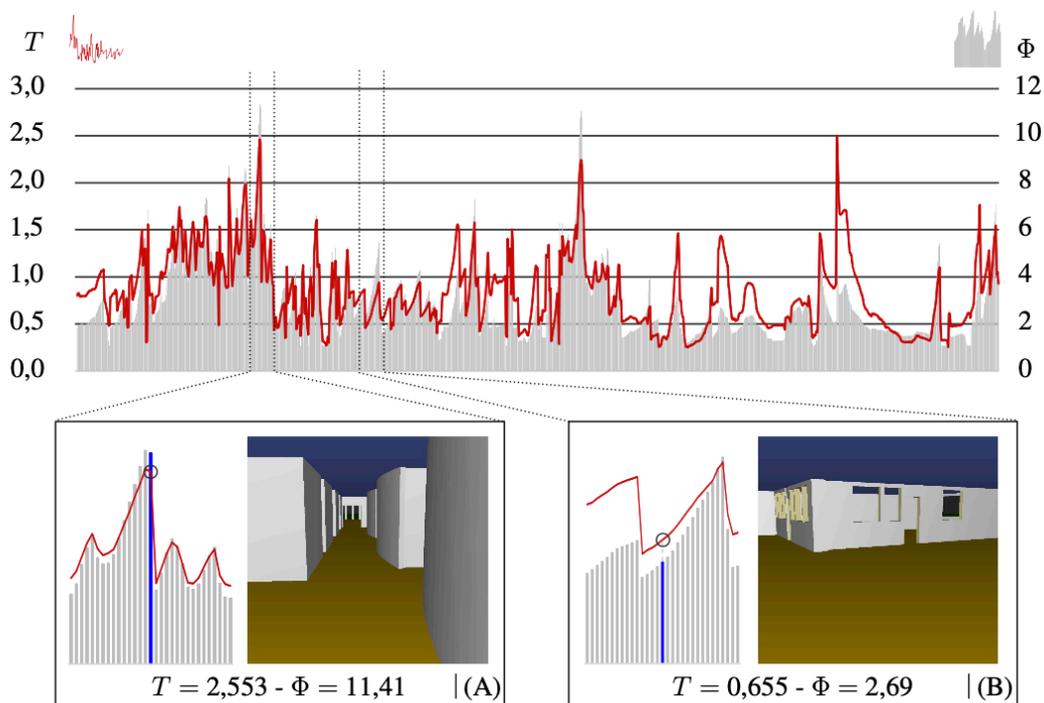


Figure 3.11 – Temps de rendu en ms ( $T$ , courbe rouge) et nombre de volumes traversés exprimé en millions ( $\Phi$ , histogramme gris) pour l'algorithme Ray casting avec 1 082 points de vue extraits durant une visite d'un étage du bâtiment ADMIN : depuis le point de vue (A), les rayons traversent un long couloir contenant des murs courbes concaves et donc beaucoup plus de volumes que depuis le point de vue (B).

### 3.3.3 Analyse comparative des méthodes

#### Ray casting

Le tableau 3.3 révèle que notre structure CCSP permet une gestion des rayons primaire très efficace que ce soit sur le CPU ou sur le GPU. Elle projette près de cinq cents millions de rayons primaires par seconde sur le CPU (*Packet casting (P)*) et plus d'un milliard sur le GPU. Le tableau 3.3 compare ces résultats avec ceux mesurés avec le moteur Embree [168], et avec le moteur proposé par Aila de Nvidia [3].

		HOUSE	LAB	ADMIN	TOWER
CPU	Notre Ray Tracing	123.06	131.03	117.16	92.75
	Notre Packets Tracing	435.89	454.05	423.57	394.55
	Embree	≈ 32	≈ 21	≈ 29	≈ 14
GPU	Notre Ray Tracing	1,389.16	1,064.71	1,219.92	1,096.61
	Aila	220-240	170-230	200-240	150-210
Mem.	Notre structure	0.45	2.03	7.36	108.9
	Embree	0.5	1.8	6.7	77.7
	Aila	0.49	3.44	9.28	135.73

TABLE 3.3 – Comparaisons entre notre raycasting et Embree [168] (CPU) et l'application de Aila *et al.* [3] (GPU) : performance en images par seconde et occupation mémoire en Moctet.

Cette comparaison révèle clairement que notre approche est plus performante que ces moteurs de lancé de rayons de pointe, que ce soit sur le CPU ou sur le GPU. Cependant, elle doit être prise avec précaution. En effet, le CCSP est particulièrement bien adapté aux scènes architecturales : elles sont composées de grandes surfaces occultantes, de sorte qu'un rayon donné n'a qu'à traverser quelques volumes pour trouver la plus proche intersection de rayons. De plus, la recherche de face de sortie est conçue pour les scènes architecturales, et utilise des informations topologiques dans toutes les dimensions. Même si cette haute performance ne prouve pas que la partition d'espace sous contrainte est une structure d'accélération alternative, elle révèle que cette approche mérite d'être explorée davantage.

#### Path tracing

Le tableau 3.2 montre les résultats de notre algorithme utilisé pour du tracé de chemins sur GPU. Nous pouvons remarquer l'écart de performance entre l'éclairage direct et le tracé de chemin en ce qui concerne le nombre de rayons projetés par seconde (perte de près de 75 %). Cette perte s'explique d'une part parce que la performance mesurée inclut le temps nécessaire pour générer des directions aléatoires pour les rebonds et pour choisir aléatoirement la source lumineuse testée. D'autre part, elle est due à l'exécution divergente au sein de chaque warp GPU. En effet, notre moteur est assez naïf, puisqu'il calcule chaque chemin l'un après l'autre sans stratégie spécifique de distribution des tâches.

#### Visualisation de scènes meublées

Comme illustré dans section 3.2.3, n'importe quelle structure d'accélération secondaire peut être utilisée pour rendre des objets tels que des meubles. En pratique, nous utilisons un BVH. Étant donné qu'un même objet peut apparaître plus d'une fois dans une scène, nous utilisons un système d'instanciation des objets pour réduire la consommation de mémoire.

Nous comparons nos performances à l'application de lancé de rayons sur GPU proposée par Aila [3]. Si nous bénéficions d'un système d'instanciation des objets pour économiser la consommation de mémoire, leur implémentation n'en fournit pas. Par conséquent, nous ne pouvons comparer les performances que sur la scène HOUSE, seule scène logeable en mémoire sans installation et pour laquelle nous sommes plus performants. En ce qui concerne l'éclairage avec la prise en compte des objets, nous avons enregistré en moyenne 32,77 FPS et 199,91 Mrays/s. Une fois encore, en termes de Mrays/s, les performances de notre structure restent stables.

Même si l'utilisation d'une structure secondaire de type BVH est une solution suffisante pour disposer d'une application interactive de visite de bâtiments meublés, elle constitue également un inconvénient majeur de l'approche proposée. Dans le cas des scènes meublées, les performances des méthodes de lancé de rayons dépendent directement de la structure accélératrice utilisée pour le rendu des objets. De plus, le passage d'une structure d'accélération à l'autre entraîne une divergence sur le GPU, ce qui ralentit le rendu. Du côté positif, le CCSP permet d'éviter de gérer toute la géométrie du mobilier.

### 3.3.4 Bilan

L'utilisation d'une CCSP comme structure accélératrice pour les méthodes de lancé de rayons offre plusieurs propriétés intéressantes : sa traversée a une complexité linéaire en nombre de volumes traités ; la traversée du CCSP donne l'intersection la plus proche, et aucun autre test d'intersection n'est nécessaire ; elle exploite implicitement la connectivité entre les rayons dans la construction des chemins de lumière ; elle fournit des informations topologiques dans toutes les dimensions, optimisant la recherche de la face de sortie ; enfin, elle est bien adaptée à une utilisation parallèle sur CPU ou GPU.

Nos expérimentations montrent l'efficacité de la méthode, même pour des scènes meublées composées de plusieurs centaines de millions de polygones avec le calcul de centaines de millions de rayons par seconde.

## 3.4 Visualisation topologique : généralisation

Le principal inconvénient de l'approche proposée pour la visualisation de bâtiment est qu'elle nécessite la description complète de la topologie pour construire une structure CCSP. Même si le modèleur MokaArchi fournit toutes ces informations, notre structure ne peut pas être utilisée avec une entrée purement géométrique. Par conséquent, nous avons souhaité appliquer la CCSP à tout type d'environnement, décrit uniquement par sa géométrie.

Cette généralisation repose sur trois points clés principaux. Premièrement, il est nécessaire de reconstruire la topologie d'une scène à partir de sa description géométrique. Ce processus n'est pas une tâche triviale puisqu'il doit traiter les incohérences géométriques telles que les éléments dégénérés et les problèmes de corafinement 3D. Ensuite, une partition d'espace convexe contraint en 3D doit être construite, en utilisant la description topologique et géométrique, afin de l'utiliser comme structure d'accélération dédiée au lancé de rayons. Ce processus nécessitera la recherche d'une heuristique bien adaptée minimisant le coût moyen de traversée d'un rayon donné à travers la CCSP. Enfin, un algorithme de recherche de face de sortie efficace et général doit être conçu.

La généralisation de la CCSP aux objets quelconques nécessite une méthode de décomposition convexe de polyèdres 3D. Fortune a introduit ce concept en proposant un traçage topologique du faisceau utilisant une subdivision convexe acyclique respectant les obstacles de la scène [49]. En 2008, Lagae et Dutré ont proposé d'utiliser une tétraédrisation de Delaunay contrainte (que nous nommerons CDT dans la suite du manuscrit) comme structure d'accélération pour le lancer de rayons [96].

Au cours de la thèse de Maxime Maria, nous avons étendu ce dernier travail pour proposer un nouvel algorithme de traversée de CDT. Celui-ci est basé sur une manière spécifique de représenter et de traverser les tétraèdres (issu de la description topologique). Il est intrinsèquement plus efficace puisqu'il nécessite moins d'opérations arithmétiques. Ensuite, il est plus adapté à la programmation parallèle puisqu'il n'ajoute pas de divergence supplémentaire. Enfin, il est robuste et ne provoque aucune erreur due à une instabilité numérique, que ce soit sur le CPU ou sur le GPU.

### 3.4.1 Construction de la CDT

Une CDT ne peut pas être construite à partir de tous les modèles géométriques d'entrée. Une condition suffisante est que le modèle géométrique d'entrée soit un complexe linéaire par morceaux (PLC) [119]. En 3D, toute intersection non vide entre deux faces d'un PLC doit correspondre à une arête ou un sommet partagé. En d'autres termes, il n'y a pas d'intersection entre les primitives géométriques.

En infographie, une scène est généralement représentée comme un ensemble non structuré de polygones, communément appelé *soupe de polygones*. Dans ce cas, certaines intersections peuvent exister. Néanmoins, il est possible de construire un PLC à partir d'une telle représentation géométrique, en utilisant une technique de réparation de maillage. Ce point sera traité dans le chapitre 5.

À partir d'un PLC, la CDT peut être construite en utilisant la bibliothèque *Tetgen* [77]<sup>1</sup>. Il en résulte un maillage tétraédrique, contenant deux types de faces : les faces occultantes, appartenant à la géométrie de la scène ; et de faces non occultantes, introduites pour construire la partition. Évidemment, un rayon donné doit traverser ces dernières, car une face non occultante n'appartient pas à la géométrie d'entrée.

### 3.4.2 Traversée de la CDT

Comme nous l'avons vu précédemment pour l'utilisation d'une CCSP comme structure d'accélération, la recherche de l'intersection la plus proche entre un rayon et la géométrie de la scène s'effectue en deux étapes principales : (1) le volume contenant l'origine du rayon est localisé (ici un tétraèdre) ; (2) le rayon traverse la CDT en parcourant un tétraèdre à la fois jusqu'à ce qu'il rencontre une face occultante.

Pour une application interactive, la localisation de l'origine reste un problème négligeable, puisqu'elle n'apparaît que pour la première image. Ensuite, une variante de l'algorithme de traversée (prenant en compte une distance maximale) résout efficacement ce problème lors du déplacement de la caméra. La localisation de l'origine des rayons non primaires est évitée en exploitant la connectivité implicite des rayons dans la CDT : le point de départ et le volume correspondent tous deux à l'arrivée du rayon précédent.

L'algorithme de traversée de la CDT repose sur une boucle, et recherche la face de sortie d'un rayon depuis l'intérieur d'un tétraèdre. Nous proposons une nouvelle méthode rapide et robuste pour résoudre le problème de la face de sortie. Les tests sont ordonnancés en utilisant les informations topologiques du modèle et les calculs sont réduits en utilisant une nouvelle fois les coordonnées de Plücker (cf. section 3.2.1) pour déduire le positionnement du rayon par rapport aux bords des faces.

### Recherche de la face de sortie

Notre algorithme suppose que la face d'entrée est connue, le rayon provient du tétraèdre précédent et il traverse le tétraèdre actuel. C'est pourquoi, pour une face d'entrée donnée, son complémentaire dans le tétraèdre est utilisé : un sommet, trois arêtes et trois faces.

1. *TetGen* est un programme permettant de générer des maillages tétraédriques à partir de données polyédrique 3D

---

**Algorithme 3.2** Recherche de la face de sortie à l'intérieur d'un tétraèdre
 

---

**ENTRÉE:**  $F_e = \{S, \Lambda_0, \Lambda_1, \Lambda_2\}$  : Face d'entrée;  $\Lambda_r$  : rayon;

**SORTIE:**  $F_s$  : face de sortie;

- 1:  $side \leftarrow \Lambda_r \otimes F_e.\Lambda_2$ ;
  - 2:  $id \leftarrow (side \geq 0)$ ;  $\{id \in \{0, 1\}\}$
  - 3:  $side \leftarrow \Lambda_r \otimes F_e.\Lambda_{id}$ ;
  - 4:  $id \leftarrow id + (side < 0)$ ;  $\{id \in \{0, 1, 2\}\}$
  - 5:  $F_s \leftarrow \text{getFace}(F_e, id)$ ;
  - 6: **return**  $F_s$ ;
- 

$\Lambda_0, \Lambda_1$  et  $\Lambda_2$  sont désignés pour représenter les arêtes complémentaires (qui ne sont pas incidentes à la face d'entrée), avec une orientation dans le sens antihoraire depuis l'intérieur du tétraèdre (cf. figure 3.12). Les faces complémentaires sont désignées par un identifiant local 0, 1 et 2, en utilisant la convention suivante : la face 0 est délimitée par les arêtes  $\Lambda_2$  et  $\Lambda_0$ , la face 1 est délimitée par les arêtes  $\Lambda_0$  et  $\Lambda_1$ , la face 2 est délimitée par les arêtes  $\Lambda_1$  et  $\Lambda_2$ .

En utilisant l'opérateur de *side* de Plücker, la face traversée par  $r$  est déterminée :

- face 0 si et seulement si  $r \otimes \Lambda_2 < 0$  et  $r \otimes \Lambda_0 \geq 0$  (cf. figure 3.12.d);
- face 1 si et seulement si  $r \otimes \Lambda_0 < 0$  et  $r \otimes \Lambda_1 \geq 0$ ;
- face 2 si et seulement si  $r \otimes \Lambda_1 < 0$  et  $r \otimes \Lambda_2 \geq 0$ .

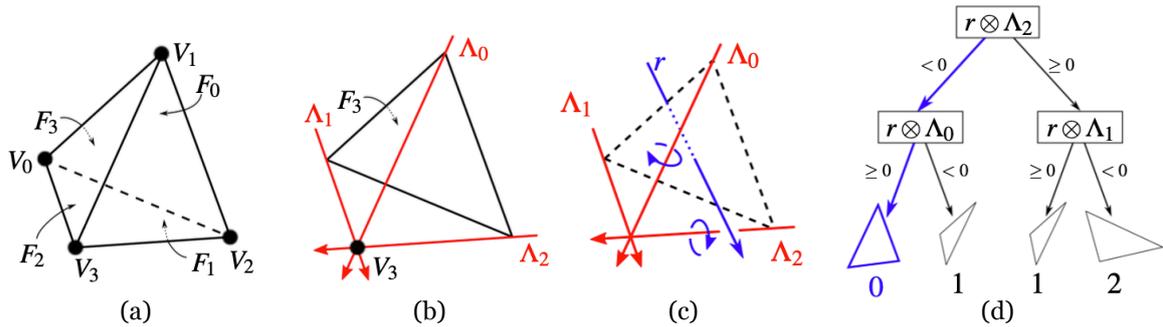


Figure 3.12 – Description et traversée d'un tétraèdre : (a) numérotation des sommets et des faces; (b) à titre d'exemple, le sommet complémentaire de  $F_3$  est  $\{V_3\}$ , et ses arêtes sont  $\Lambda_0 = V_1V_3$ ,  $\Lambda_1 = V_0V_3$  et  $\Lambda_2 = V_2V_3$ . (c) Le rayon  $r$  entre dans le tétraèdre par la face arrière; (d)  $r \otimes \Lambda_2 < 0$  et  $r \otimes \Lambda_0 \geq 0$ , donc la face de sortie est identifiée par 0.

Ces conditions sont organisées dans un arbre de décision (cf. figure 3.12.d). Chaque feuille correspond à une face de sortie, et chaque noeud intérieur répond à un opérateur *side* entre le rayon  $r$  et les arêtes  $\Lambda_i$ . L'ordonnancement des tests est guidé par les informations topologiques de dimension 2, les liens entre les faces. Comme toutes les branches de l'arbre de décision ont une profondeur fixe de 2, l'algorithme répond en utilisant exactement deux opérations. De plus, il est optimisé pour fonctionner efficacement sans aucun `if` conditionnel (cf. Algorithme 3.2).

Cet algorithme se termine par le renvoi du numéro de la face du tétraèdre en fonction de la face d'entrée et de l'étiquette de la face de sortie  $\in \{0, 1, 2\}$ . Elle répond à l'aide d'une table de consultation, définie en utilisant une combinatoire simple (cf. tableau 3.4), en supposant un étiquetage cohérent des faces du tétraèdre (cf. figure 3.12.a).

Identifiant de sortie	Face d'entrée			
	$F_0$	$F_1$	$F_2$	$F_3$
0	$F_1$	$F_0$	$F_0$	$F_0$
1	$F_2$	$F_3$	$F_1$	$F_2$
2	$F_3$	$F_2$	$F_3$	$F_1$

TABLE 3.4 – Face de sortie en fonction de l'identifiant de la face d'entrée  $\in \{0, 1, 2\}$ , et selon une numérotation cohérente des faces (cf. figure 3.12.a).

	BANANA	BUNNY	ARMADILLO	
				
Tetraèdre	71,300	682,733	2,990,552	
faces occultantes	24,568	222,775	1,105,218	
faces non-occultantes	117,994	1,142,650	4,875,834	
Memoire (MB)	6	62	273	
Ray-casting	CPU	9.76	10.5	6.75
	GPU	428	289	123

TABLE 3.5 – Caractéristiques des scènes et performances : nombre de tétraèdres, nombre de faces occultantes (les faces provenant du modèle), nombre de faces non occultantes (les faces créées pendant la tétraédrisation), mémoire occupée et performances de lancé de rayons en millions de rayons par seconde sur le CPU et le GPU.

### 3.4.3 Résultats

Les performances sont évaluées à l'aide de trois scènes de test. Leurs principales caractéristiques ainsi que les performances sont présentées dans le tableau 3.5. La plus simple est une banane composée de 25k faces occultantes. Les deux autres correspondent à des versions raffinées de deux scènes bien connues de l'université de Standford : BUNNY et ARMADILLO. Elles comptent respectivement 200k et 1.1M de faces occultantes.

Les performances sont mesurées en termes de millions de rayons lancés par seconde (Mrays/s), pour des images d'une définition de  $1024 \times 1024$  pixels et sans anti-aliasing. L'ordinateur utilisé possède un processeur Intel® Core™ i7-4930K CPU @ 3.40Ghz, 32 Gb de RAM et une NVidia® GeForce® GTX 680. Les algorithmes sont rendus parallèles sur le CPU (avec OpenMP) et le GPU (avec CUDA et la technique des threads persistants [2, 3]). Sur le CPU, le lancer de rayons atteint 9 Mrays/s en moyenne, et 280 Mrays/s sur le GPU.

#### Traversée de la structure

L'intersection rayon/scène la plus proche est trouvée en traversant le CDT d'un tétraèdre à la fois jusqu'à ce qu'il rencontre une face occultante. La complexité de la traversée pour un rayon donné est linéaire dans le nombre de tétraèdres traversés. La figure 3.13 montre la relation entre le temps d'exécution ( $T$ ) et le nombre de tétraèdres traversés par image ( $\Phi$ ). Les statistiques sont extraites en utilisant 1282 points de vue de la scène BUNNY, sur CPU.

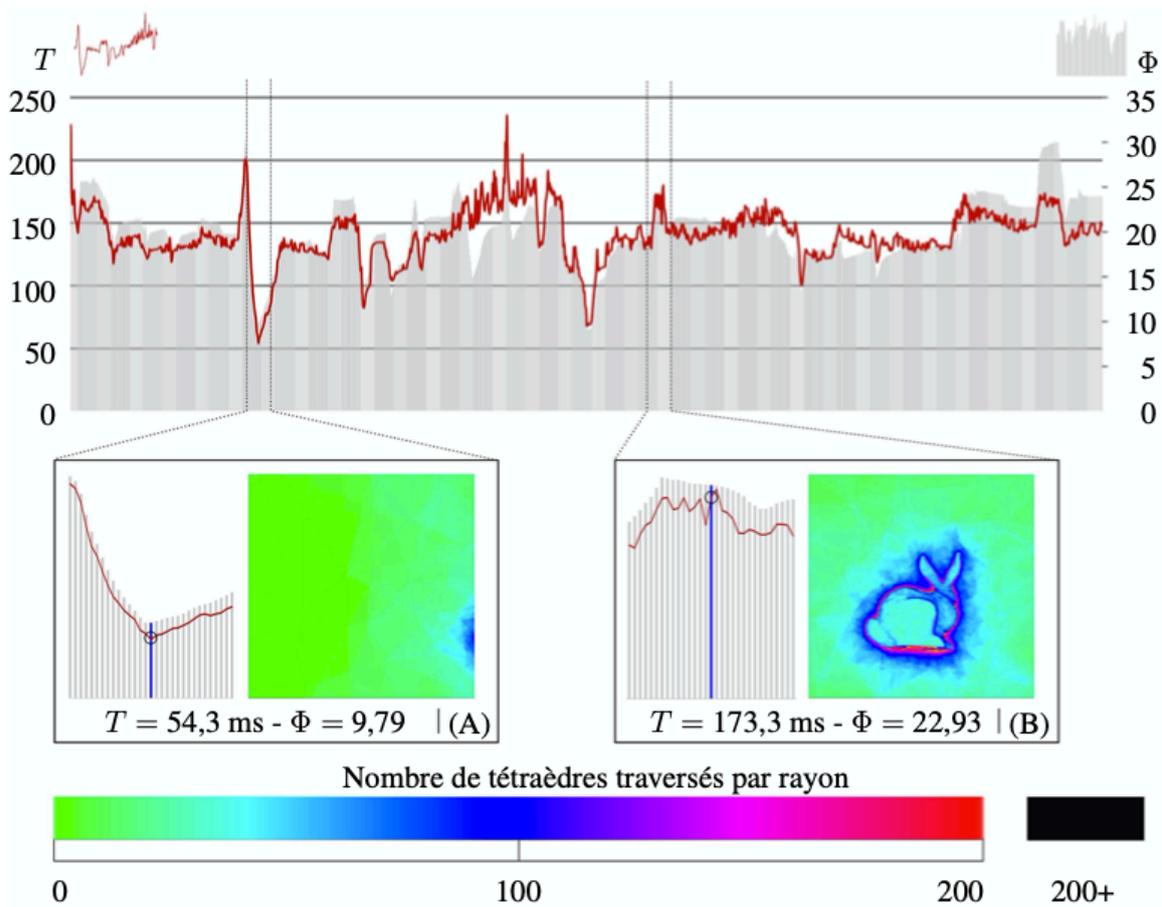


Figure 3.13 – Temps de rendu sur le CPU en ms ( $T$ , courbe rouge) et nombre de traversées en millions ( $\Phi$ , barres grises) en utilisant 1 282 points de vue et la scène BUNNY. Points de vue et la scène BUNNY ; (A)  $T = 40,1 \text{ ms}$   $\Phi = 9,6$  ; (B)  $T = 122 \text{ ms}$   $\Phi = 22,71$ .

On constate que le temps d'exécution est proportionnel à  $\Phi$  : sur le point de vue (A), près de 10 millions de tétraèdres sont parcourus en 40.1 ms ; sur le point de vue (B), près de 23 millions de tétraèdres sont parcourus en 122 ms. L'image faussement colorée du point de vue (B) révèle que les rayons qui vont près de la limite de l'objet traversent plus de tétraèdres. Ces mesures montrent que la configuration du CDT a un impact direct sur les performances de traversée, et que sa construction doit être bien contrôlée.

### Comparaisons

Nous n'avons pas remarqué de divergence numérique avec notre méthode. Ceci est notamment expliqué par le nombre d'opérations arithmétiques effectuées. En effet, notre algorithme nécessite moins d'opérations que les autres méthodes utilisées, de sorte que moins d'erreurs numériques sont accumulées, ce qui conduit à un résultat plus précis. En raison de l'instabilité des autres méthodes, nous ne pouvons pas comparer facilement notre algorithme avec les précédents dans le contexte du rendu d'images. Ainsi, notre algorithme est comparé avec le traceur de rayons implémenté par Aila [3]. La structure d'accélération est un BVH, construit avec SAH. Cette implémentation est, au moment de la réalisation de ce travail, la plus rapide. Le tableau 3.6 résume cette comparaison. Il faut noter la différence d'amplitude des performances entre CDT et BVH. Cet écart peut être expliqué en analysant la complexité de traversée des deux structures. La traversée de CDT a une complexité en nombre de tétraèdres traversés, de sorte que

les performances dépendent fortement du point de vue. Au contraire, la complexité de BVH dépend du nombre global de primitives géométriques composant la scène. De plus, lorsqu'elle est construite suivant SAH, ses performances sont moins impactées par le point de vue, et l'amplitude des performances est plus faible.

Pour la scène ARMADILLO, les performances de BVH sont bien supérieures à celles de CDT. Cependant, ces résultats doivent être pris avec précaution. D'une part, le BVH et les autres structures d'accélération habituelles sont étudiées depuis plusieurs décennies, alors que le CDT a été beaucoup moins étudié. D'autre part, étant donné sa complexité de traversée, la CDT est particulièrement adaptée aux scènes fortement occultées (moins les tétraèdres sont traversés, plus les performances sont élevées), ce qui n'est pas le cas avec des scènes telles que ARMADILLO.

	CDT	BVH [3]
BANANA	315-947	200-260
BUNNY	130-1040	160-260
ARMADILLO	82-160	130-260

TABLE 3.6 – Comparaison des performances par rapport à [3], en nombre d'images par seconde.

### 3.4.4 Bilan

Dans cette section, nous proposons un nouvel algorithme pour le parcours de rayons au travers d'une tétraédrisation de Delaunay Contraintes (CDT). Grâce à une manière spécifique de représenter les tétraèdres, cet algorithme offre plusieurs avantages par rapport aux précédents. Premièrement, il est intrinsèquement plus efficace puisqu'il nécessite moins d'opérations arithmétiques. Ensuite, il est plus adapté au calcul parallèle, puisqu'il comporte un nombre fixe d'opérations, de sorte qu'il n'ajoute pas de divergence supplémentaire entre les fils. Enfin, il est robuste et fonctionne avec des flottants 32 bits, que ce soit sur le CPU ou le GPU.

La complexité de la traversée des rayons d'un CDT est linéaire par rapport au nombre de tétraèdres traversés : moins il y a de tétraèdres traversés, plus les performances sont élevées. Ainsi, l'efficacité de la traversée dépend fortement de la configuration du CDT.

Notre CDT est directement construite à partir des informations géométriques et topologiques disponibles lors de la phase de modélisation. Elle contient ainsi plus d'informations topologiques que les modèles habituellement employés. Nous mettons à contribution les relations topologiques inutilisées habituellement pour le calcul d'images de synthèse avec pour objectif de réduire fortement les temps de calcul.

## 3.5 Conclusion

Dans ce premier axe de recherche, nous avons présenté comment un modèle géométrique, topologique et sémantique dédié à la modélisation architecturale peut être utilisé directement comme structure accélératrice pour une application de lancé de rayons interactif. Cette structure est basée sur une partition spatiale contrainte et convexe (CCSP). La partition est dite contrainte, car elle respecte parfaitement la géométrie de la scène, au sens où les primitives géométriques sont confondues avec les bords des volumes constituant la partition. Le parcours de la CCSP est simple et optimisé par une stratégie d'ordonnement des traitements utilisant les relations topologiques entre les primitives de la scène. Ses performances ont été évaluées à la fois sur CPU et sur GPU et les expérimentations montrent qu'elle

surpasse largement les structures accélératrices classiques. En effet, elle permet de lancer des centaines de millions de rayons à la seconde pour des scènes architecturales composées de plusieurs centaines de millions de polygones.

Toutefois, notre processus de construction de la CCSP est basé sur une application dédiée de reconstruction de scènes architecturales 3D à partir de plans numériques 2D. La visualisation par lancé de rayons est donc conditionnée par le résultat de l'application de modélisation et la possession des plans numériques d'origine. Il n'est pas possible de l'utiliser pour des scènes construites directement en 3D par un autre outil de modélisation (par exemple les scènes SodaHall, Sponza, San Miguel, etc.). C'est pourquoi nous avons poursuivi notre étude pour la visualisations d'objets quelconques.

Après avoir étudié les méthodes existantes, nous avons choisi de construire une partition topologique de l'espace basée sur une tétraédrisation de Delaunay contrainte. Cette représentation particulière des tétraèdres nous permet de proposer un algorithme de recherche de la face de sortie d'un rayon depuis l'intérieur d'un tétraèdre. Cet algorithme est plus efficace que ceux présentés dans la littérature, il utilise notamment les liens topologiques entre les faces du tétraèdre pour ordonnancer les calculs et réduire les problèmes de stabilité numérique sur CPU et GPU. Bien que les performances restent modestes par rapport à l'utilisation de structures accélératrices classiques telle qu'une hiérarchie de volumes englobants, elles ont mis en avant des perspectives de recherche intéressantes. Tout d'abord, la méthode pourrait être enrichie par une construction d'une tétraédrisation se basant sur une heuristique selon le coût de traversée des tétraèdres (type SAH).

Récemment, de nouveaux travaux ont été réalisés par U. Güdükbay et I. Wald avec pour objectif de proposer un lancé de rayons volumique basée sur la traversé d'éléments tétraédriques [6, 141]. Ces méthodes assurent une meilleure gestion des caches GPU et du stockage en mémoire permettant de gérer des environnements plus complexes. Les performances sont comparables aux méthodes de traversée basées sur les structures d'accélération de l'état de l'art telles que les Kd-tree et BVH. Les auteurs précisent toutefois que la principale limitation de l'utilisation d'une partition basée sur des tétraèdres réside dans le processus de génération de maillage tétraédrique qui est coûteux en calcul et nécessite un prétraitement où les intersections géométriques doivent être résolues pour que la géométrie résultante soit un complexe linéaire par morceaux (PLC).

À partir de ces travaux, de nombreuses perspectives de recherche restent ouvertes. Par exemple, il serait intéressant d'étudier le comportement de la CCSP pour des environnements de simulations déformables et/ou dynamiques. Si les changements géométriques ne modifient pas la topologie de la partition et respectent la contrainte de convexité, alors aucune opération n'est nécessaire (excepté la mise à jour des coordonnées géométriques) et la CCSP reste valide. Dans le cas contraire, il est nécessaire de modifier la partition. Cependant, il est possible d'exploiter la structure de graphe de la CCSP pour la modifier localement et garantir sa validité. Ainsi, la CCSP unirait les structures utilisées en modélisation, simulation et animation.

L'ensemble de ce travail nous a permis également de mettre en évidence la pertinence d'utiliser une structure topologique et hiérarchique pour gérer la répartition des primitives géométriques dans la scène. En effet, la gestion du mobilier par localisation dans des boîtes englobantes, elle même incluses dans la structure du bâtiment, nous a permis de gérer un affichage interactif pour des scènes architecturales composées de plusieurs centaines de millions de polygones. Dans la suite de ce manuscrit, nous proposons d'approfondir cette utilisation avec les problématiques de visualisation des détails fins en synthèse d'images photoréalistes.

## Publications relatives au chapitre

### Revue internationale

*Constrained Convex Space Partition for ray-tracing.*  
M. Maria, S.Horna, L. Aveneau.  
Computer Graphics Forum, 2015, pp. Volume 36, pages 288-300.

### Conférences internationales avec comité de lecture

*Efficient Ray Traversal of Constrained Delaunay Tetrahedralisation.*  
M. Maria, S.Horna, L. Aveneau.  
Conference on Computer Graphics Theory and Applications (GRAPP), 2017.

*Topological Space Partition for Fast Ray Tracing in Architectural Models.*  
S. Horna, G. Damiand, A. Diakité, D. Meneveaux.  
Eurographics Workshop on Urban Data Modelling and Visualisation (UDMV), 2014, pp. 13–18.

*Topological Space Partition for Fast Ray Tracing in Architectural Models.*  
M. Maria, S.Horna, L. Aveneau.  
Conference on Computer Graphics Theory and Applications (GRAPP), 2014.

### Conférences nationales sans comité de lecture

*Visualisation interactive de grands bâtiments.*  
M. Maria, S.Horna, L. Aveneau.  
Journées Françaises de l'informatique Graphique, 2013.Limoges - Meilleur papier

*Visualisation interactive de grands bâtiments.*  
M. Maria, S.Horna, L. Aveneau.  
Groupe de Travail Rendu, 2013.Paris

---

# VISUALISATION D'ARRONDIS RÉALISTES

---

## Sommaire

---

<b>4.1 Construction de la structure accélératrice SREC-RT . . . . .</b>	<b>55</b>
<b>4.2 Utilisation de SREC-RT en lancé de rayons . . . . .</b>	<b>59</b>
<b>4.3 Représentation des arrondis . . . . .</b>	<b>61</b>
<b>4.4 Résultats . . . . .</b>	<b>64</b>
<b>4.5 Conclusion . . . . .</b>	<b>68</b>

---

Nous n'y prêtons pas une attention particulière, mais les bords des objets qui nous entourent sont systématiquement arrondis. Mis à part les objets tranchants (couteaux, lame de cutter, etc.), les extrémités des objets ne sont pas composés d'arêtes franches, un chanfrein, même très léger est toujours présent. Ces courbures sont produites lors du processus de fabrication : moulage, pliage ou encore chanfreinage. L'arrondi à un effet plus ou moins prononcé, mais dans tous les cas, sur le plan visuel, il réfléchit la lumière d'une manière bien particulière à laquelle notre sensibilité visuelle est très attachée.

En effet, la variation de l'orientation de ces surfaces produit des détails d'ombrage qui sont visuellement essentiels au réalisme des scènes synthétiques. Plus la surface est spéculaire, plus ses reflets sont fins et saillants. Les effets de lumière sur le bord des objets doivent donc être traités avec une attention particulière pour ne pas entraîner d'incohérence visuelle lors de la génération d'images photoréalistes. Les reflets et les ombres qui apparaissent dans ces zones aident notamment l'observateur à distinguer clairement les silhouettes d'un objet. Par exemple, dans la figure 4.1, sans arrondi, le Rubik's cube semble être constitué d'un seul bloc, alors que les petits cubes sont visibles individuellement lorsque les bords et les coins sont arrondis.

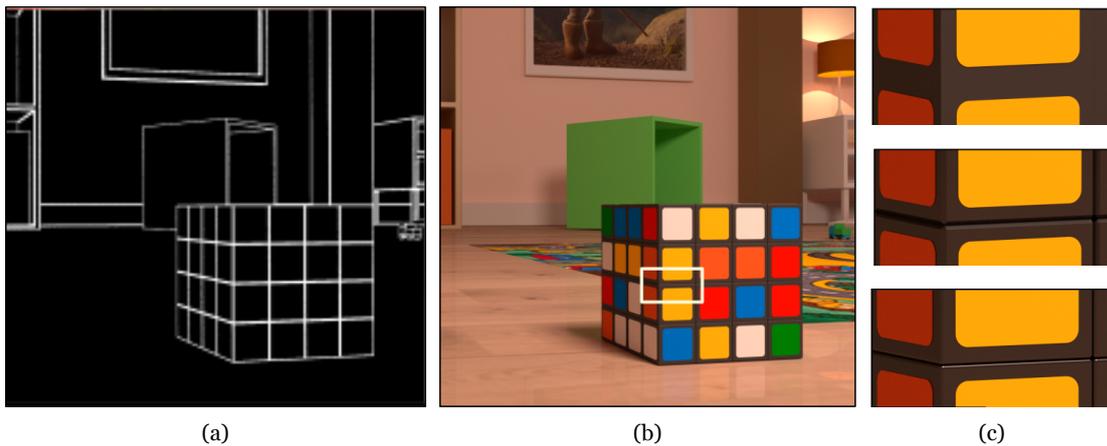


Figure 4.1 – La présence des surfaces arrondies change la perception visuelle des objets : (a) masque d'une image illustrant les pixels concernés par un arrondi (b) image calculée par notre méthode (c) différence visuelle entre un modèle sans arrondi, ou avec un arrondi plus ou moins discrétisé.

Nous avons vu dans la section 2.1.3 que dans la plupart des modélisateurs géométriques, la représentation des arrondis est réalisée avec des maillages polygonaux denses. Cela a pour effet de discrétiser le chanfrein tout en augmentant considérablement la complexité des scènes. La répartition hétérogène des primitives géométriques peut mettre en défaut les structures accélératrices classiques (cf. section 2.1.1). De plus, la subdivision du maillage ne permet pas de simuler correctement les effets lumineux, même avec un niveau de subdivision élevé. Cela augmente fortement la quantité de mémoire nécessaire et lorsque la distance d'arrondi est grande ou lorsque les chanfreins sont observés de près, des discontinuités restent visibles. L'utilisation de surfaces paramétriques permet une meilleure visualisation des chanfreins. Néanmoins, ces éléments paramétriques sont difficiles à intégrer dans le processus de rendu et le coût d'intersection d'une telle représentation est supérieur à un ensemble polygonal.

L'axe de recherche présenté dans ce chapitre aborde le rendu des effets de chanfrein lisses avec un faible impact sur la complexité géométrique et avec une gestion des calculs d'intersections rayon-objet efficace. Il se place de manière plus générale dans le projet région SIMOREVA (2018 - 2022) qui traite de la perception des environnements virtuels. Les travaux présentés dans la suite ont été majoritairement réalisés par Simon Courtin au cours de sa thèse dont le sujet concerne le rendu réaliste des bords arrondis. Pour cela nous proposons une nouvelle structure de détection topologique et hiérarchique (dénommée SREC-RT) qui ne modifie pas la géométrie originale et qui permet d'éditer facilement les paramètres d'arrondi. Notre objectif principal est d'utiliser cette structure pour mieux répartir les calculs du lancer de rayons et de nous abstraire du modèle de représentation utilisé pour modéliser les chanfreins (discrétisation polygonale ou surfaces paramétriques). La figure 4.2 illustre une vue d'ensemble de ce travail avec la construction de la structure SREC-RT et son utilisation pendant le processus de rendu.

La construction de SREC-RT consiste à générer une structure de détection globale du type *domaine arrondi* pour chaque face d'un objet. Le domaine arrondi d'une face correspond à la zone le long d'un bord qui doit être remplacée par une géométrie de chanfrein. Chaque arête et sommet de la géométrie originale sont ainsi traités avec leur géométrie adjacente (faces adjacentes et arêtes incidentes) pour déterminer la position des surfaces arrondies et leur connexion sur la géométrie originale. Cette structure englobante est basée sur des volumes limites définis par des cylindres et des sphères, chacun délimitant avec précision le chanfrein. L'avantage d'utiliser cette hiérarchisation de l'espace est que le modèle de représentation des surfaces d'arrondi sous-jacentes peut être défini par n'importe quelles primitives géométriques (linéaire ou paramétrique). Pour améliorer la gestion de la propagation des rayons le long des chanfreins, des liens topologiques entre les volumes limites sont instanciés, construits à partir de la géométrie d'entrée et imbriqués dans la structure accélératrice utilisée pour le lancer de rayons.

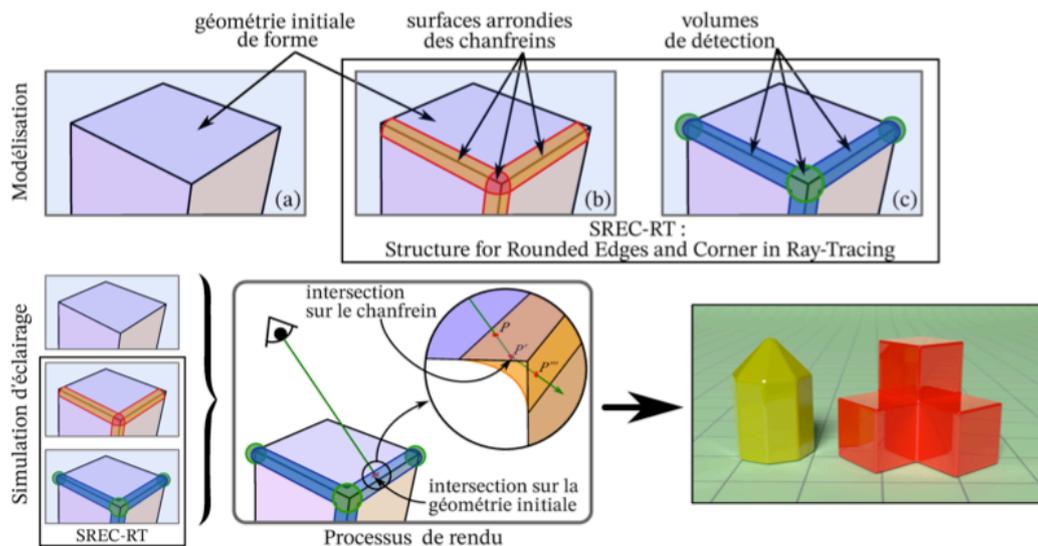


Figure 4.2 – SREC-RT est construite à partir (a) de la géométrie d’entrée, en positionnant (b) des zones d’arrondis sur les bords des polygones. Le résultat est une (c) structure de volumes englobants utilisée pendant le rendu pour déterminer efficacement les intersections des rayons avec la géométrie arrondie.

Enfin, SREC-RT est utilisée pendant le processus de rendu : (i) elle permet de déterminer efficacement si une intersection se produit avec un domaine arrondi, minimisant ainsi les calculs d’intersection de rayons ; (ii) le trajet du rayon est étendu jusqu’à l’intersection de la surface arrondie uniquement lorsque cela est nécessaire ; (iii) les liens topologiques assurent une propagation efficace du rayon à l’intérieur du domaine d’arrondi dans les cas où les rayons rasants n’intersectent pas immédiatement le chanfrein.

Dans la suite de ce chapitre, la section 4.1 détaille le processus de construction de la structure SREC-RT. La section 4.2 décrit le processus de lancé de rayons en utilisant cette nouvelle structure. La section 4.3 présente une solution de représentation des arrondis permettant une meilleure précision géométrique. Nous proposons ainsi une solution avec des surfaces paramétriques, basée sur des patches de Béziérs triangulaires modifiés de type Gregory pour les coins arrondis et des courbes de Béziérs 2D extrudées pour les bords arrondis. La section 4.4 décrit les résultats obtenus. Notre méthode a été intégrée dans le modèleur *Blender* [20] avec les moteurs de rendu *Cycles* [21] et *PBRT* [127]. Nous analysons les comparaisons générées avec les méthodes existantes. Enfin, un bilan est présenté dans la section 4.5.

## 4.1 Construction de la structure accélératrice SREC-RT

Les arrondis sont définis à partir d’une distance d’arrondi  $d$  spécifiée par l’utilisateur pour chaque objet sélectionné. L’objectif est de construire une structure composée de volumes englobants qui entourent les régions concernées par des arrondis. Notre processus est générique et permet de construire la structure quelle que soit la configuration géométrique des données d’entrée.

### 4.1.1 Positionnement des surfaces arrondies

La figure 4.3.a illustre les notations utilisées et le positionnement d'un bord arrondi le long d'une arête. Considérons deux faces adjacentes  $a$  et  $b$  avec leurs normales respectives  $\vec{N}_a$  et  $\vec{N}_b$ . L'arête incidente est définie par deux sommets  $v_1$  et  $v_2$ . Soit  $\vec{E} = v_2 - v_1$  le vecteur orienté pour l'arête ( $\vec{E}' = \vec{E}/|\vec{E}|$  est sa normalisation). Nous considérons  $\vec{F}_a = \vec{N}_a \times \vec{E}'$  et  $\vec{F}_b = \vec{N}_b \times -\vec{E}'$ , les directions de rétraction géométrique des faces  $a$  et  $b$  affectées par la surface arrondie. Les points  $d_1 = v_1 + \vec{F}_a \cdot d$  et  $d_2 = v_1 + \vec{F}_b \cdot d$  définissent les limites de la surface arrondie le long du bord considéré à  $v_1$ . Les points  $d_3$  et  $d_4$  sont définis de façon similaire en  $v_2$ .

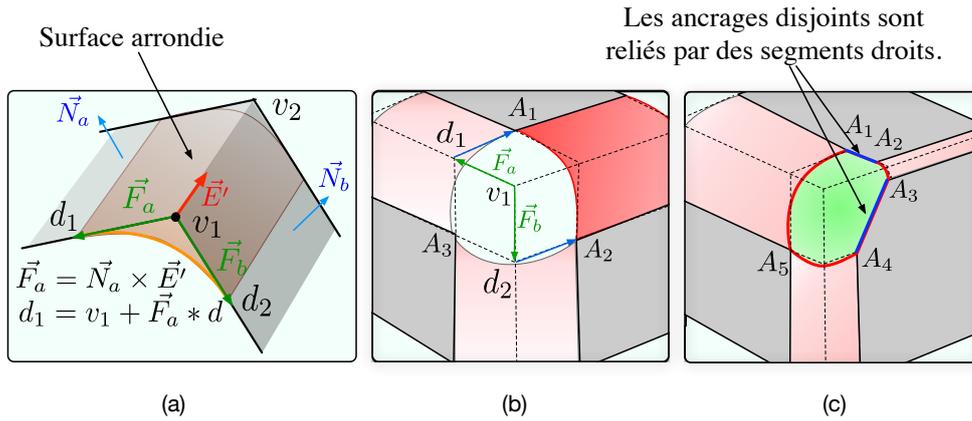


Figure 4.3 – Positionnement des arrondis (a) spécification pour une arête. (b-c) arrondi d'un sommet soit (b) incident à des arêtes d'arrondi identiques ou (c) cas général où les distances d'arrondi des arêtes sont hétérogènes.

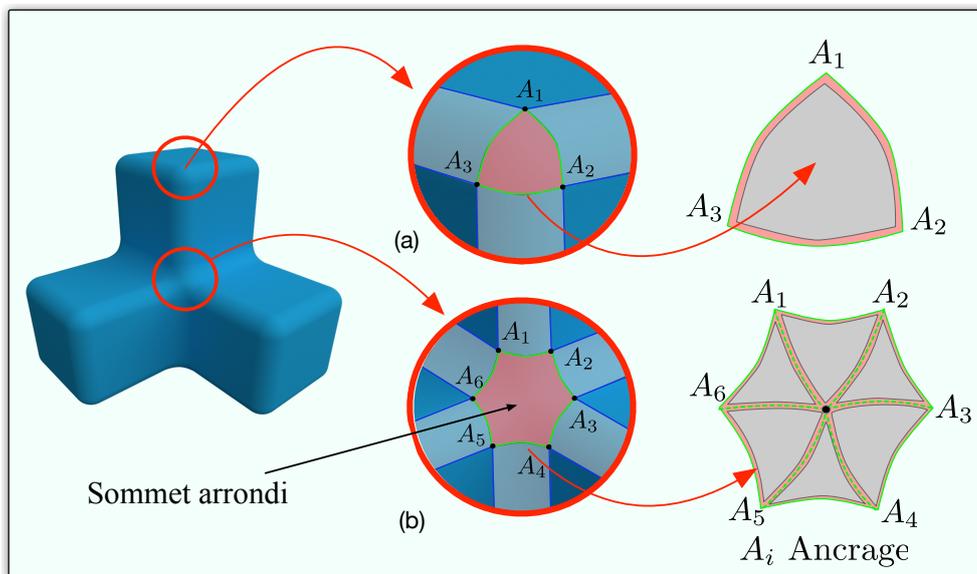


Figure 4.4 – Notre opération d'arrondi peut être utilisée pour les objets incluant des sommets arrondis incidents (a) à des arêtes convexes (b) des arêtes concaves et convexes. Les sommets arrondis (en rouge) sont délimités par les points d'ancrage  $A_i$ , positionnés sur les arêtes incidentes.

Le processus est général, quel que soit le nombre d'arêtes incidentes et leur configuration (arêtes rentrantes ou sortantes). Pour assurer la connexion entre la surface des chanfreins et la face initiale, des points d'ancrage (notés  $A_i$ ) sont calculés avant de créer une surface arrondie aux sommets (cf. figure 4.3.b). La frontière d'un sommet arrondi est définie en utilisant les points d'ancrage deux par deux dans le sens horaire. Cette méthode généralisée permet de traiter des sommets incidents à des arêtes arrondies de distances d'arrondi hétérogènes (cf. figure 4.3.c). Dans ce cas, la limite de l'arrondi ne définit pas une surface fermée et les points d'ancrage disjoints sont reliés à l'aide de segments droits (entre  $A_1$  et  $A_2$ , ou  $A_3$  et  $A_4$  dans l'exemple illustré dans la figure 4.3.c. Ce procédé s'applique à n'importe quelle configuration de sommets, indépendamment de la distance d'arrondi  $d$  des arêtes et de leur convexité (cf. figure 4.4).

La construction ci-dessus définit entièrement la limite des domaines d'arrondi. La surface arrondie sous-jacente peut être définie par n'importe quel processus et primitives d'arrondi (linéaire ou paramétrique). La section 4.3 propose une solution pratique basée sur les surfaces paramétriques.

#### 4.1.2 Structure de détection

La structure définie dans cette section délimite avec précision le domaine arrondi pour toutes les surfaces de la géométrie originale (cf. figure 4.5). Elle correspond aux volumes englobants basés sur des cylindres et des sphères, appelés *volumes de détection*.

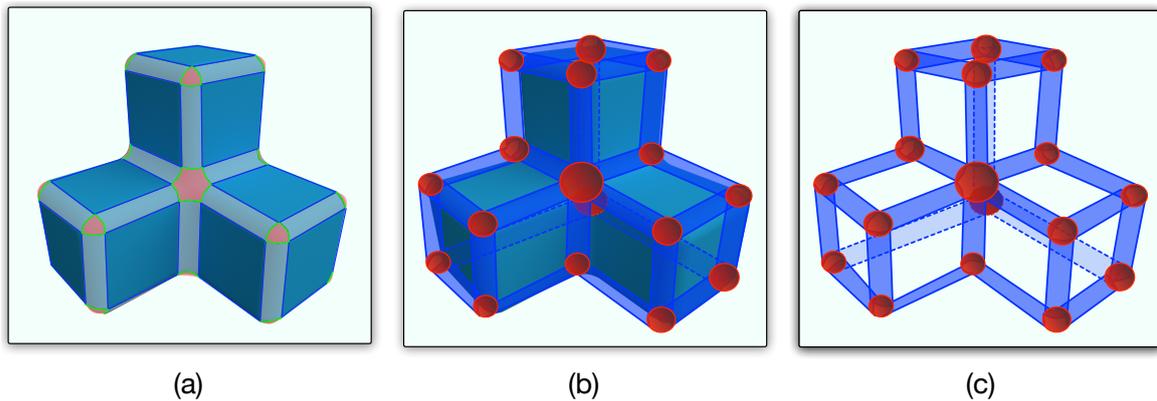


Figure 4.5 – Positionnement de la structure de détection constituée de cylindres et de sphères englobants autour des arrondis. (a) Surfaces arrondies pour les arêtes (bleu clair) et les sommets (orange). (b)-(c) Les cylindres (bleu) et les sphères (rouge) de détection sont placés autour des arêtes et des sommets, entourant exactement les surfaces arrondies..

Chaque arête chanfreinée est délimitée par un cylindre dont l'axe est parallèle à l'arête. Le cylindre délimite précisément le domaine arrondi sur les deux faces adjacentes (c'est-à-dire qu'il définit exactement la limite de l'arrondi sur chaque face impactée par le processus d'arrondi). Les figures 4.6.b et 4.6.c illustrent le positionnement d'un cylindre de détection associé à une arête. Il est basé sur le vecteur de mi-distance  $\vec{H} = (\vec{F}_a + \vec{F}_b) / \|\vec{F}_a + \vec{F}_b\|$  et sur le rayon du cylindre de détection  $r$  tel que  $r = \frac{d}{2 \cos(\theta/2)}$ , avec  $\theta$  l'angle entre les deux faces adjacentes, et  $d$  la distance d'arrondi (définie par l'utilisateur). Les points  $P_1 = v_1 + \vec{H} \cdot 2r$  et  $P_2 = v_2 + \vec{H} \cdot 2r$  correspondent à la limite de l'axe central du cylindre de détection.

Chaque sommet arrondi est délimité par une sphère, dont le barycentre est le centre de masse des points d'ancrage du sommet associé. Le rayon de la sphère est défini comme la distance maximale entre ce centre et les points d'ancrage (cf. figure 4.6.d et e), de manière à englober entièrement la surface du sommet arrondi.

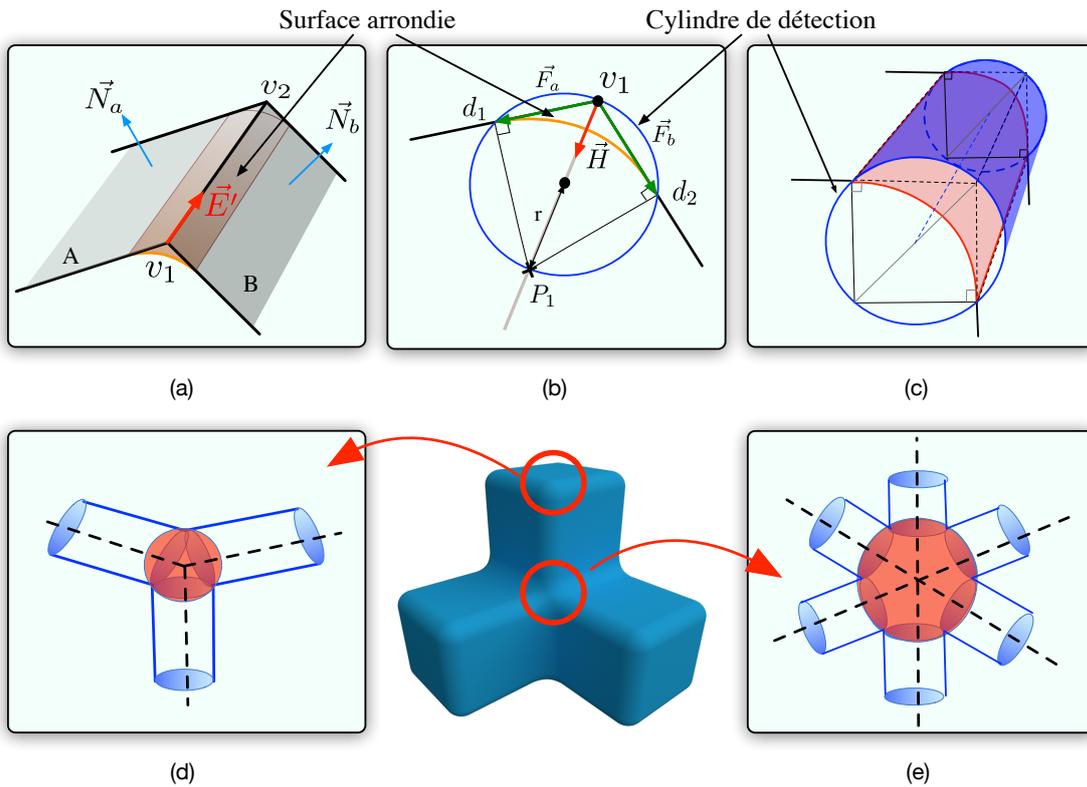


Figure 4.6 – Positionnement des primitives de détection. (a) Surface arrondie pour une arête. (b) Cylindre de détection (en bleu) de rayon  $r$  passe par les points  $v_1$ ,  $d_1$ , et  $d_2$ . (c) Le cylindre délimite exactement l'arrondi de l'arête. (d) Un sommet arrondi est délimité par une sphère (en rouge) et relié à ses cylindres de détection incidents (en bleu). (e) Le même processus est appliqué aux sommets sans qu'il soit nécessaire de distinguer les surfaces convexes et concaves correspondant aux arêtes arrondies incidentes.

Comme nous visons à gérer des scènes comportant un grand nombre de chanfreins, nous avons choisi une organisation des primitives de détection basée sur un Kd-tree. Cette structure s'est avérée expérimentalement plus efficace qu'une hiérarchie de volumes englobants (BVH). Les feuilles de notre Kd-tree contiennent l'ensemble des primitives de détection (cylindres et sphères). Il est inséré dans la structure SREC-RT et est construit pour chaque objet ou groupe d'objets simples, de manière à bénéficier de profondeurs d'arbre modérées.

### 4.1.3 Structure Topologique

Pour un sommet  $C$  donné, le volume de détection est une sphère, et plusieurs cylindres de détection sont attachés pour gérer les arêtes incidentes  $E$ . Lorsqu'un rayon sort d'un volume de détection, son chemin doit être prolongé efficacement, soit à travers le volume de détection suivant, soit à l'extérieur de l'objet. Pour traiter la propagation des rayons, un graphe représentant les adjacences et les incidences entre les volumes de détection est directement déduit des connexions entre les arêtes et les sommets. Il définit la structure topologique employée lors du parcours des rayons, pour traverser les volumes de détection successifs (ce qui est fréquent dans le traitement des rayons rasants). Pour déterminer la propagation d'un rayon qui traverse un volume de détection, nous avons choisi de définir des interfaces planes (désignées par  $I$ ) entre les volumes de détection. Une arête est liée à une interface à chacune de ses deux extrémités. Un sommet est lié à plusieurs interfaces, et dans la plupart des cas le nombre d'interfaces est égal au nombre d'arêtes qui lui sont incidentes. Une interface est reliée (selon la configuration géométrique) soit à deux volumes de détection, soit à un volume de détection et une sortie. La figure 4.7 illustre la configuration associée à un ensemble des arêtes et de sommets, ainsi que leurs interfaces.

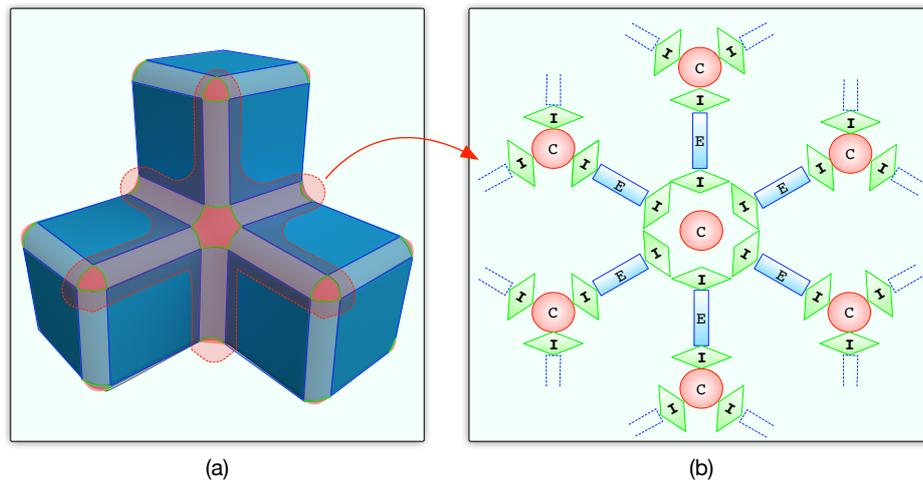


Figure 4.7 – La structure topologique correspondant aux régions arrondies mises en évidence dans (a) est construite autour des arêtes et des sommets arrondis de l’objet. (b) Chaque arête  $E$  (en bleu) est reliée à deux sommets  $C$  (en rouge) par des interfaces planes  $I$  (en vert).

## 4.2 Utilisation de SREC-RT en lancé de rayons

Notre structure est utilisable avec toutes méthodes de rendu basée sur du lancé de rayons. Les rayons peuvent provenir de l’observateur, de sources lumineuses, de la réflexion ou de la réfraction. Le processus de traversée de SREC-RT est général et peut donc être utilisé pour ces différents cas d’utilisation.

### 4.2.1 Processus détaillé

Pour un point d’intersection donné  $P_i$  sur la géométrie originale, la structure de détection permet de déterminer s’il appartient au domaine arrondi (recherche dans le Kd-tree de l’objet intersecté). La figure 4.8.a illustre les différents cas. Si l’intersection n’est pas dans une primitive de détection (comme pour  $P_a$ ), le rayon est directement utilisé dans le processus d’illumination. Sinon, l’intersection se trouve à l’intérieur d’au moins une primitive de détection (comme pour  $P_b$ ) et la surface arrondie sous-jacente associée est utilisée pour calculer l’intersection des rayons.

Pour un point d’intersection  $P'$  sur une surface arrondie (cf. figure 4.8.b), la normale associée est calculée pour poursuivre les processus de rendu.  $P'$  étant placé sous la surface de l’objet, les rayons secondaires doivent partir du maillage de la géométrie d’origine au lieu de  $P'$ . Ils sont d’abord définis par leur origine  $P'$  et une direction (réfléchi, réfracté, ombre, etc.) fixée par le tracé de chemins, mais tracés à partir d’une origine située sur la surface du maillage ( $P''$  ou  $P'''$  sur la figure).

### 4.2.2 Utilisation de la structure topologique de détection

L’avantage d’utiliser la structure SREC-RT réside dans le fait que la plupart des intersections sont en pratique calculées sur la géométrie d’origine de la scène (sans les arrondis). La complexité géométrique associée aux surfaces arrondies n’est pas prise en compte dans la structure accélératrice initiale. Une seule consultation du Kd-tree est effectuée pour un point d’intersection sur les objets arrondis. Nous verrons dans la section 4.4 que dans le cas de détails fins, seuls quelques rayons sont finalement impactés par la traversée de la structure topologique. Mais cette gestion entraîne un gain substantiel en temps de traitement.

La traversé d'un rayon pour une primitive de détection donnée n'entraîne pas systématiquement d'intersection avec la surface d'arrondi sous-jacente. Dans ce cas, la structure topologique est utilisée pour propager efficacement le rayon, sans nouvelle recherche dans le Kd-tree (cf. figure 4.9). L'algorithme consiste à estimer quel plan d'interface correspond au changement de volume de détection (en utilisant l'intersection rayon-plan); ainsi, la structure topologique permet de directement sélectionner la prochaine primitive de détection à visiter. De même, si le rayon ne croise aucune surface arrondie, les plans d'interface permettent de déterminer le point de sortie (comme pour le rayon  $ray_b$  dans les figures 4.9.a et 4.9.b).

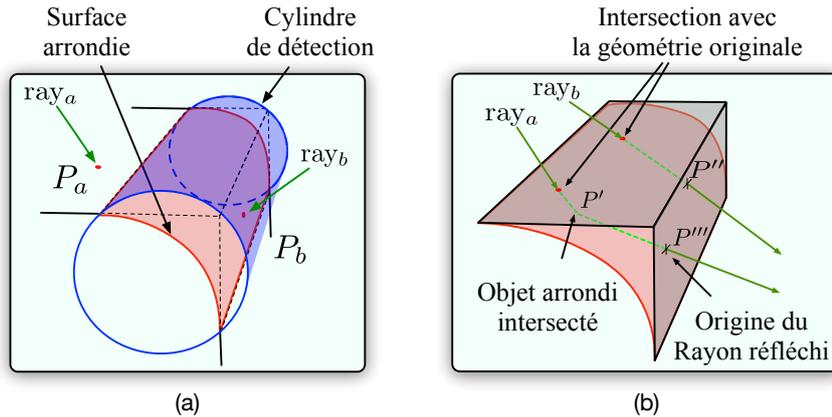


Figure 4.8 – Intersection rayon-objet utilisant le maillage original. (a) Avec le rayon $_a$ , l'intersection  $P_a$  est située sur la géométrie d'origine et en dehors de la structure de détection. Le processus de traçage de rayons est effectué normalement. Avec le rayon $_b$ , l'intersection  $P_b$  se trouve à l'intérieur d'un cylindre de détection et le test d'intersection est effectué avec la surface arrondie. (b) Le rayon $_a$  intersecte la surface arrondie en  $P'$ , et l'origine du rayon réfléchi est fixée à  $P'''$  sur le maillage original. Avec le rayon $_b$ , aucune intersection n'est trouvée sur la surface arrondie sous-jacente, et l'origine du rayon $_b$  est déplacée à  $P'''$ .

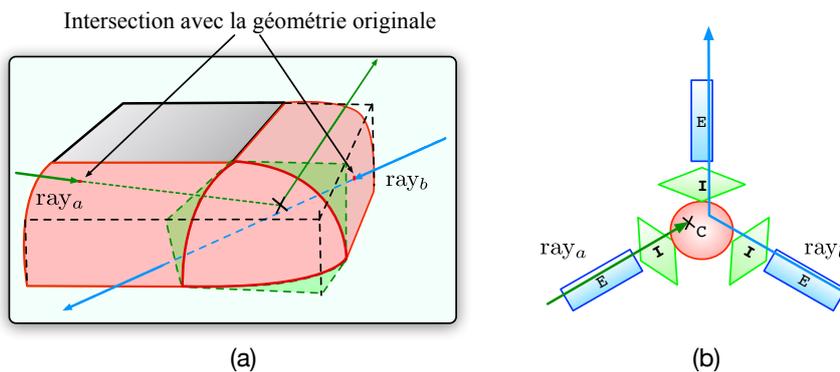


Figure 4.9 – Propagation du rayon au travers de la structure de détection grâce aux graphes topologiques. (a)  $ray_a$  ne touche pas la surface arrondie dans le premier cylindre de détection. La structure topologique guide la recherche dans le volume de détection suivant (la sphère d'angle), et une intersection est trouvée sur l'angle arrondi. (b) Vue de la structure avec le parcours schématisé des rayons : le rayon $_a$  touche un coin arrondi à C, tandis que le rayon $_b$  traverse différentes primitives de détection sans toucher la géométrie arrondie.

### 4.2.3 Cas particulier : sommets concaves

Dans le cas des arrondis concaves, la géométrie d'origine n'englobe pas la surface de l'arrondi, et les rayons rasants peuvent manquer l'intersection (cf. figure 4.10.a). Cette configuration est gérée dans SREC-RT avec des *triangles portails* placés aux interfaces entre les jonctions arêtes-sommets concaves, de manière à définir un point d'entrée dans la structure de détection. Un triangle est construit pour chaque côté de bord arrondi, à partir de la courbure. Ces triangles sont ajoutés à la structure générale d'accélération du rendu. Lorsqu'un rayon frappe l'un de ces triangles, SREC-RT est interrogé sans traitement particulier. Les portails de triangles définissent de nouveaux points d'entrée dans SREC-RT pour garantir que les rayons ne manquent jamais une géométrie arrondie.

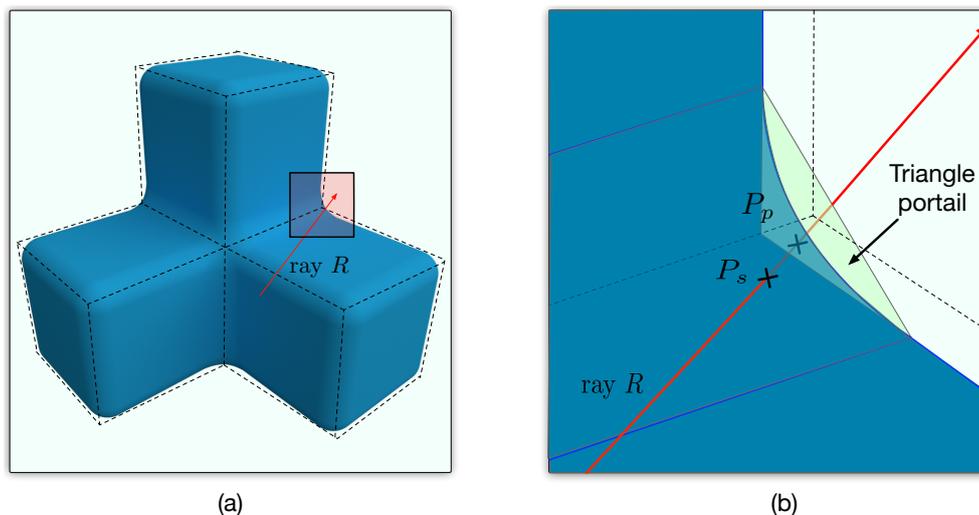


Figure 4.10 – Les rayons rasants peuvent *manquer* un chanfrein concave. (a) Un rayon  $R$  n'intersecte pas la géométrie d'origine, car il manque la zone d'arrondi qui n'est pas incluse par la géométrie d'origine, il n'explore donc pas SREC-RT. (b) Un portail triangulaire est ajouté à chaque coin concave, car les interfaces qui produisent des points  $P_p$  activent l'exploration dans SREC-RT, pour déterminer le point d'intersection exact  $P_s$ .

Remarque : les triangles portails sont considérés comme invisibles et ne modifient pas les maillages d'origine. Ils ajoutent des informations géométriques sur un petit ensemble de configurations spécifiques de coins concaves afin de définir un point d'entrée dans SREC-RT. Dans ce cas, l'étape de détection est évitée et la surface arrondie peut être directement intersectée.

## 4.3 Représentation des arrondis

SREC-RT est conçu indépendamment de la représentation des surfaces arrondies choisie. Idéalement, pour un rendu visuel de qualité, une représentation compacte de ces surfaces est préférable, ainsi qu'une continuité  $G^1$  entre la géométrie arrondie et la géométrie d'origine. Il est également nécessaire que la représentation de surface choisie soit facilement utilisable pour les méthodes de tracé de chemins, sans discontinuité, pour tout point de vue et toute configuration d'éclairage.

Les opérations d'arrondi conduisent souvent à des ensembles complexes de surfaces courbes, généralement approchées par des méthodes de subdivision (par exemple, Catmull-Clark [30]). Avec cette approche, la subdivision la plus fine (définie arbitrairement) résulte en un maillage polygonal qui est gourmand en mémoire.

### 4.3.1 Représentation et intersection des arêtes arrondies

Comme cela sera présenté dans la section 4.4, l'utilisation de répartition hiérarchique de SREC-RT permet de réduire le temps de traitement. Nous avons donc choisi de mettre à contribution ce gain de temps pour travailler sur l'amélioration visuelle et la précision dans les calculs de simulation. Nous proposons dans cette section une représentation basée sur les surfaces paramétriques (minimisant ainsi la consommation de mémoire). Elle s'appuie sur notre représentation topologique où les sommets arrondis, délimités par des sphères de détection, sont représentés par un ensemble minimal de patches triangulaires de Gregory modifiés [65]. La continuité  $G^0$  est assurée entre les chanfreins des arêtes et des sommets grâce à des patches basés sur des courbes de Bézier quadratiques extrudées et délimitées par deux courbes identiques. La surface de l'arrondi correspond donc à l'extrusion d'une courbe de Bézier le long de l'axe de l'arête (comme illustré sur la figure 4.11). Cette formulation nous permet de contrôler l'arrondi de l'arête avec un minimum de paramètres, tout en assurant la continuité avec les faces adjacentes et les sommets arrondis connectés.

Les patches ne sont pas approximés par des mailles polygonales ; ils sont directement utilisés par la procédure de lancé de rayons, offrant une qualité de lissage constante quel que soit le point de vue. Pour l'intersection des rayons, une première intersection 2D est calculée entre le rayon et la courbe de Bézier, fournissant le plan tangent 3D où l'intersection finale est déterminée. Le rayon intersecte l'angle arrondi si et seulement si le point d'intersection se trouve entre ses deux extrémités.

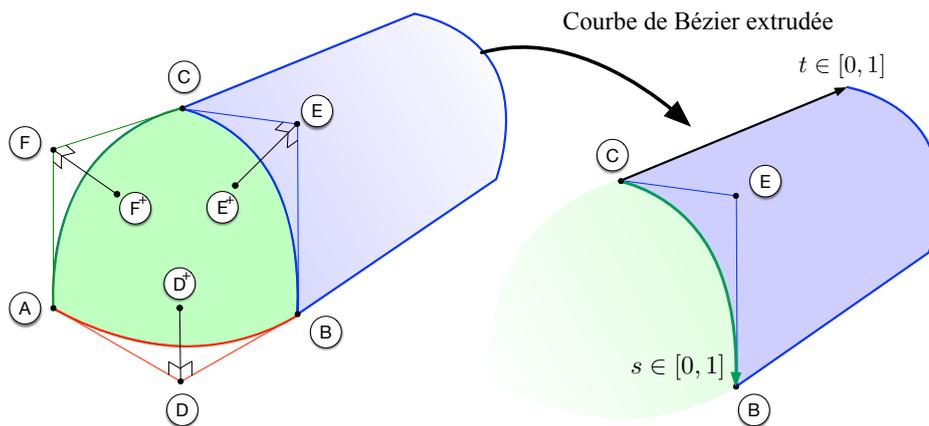


Figure 4.11 – Les sommets (à gauche) sont représentés par des patches triangulaires. Les limites sont définies par trois courbes de Bézier ( $ADB$ ,  $BEC$  et  $CFA$ ), et la forme globale est contrôlée par trois points intérieurs ( $D^+$ ,  $E^+$  et  $F^+$ ). Les arêtes (à droite) sont représentées par des courbes de Bézier linéairement extrudées.

### 4.3.2 Représentation et intersection des sommets arrondis

Représenter des surfaces courbes avec un ensemble de patches de Bézier est une approche courante [29]. Bien que des patches cubiques (ou de degré supérieur) puissent être utilisés pour contrôler la courbure d'une surface, les tests d'intersection entre les rayons et les éléments paramétriques conduisent à des calculs longs [88, 158]. De plus, pour un meilleur contrôle de la courbure il serait nécessaire d'augmenter soit le degré des patches, soit le nombre de points de contrôle intérieurs comme cela est fait pour les patches de Gregory [65]. Malheureusement, ces deux solutions ont un impact sur les besoins en mémoire. C'est pourquoi, nous proposons d'utiliser des patches triangulaires quadratiques. La représentation par patches qui en résulte permet une utilisation directe dans les systèmes de rendu par rayons ainsi qu'un contrôle fin de la continuité entre les arêtes et les sommets.

Soit un patch triangulaire  $ABC$  dont les bords sont définis par trois courbes de Bézier  $ADB$ ,  $BEC$  et  $CFA$ , comme illustré sur la figure 4.11, le patch triangulaire est défini comme une fonction à deux paramètres  $X$  :

$$\begin{aligned} X(s, t) = & (1 - s - t)^2 A + s^2 C + t^2 B \\ & + 2s(1 - s - t) ((1 - t)F + tF^+) \\ & + 2t(1 - s - t) ((1 - s)D + sD^+) \\ & + 2st ((s + t)E + (1 - s - t)E^+) \end{aligned} \quad (4.1)$$

ou  $s, t \in [0, 1]$  et  $s + t \leq 1$ . Les points  $D^+$ ,  $E^+$ , et  $F^+$  sont des points de contrôle intérieurs.

Pour un sommet relié à  $n = 3$  arêtes, la définition d'un patch est directement utilisée pour décrire la surface arrondie. Dans les cas généraux (sommets arrondis avec plus de 3 arêtes incidentes i.e.  $n > 3$ ), le patch polygonal est délimité par  $n$  courbes de Bézier. Dans la suite, une telle géométrie est appelée un domaine à  $n$  côtés. Notre approche consiste à trianguler le domaine à partir d'un point central, selon une version modifiée de la méthode de remplissage de régions [128, 146]. Cette opération construit  $n$  patches triangulaires. La figure 4.12 présente cette construction pour un domaine à 4 côtés.

Étant donné un sommet  $V$  de la géométrie d'entrée, le point central de la subdivision  $P_c$  est défini comme suit :

$$P_c = \tilde{P}_c + \lambda(V - \tilde{P}_c), \quad (4.2)$$

où  $\tilde{P}_c = \frac{1}{n} \sum_{i=0}^{n-1} P_i$  le centre de masse des points de contrôle de la courbe de Bézier environnante. Le scalaire  $\lambda \in [0, 1]$  définit l'élévation de  $P_c$ , et nous permet de contrôler la netteté du domaine à  $n$  côtés. La valeur de  $\lambda$  est choisie par l'utilisateur (nous avons fixé  $\lambda = 0,4$  dans toutes nos expériences). La normale  $\tilde{N}$  du plan tangent à la surface au point central  $P_c$  est définie par :

$$\tilde{N} = \left| \frac{1}{n} \sum_{i=0}^{n-1} (P_i - \tilde{P}) \times (P_{i+1} - \tilde{P}) \right|.$$

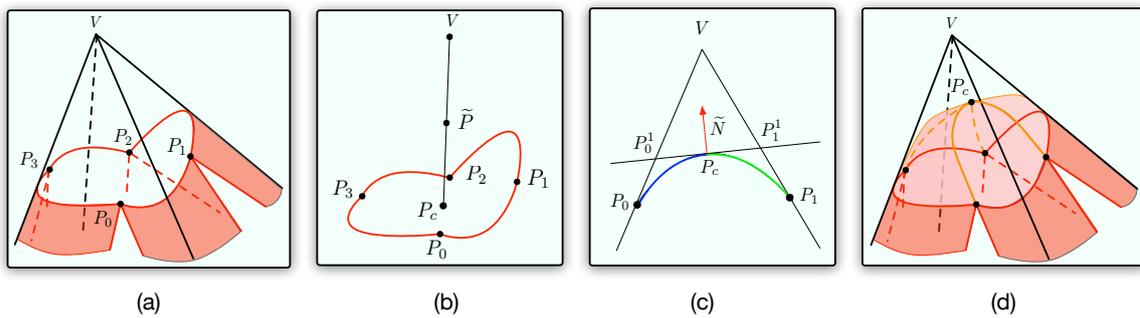


Figure 4.12 – Construction d'un arrondi correspondant à un sommet  $V$  sur la géométrie originale avec quatre courbes de Bézier (a). (b) Le point central  $P_c$  est calculé à partir de  $V$  et des points  $P_i$  (points de contrôle des courbes de Bézier). (c) Les courbes de Bézier intermédiaires sont déterminées grâce au plan défini par la normale  $\tilde{N}$  et le segment  $P_i V$ . (d) L'arrondi du sommet est un ensemble de patches de Bézier triangulaires.

Le calcul de l'intersection des rayons que nous employons pour les patches de Bézier est basé sur la méthode de Kajiya [88]. Notre algorithme teste différentes configurations et sauvegarde celle qui présente la plus petite erreur. Les détails mathématiques sont fournis dans notre article [36].

## 4.4 Résultats

Les résultats présentés dans cette section ont été produits sur un ordinateur équipé d'un processeur Intel Xeon(R) E5-1620 v3 3.50 GHz, avec 8 threads et 48 Go de RAM. Le code a été intégré à *Blender* [20] avec le moteur de rendu *Cycles* [21], afin de fournir des comparaisons justes avec les modificateurs d'arrondi existants et avec *PBRT* [127]. Toutes les images et statistiques ont été produites à une définition de  $1920 \times 1080$ , avec 512 rayons par pixel.

Une structure SREC-RT est construite par objet afin de limiter la profondeur du Kd-tree associé. La construction de la structure SREC-RT nécessite quelques dizaines de millisecondes (150ms pour les scènes les plus complexes). Les primitives de détection ont une faible empreinte mémoire avec : 7 floats / cylindre, 4 floats / sphère, 12 floats / surface paramétrique des arêtes, et 36 floats / patches de Bézier (équivalent à 2 niveaux d'un maillage de subdivision).

Les résultats suivants illustrent les comparaisons entre SREC-RT et plusieurs autres méthodes : sans arrondi et avec des mailles de subdivision géométrique. Pour des comparaisons équitables, les mailles sont calculées à partir de notre implémentation de surface paramétrique (cf. section 4.3), avec différents niveaux de subdivision  $n$ . Lorsque  $n$  tend vers l'infini, les mailles ont tendance à s'approcher très près de notre représentation paramétrique. Nous avons choisi la subdivision  $n = 1$  (2 triangles) qui conduit à un stockage mémoire égal, et la subdivision  $n = 5$  (10 triangles) qui est le niveau minimum requis pour produire un bon compromis en matière de qualité visuelle pour les points de vue choisis.

### 4.4.1 Temps de rendu et impact mémoire

Le tableau 4.1 présente les statistiques de consommation en mémoire et les temps de calcul pour l'ensemble de scènes présenté dans la figure 4.13. Le coût mémoire de la structure SREC-RT comprend les paramètres des surfaces paramétriques, les primitives de détection, la représentation topologique et la structure accélératrice Kd-tree. Dans ce tableau, la colonne *No rounded* correspond à des rendus de la géométrie d'entrée originale (sans arrondi). Les besoins en mémoire ne sont que légèrement affectés par SREC-RT (et sont approximativement équivalents aux besoins associés à un niveau de subdivision de 1). Une qualité de rendu équivalente à celle de SREC-RT nécessite un niveau de subdivision minimum de 5, ce qui augmente fortement la consommation de mémoire, et sans aucune garantie que la qualité visuelle soit constante lorsque le point de vue ou les sources de lumière se déplacent.

	Blocks			Piano			Bedroom				Kitchen		
	# tri	time	MB	# tri	time	MB	# tri	time 1	time 2	GB	# tri	time	MB
No rounded	42k	244s	72,1	5k	475s	95	541k	1276s	1224s	0.27	511k	1241s	311
<b>SREC-RT</b>	<b>42k</b>	<b>395s</b>	<b>92</b>	<b>5k</b>	<b>550s</b>	<b>96</b>	<b>541k</b>	<b>1457s</b>	<b>1442s</b>	<b>0.65</b>	<b>511k</b>	<b>1502s</b>	<b>341</b>
5 sub.	1M	276s	348	107k	485s	123	23M	1328s	1248s	6.54	953k	1257s	438
1 sub.	63k	252s	93	13k	488s	97	2M	1260s	1198	0.68	540k	1252s	321

TABLE 4.1 – Statistiques de rendu pour chaque scène illustrée dans la figure 4.13 avec, le nombre de triangles, le temps de rendu et l'utilisation de la mémoire. Pour la scène de la chambre à coucher, les temps 1 et 2 correspondent à deux points de vue différents. L'utilisation de SREC-RT est comparée aux méthodes basées sur le maillage avec un niveau de subdivision de 5 (qualité visuelle égale) et un niveau de subdivision de 1 (mémoire équivalente).

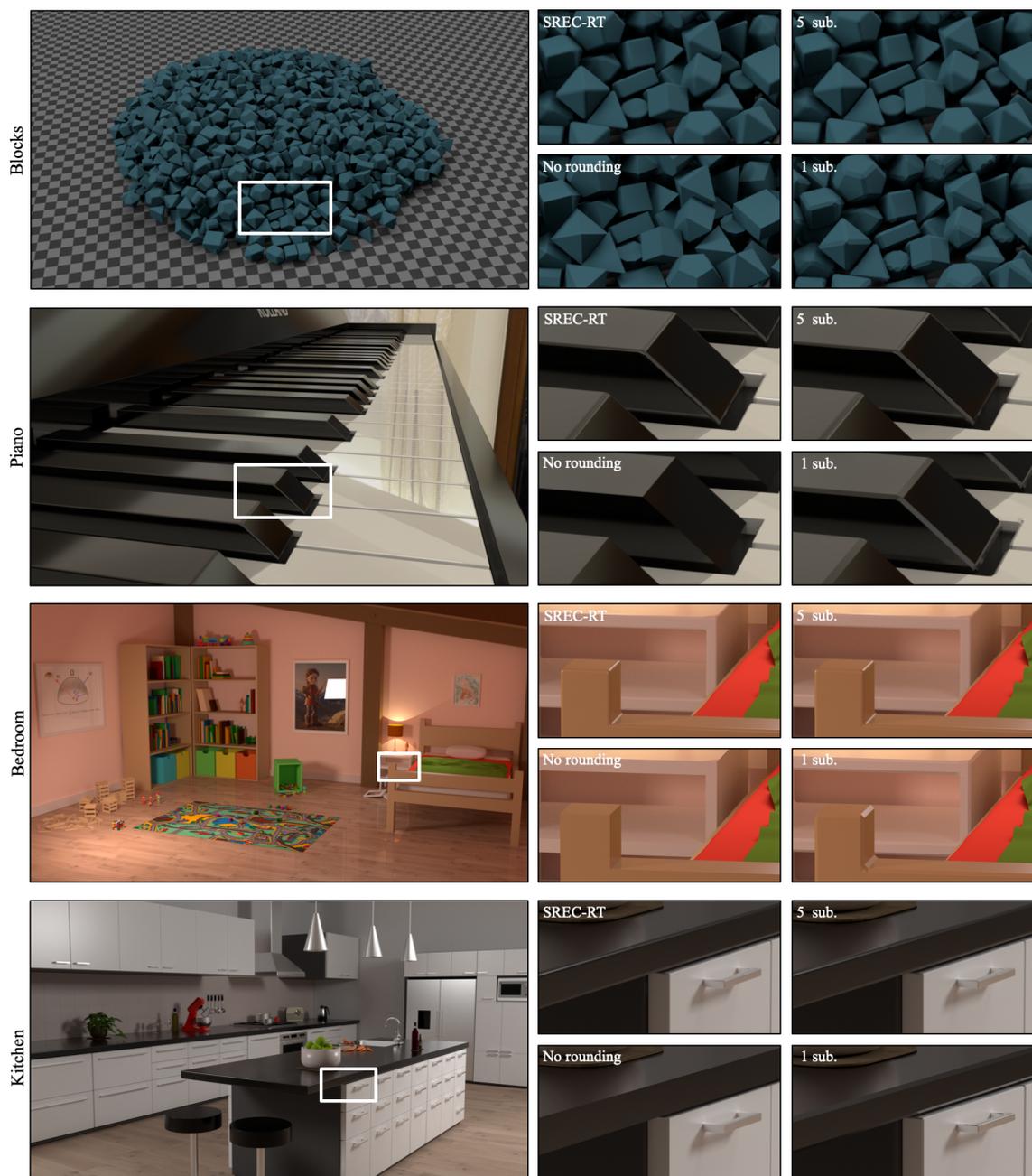


Figure 4.13 – Diverses configurations de scènes illustrant des bords et des sommets arrondis. SREC-RT donne toujours une qualité de rendu constante, indépendamment du point de vue ou de la configuration d’éclairage. Pour obtenir une qualité de rendu équivalente avec des maillages de subdivision, le nombre de subdivisions doit être au moins de 5 ce qui augmente considérablement la quantité de mémoire nécessaire (Tableau 4.1).

L’analyse des temps de traitement du tableau 4.1 met en évidence le fait que le temps de calcul dépend fortement du nombre de rayons qui atteignent les surfaces arrondies. Autrement dit, le temps de calcul peut augmenter différemment en fonction du nombre d’arrondis visibles à partir du point de vue. La scène Blocks représente la pire configuration pour notre méthode car les objets ont été fortement arrondis, et sont concentrés dans la même région. Avec le point de vue choisi, le nombre de rayons affectés par la traversée d’une structure SREC-RT représente 10 % de tous les rayons émis lors du rendu (rayons primaires et secondaires).

L'utilisation de la structure SREC-RT évite de nombreux tests d'intersection entre les rayons et les surfaces paramétriques. Le temps de rendu diminue significativement, la figure 4.14 illustre cette analyse effectuée pour une séquence d'images (équivalente à une video de survol de la scène). Pour la plupart des images, SREC-RT réduit l'impact des tests d'intersection sur les surfaces paramétriques (cf. figure 4.14.a), en particulier lorsque des bords arrondis correspondent à de longues arêtes (cf. figure 4.14.b). Pour les points de vue rapprochés, où les arrondis sont concentrés dans l'image, SREC-RT n'offre pas de réel avantage par rapport au rendu direct des surfaces paramétriques, mais les temps restent équivalents (cf. figure 4.14.c).

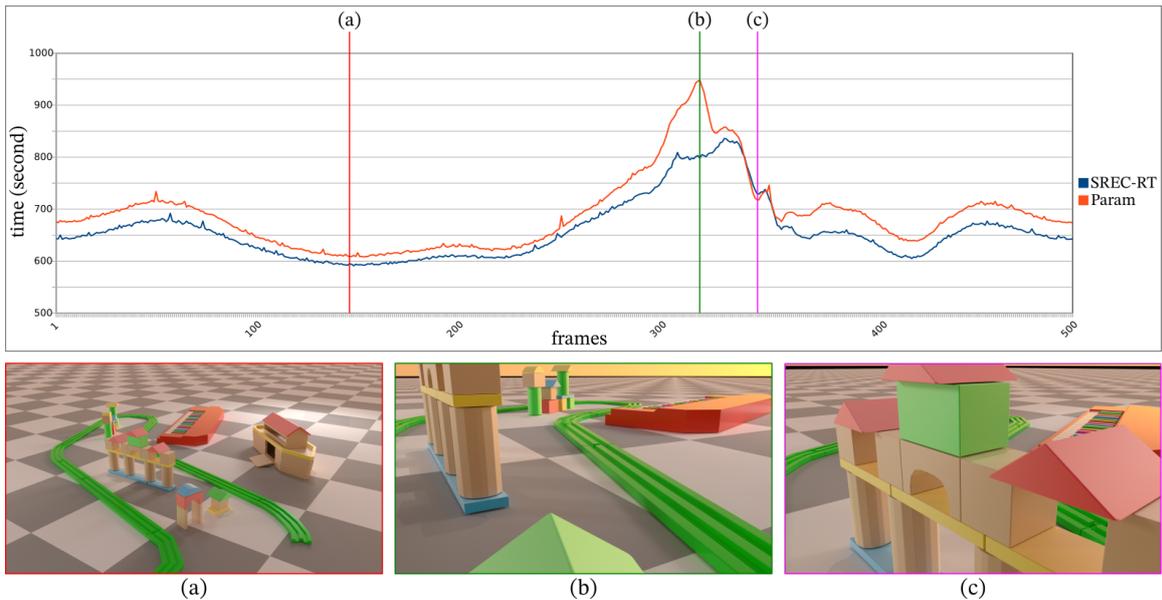


Figure 4.14 – En haut : Temps de rendu pour chaque image. Le temps de rendu pour l'intersection directe de surfaces paramétriques correspond à la courbe orange. Les surfaces sont considérées comme des primitives géométriques (triangles dans ce cas), ajoutées à la structure d'accélération du maillage. Toutes les surfaces sont intersectées comme des primitives géométriques classiques. Les intersections avec des surfaces gérées par notre structure supplémentaire, utilisant SREC-RT, suivent la courbe bleue. En bas : points de vues sélectionnés dans la vidéo : (a) Perspective large, (b) Plusieurs bords longs, et (c) Haute concentration de bords arrondis.

#### 4.4.2 Comparaison visuelle des arrondis

Nous avons expliqué en début de ce chapitre l'importance d'un rendu visuel de qualité pour les arrondis dans la visualisation de scènes 3D (reflets, direction des rayons incidents, distinction des silhouettes, etc.). Dans la figure 4.15, les formes de surbrillance observées sur l'objet dépendent clairement du nombre de subdivisions (ici 1, 4, ou 10). Produire un résultat proche de notre méthode avec des maillages de subdivision nécessite un niveau de subdivision élevé, et les changements de configurations d'éclairage conduisent inévitablement à des surbrillances incohérentes et à des artefacts temporels.

La gestion correcte des arêtes et des sommets arrondis est également très importante pour le rendu de caustiques précises (cf. figure 4.16). Un nombre trop faible de subdivisions a tendance à créer des caustiques en forme d'étoile à cause des orientations discontinues des faces du maillage. Cela ne permet pas de représenter l'aspect lisse des caustiques qui devrait apparaître, contrairement à notre représentation paramétrique de forme arrondie.

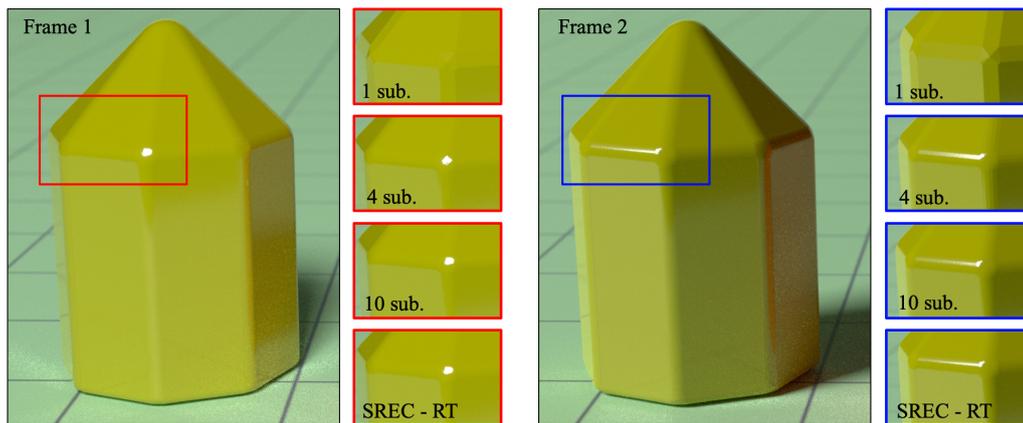


Figure 4.15 – Deux images d’une animation d’éclairage, où les surbrillances sur les arrondis se déplacent. Les surbrillances sont cohérentes avec la méthode SREC-RT, contrairement à la méthode de subdivision. Les formes de surbrillance sont sujettes à un scintillement temporel et à des artefacts qui peuvent tous deux être réduits, mais au prix d’un niveau de subdivision accru.

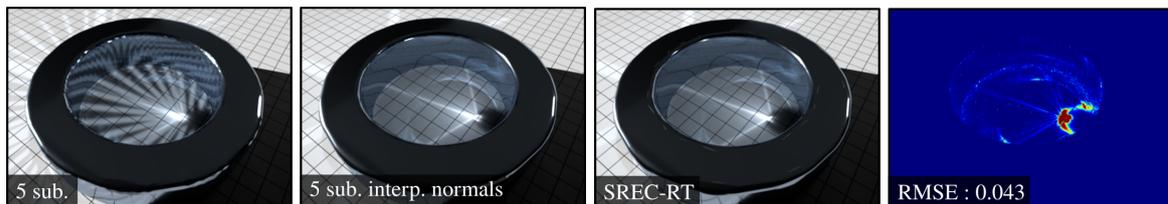


Figure 4.16 – La précision du calcul des caustiques dépend fortement du nombre de subdivisions utilisé pour représenter les bords arrondis. Avec SREC-RT et notre représentation paramétrique de la surface, les caustiques calculées sont plus lisses qu’avec 5 de niveaux de subdivision. Bien que l’interpolation normale corrige l’apparence des caustiques, les normales interpolées restent différentes de la surface paramétrique réelle, comme l’illustre l’image en fausses couleurs qui affiche la différence absolue entre l’image SREC-RT et l’image 5 sub. avec interpolation des normales.

#### 4.4.3 Interpolation des normales

Lors de la visualisation d’un maillage discrétisé, les calculs des ombres et des reflets lumineux ne permettent pas d’obtenir directement un aspect lisse des arrondis. Pour palier ce problème, une technique classique consiste à interpoler les normales de surface sur des polygones et calculer les couleurs des pixels en fonction des normales interpolées et d’un modèle de réflexion. Même si cette solution permet d’obtenir un aspect plus lisse et visuellement satisfaisant, l’utilisation de l’interpolation peut faire apparaître des artefacts visuels (cf. figure 4.17).

Ces incohérences sont dues à une mauvaise estimation de la normale. En effet, celles-ci sont calculées à partir des surfaces de subdivision qui ne sont qu’une approximation de l’arrondi souhaité. Des solutions intéressantes ont été proposées pour réduire ces artefacts dans les cas spécifiques du bump mapping et de l’interpolation normale, mais les auteurs soulignent toujours les limites dues à l’interpolation [138, 144]. En effet, le lissage des normales n’évite pas les discontinuités observées au niveau des silhouettes d’un objet. En outre, les maillages présentant des niveaux de subdivision même élevés peuvent encore être indirectement visibles, par exemple dans le cas d’ombres générées par un éclairage à angle rasant, comme l’illustre la figure 4.18. L’utilisation de surfaces paramétriques garantit que la géométrie et les normales associées aux arrondis sont réellement lisses et conformes aux points d’intersection des rayons, ce qui assure la cohérence entre la forme de la surface et son ombrage.

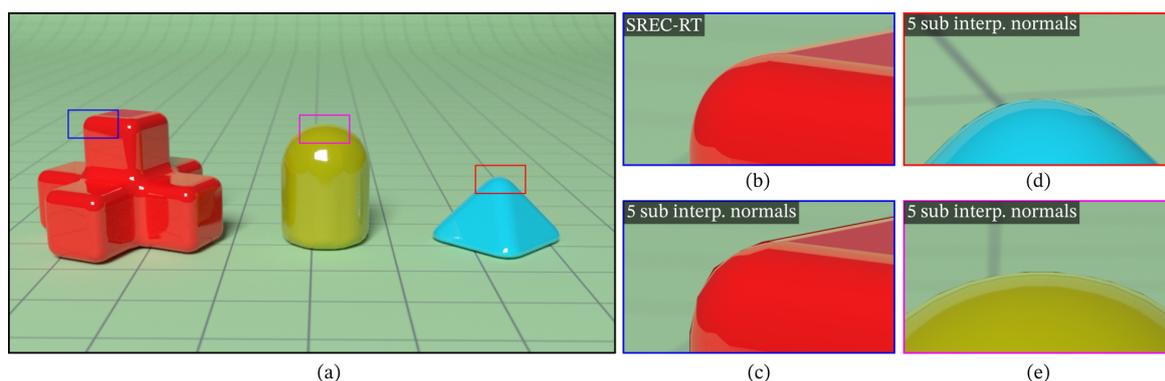


Figure 4.17 – Objets arrondis avec grossissement sur les bords arrondis. (a,b) Avec SREC-RT. (c,d,e) Avec 5 subdivisions géométriques utilisant l'interpolation normale. L'interpolation souffre d'artefacts au niveau des silhouettes des objets. Des différences peuvent être observées dans les reflets lumineux en raison des incohérences entre le point d'intersection et les normales interpolées.

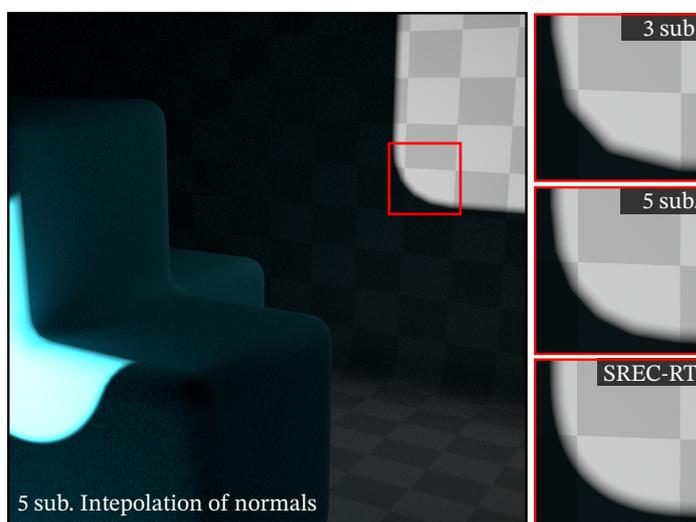


Figure 4.18 – Les méthodes de subdivision peuvent conduire à une discrétisation des silhouettes au niveau des ombres. Contrairement à SREC-RT, l'utilisation de subdivision lors du calcul des ombres laisse apparaître la discrétisation du maillage. L'interpolation des normales ne corrige pas ce problème.

## 4.5 Conclusion

Nous avons présenté dans ce chapitre une structure topologique et hiérarchique appelée SREC-RT, conçue pour améliorer la visualisation des bords arrondis avec une application de lancé de rayons. SREC-RT est construite automatiquement à partir de la géométrie d'une scène 3D pour les objets choisis par l'utilisateur. L'utilisation de cette structure n'est pas invasive, la scène originale est conservée et les paramètres de chanfreins sont manipulés de manière indépendante.

SREC-RT repose sur une structure de détection qui permet de localiser efficacement les zones concernées par un arrondi. Elle peut donc être considérée comme une structure accélératrice hiérarchique qui permet de déterminer efficacement si un point d'intersection est situé sur un bord arrondi ou un coin arrondi. Elle n'est pas limitée aux configurations géométriques simples et peut être appliquée à tout objet, indépendamment de sa convexité. Les liens topologiques que comporte la structure sont utilisés pour calculer le parcours des rayons au travers de la structure de détection. Cet atout majeur permet de localiser les traitements puisqu'il évite un nouveau parcours global de la structure dans le cas des rayons rasants.

Le choix de représentation géométrique des arrondis reste très souple : subdivision plus ou moins dense, surfaces paramétriques, maillages, etc. Pour améliorer la qualité visuelle des arrondis, nous avons proposé une solution basée sur des patches triangulaires quadratiques de Bézier pour les coins et des courbes de Bézier extrudées pour les bords. Cette représentation ne nécessite qu'une faible empreinte mémoire par rapport aux maillages de subdivision.

Par rapport aux méthodes d'arrondi basées sur le maillage, nos résultats montrent que SREC-RT est drastiquement moins gourmand en mémoire avec seulement un faible surcoût en de temps de calcul. Enfin, elle assure une qualité constante, indépendamment du point de vue choisi et/ou de la configuration d'éclairage.

Les perspectives directes de ces travaux concernent l'utilisation de SRECT-RT de manière plus judicieuse pour par exemple guider les échantillonnages réalisés au cours du processus de lancé de rayons : longueur des chemins de lumière, distance du point de vue, empreinte pixel de l'arrondi. Ces différents points seront détaillés dans le chapitre 6.

## Publications relatives au chapitre

### Revue internationale

*SREC-RT : A Structure for Ray Tracing Rounded Edges and Corners.*

S. Courtin, S. Horna, M. Ribardière, P. Poulin, D. Meneveaux. .

Computer Graphics Forum , 2021, pp. Volume 40, pages 200-214.

### Conférences internationales avec comité de lecture

*Efficient Rendering of Rounded Corners and Edges for Convex Objects.*

S. Courtin, S. Horna, M. Ribardière, P. Poulin, D. Meneveaux.

Lecture Notes in Computer Science - Advances in Computer Graphics. Proceedings of CGI 2019. Volume 11542

### Conférences nationales sans comité de lecture

*Arrondis et simulation d'éclairage basée rayons : une structure dédiée aux surfaces implicites.*

S. Courtin, S. Horna, M. Ribardière, P. Poulin, D. Meneveaux .

Journées Françaises de l'informatique Graphique, 2019.Marseille - 2<sup>eme</sup> meilleur papier

### Projets et collaboration

- 2018 : Projet région SIMOREVA avec les universités de Poitiers (XLim), Montréal (LIGUM), l'INRIA, le L2PN de l'Institut d'Optique et le NTNU colorlab en Norvège
- Collaboration avec Pierre Poulin (laboratoire LIGUM de Montréal) dans le cadre de la thèse de Simon Courtin.



# RECONSTRUCTION TOPOLOGIQUE D'IMAGERIE MÉDICALE

## Sommaire

<b>5.1 Correction et reconstruction topologique</b> . . . . .	<b>73</b>
<b>5.2 Reconstruction 3D de données médicales à multi label sémantiques</b> . . . . .	<b>78</b>
<b>5.3 Résultats et comparaisons</b> . . . . .	<b>91</b>
<b>5.4 Conclusion</b> . . . . .	<b>96</b>

De nos jours, les modèles numériques 3D sont des composants clés dans de nombreux secteurs industriels et scientifiques, tels que l'architecture, les géosciences, la simulation et bien sûr l'infographie. De nombreux outils proposent de construire des objets virtuels, soit par construction, soit par reconstruction à partir de données d'acquisition (nuage de points, images, etc.). Cependant, chaque application a ses propres exigences de qualité qui restreignent la classe des modèles acceptables et pris en charge [8, 23]. Par exemple, les modèles utilisés dans le cadre de l'impression 3D doivent avoir une représentation volumique pour estimer l'épaisseur des zones fragiles. En simulation, les relations de voisinage entre volumes sont nécessaires pour calculer les transferts d'énergie (simulation acoustique, thermique ou d'éclairage). En modélisation géométrique et topologique, les opérations complexes comme les opérations booléennes, ou le remplissage de trous nécessitent une description spatiale volumique fermée des objets.

Bien que la modélisation 3D soit de plus en plus courante, les modèles 3D correspondent souvent à une représentation géométrique : soit par un ensemble de faces, soit (dans le meilleur des cas) par un ensemble de volumes. Aucune information topologique supplémentaire n'est disponible, hormis le voisinage frontal des volumes. Ainsi, les objets produits peuvent contenir des incohérences, notamment des interpénétrations entre éléments de même dimension (volume à volume, face à face ou bord à bord). Comme nous l'avons vu dans les chapitres 3 et 4, bien que des bibliothèques de modèles d'objets 3D soient de plus en plus créées/publiées, ces objets ne peuvent pas être utilisés directement. En effet, l'ensemble de primitives géométriques qui les compose ne respecte pas les propriétés d'un complexe linéaire par morceaux (PLC). Une correction géométrique et une reconstruction d'informations dédiées doivent donc être réalisées au préalable pour contrôler la cohérence du modèle. Par exemple, la figure 5.1 représente un ensemble de surfaces géologiques, qui ne peuvent pas être utilisées pour détecter un réservoir de pétrole par simulation d'écoulement, en raison des nombreuses intersections entre les faces. Un

processus dédié doit être utilisé pour construire un modèle valide pour les algorithmes de simulation [24, 71, 126, 174]. Nous avons également vu dans la section 3.4.1 qui traite du lancé de rayons dans une structure de tétraèdres, qu'une CDT ne peut pas être construite à partir de n'importe quel ensemble de faces. Le modèle géométrique d'entrée doit correspondre à un complexe linéaire par morceaux (PLC - c'est à dire une ensemble de faces dépourvue d'intersection). Il est donc indispensable de proposer des méthodes permettant de contrôler et éventuellement corriger les modèles géométriques 3D fournis.

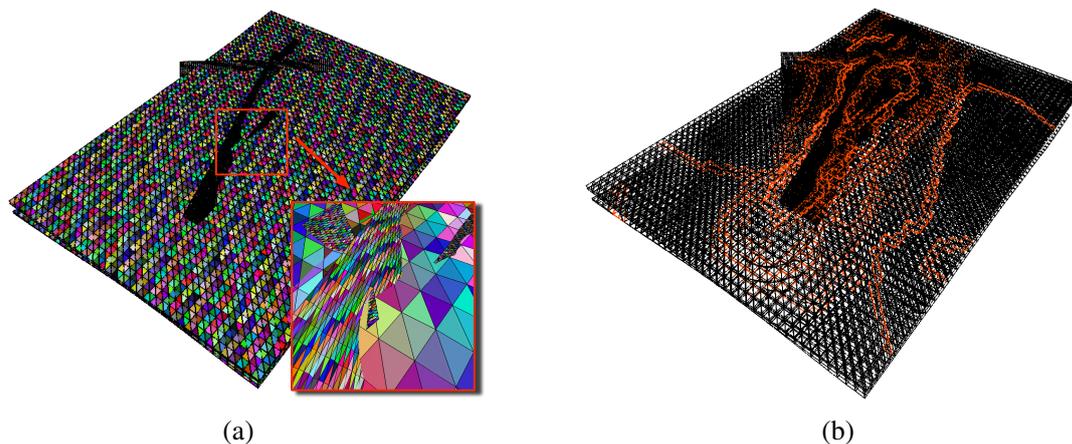


Figure 5.1 – Modèle de réservoir généré à partir de données géologiques (ensemble de failles et d'horizons géologiques) : (a) interprétation directe de l'étude sismique et du forage avec des faces de recouvrement ; (b) détection en rouge de toutes les intersections géométriques qui empêchent la simulation de l'écoulement.

Dans l'axe de recherche présenté dans ce chapitre, nous nous sommes intéressés à la reconstruction des informations topologiques à partir d'un ensemble de polygones. Notre objectif est de produire des modèles contraints par des propriétés géométriques topologiques et sémantiques qui permettent d'une part de contrôler la cohérence des objets, améliorer les processus de reconstruction grâce aux informations topologiques et fournir des modèles utilisables directement pour des méthodes de simulations (sans intersection, caractérisés par les données métiers). Les premiers résultats obtenus nous ont amené à étendre notre travail dans le contexte de la reconstruction de donnée 3D à partir d'acquisitions médicales. Au travers des projets régions IMPROVED (2019) et MEDIA IRM 3D (2020), nous avons collaboré avec des praticiens hospitaliers pour développer des systèmes de reconstruction et de visualisation 3D à partir de données cliniques.

En effet, ces dernières années, les technologies d'imagerie telles que la tomographie par ordinateur (CT) et l'imagerie par résonance magnétique (IRM) se sont répandues et sont désormais couramment utilisées en médecine pour étudier les organes internes. Habituellement, les praticiens basent essentiellement leur analyse sur des coupes 2D, qui ne correspondent qu'à des vues restreintes d'une pathologie. Récemment, les opérations de reconstruction 3D à partir d'acquisitions médicales se sont développées, car elles fournissent une visualisation plus globale des organes, améliorant l'identification de leurs composants et permettent d'effectuer des analyses supplémentaires de l'activité fonctionnelle anatomique.

Cependant, la reconstruction de modèles 3D à partir d'images médicales reste un défi (cf. section 2.3.2). La méthode la plus utilisée consiste à reconstruire un modèle 3D basé sur les voxels [85], qui est généré à partir d'un processus qui détermine à quel élément physiologique appartient chaque voxel (la segmentation). Malheureusement, cette représentation contraint le calcul des propriétés géométriques à la taille des voxels [115] alors que de nombreuses analyses statistiques nécessitent une plus grande précision. Par exemple, le volume exact d'une tumeur doit souvent être évalué ou, dans le cas des neurosciences, l'épaisseur du cortex cérébral doit être mesuré pour identifier les schémas spatiaux de la maladie d'Alzheimer [148].

Ainsi, une représentation des différentes régions sous la forme d'un maillage 3D continu, représenté par des volumes fermés, est beaucoup plus adaptée qu'un modèle basé sur les voxels, car elle est plus précise et permet de meilleures analyses (cf. section 2.3.2).

Bien que de nombreux pipelines existent pour reconstruire des maillages à partir de données médicales, ceux-ci nécessitent un temps d'exécution très long et ne permettent pas de contrôler et corriger les incohérences. Une intervention d'un utilisateur expert est souvent nécessaire, notamment en cas de données d'entrée bruitées. De plus, dans le meilleur des cas, les surfaces reconstruites sont simplement juxtaposées et peuvent se chevaucher, aucun voisinage topologique entre les composants n'est reconstruits. Sans post-traitement, ces incohérences rendent le modèle produit inutilisable pour des processus de simulation physiologiques (par exemple pour la détection des tumeurs).

Dans ce chapitre, nous proposons tout d'abord un processus de reconstruction de modèle 3D s'appuyant sur les informations topologiques pour guider la reconstruction et pour produire des modèles contraints directement utilisables dans les applications de simulation. Dans la suite du document, nous nous intéressons plus particulièrement au domaine médical en proposant une méthode de reconstruction topologique d'un modèle 3D à partir d'un ensemble de primitives géométriques ainsi que des opérations de correction des incohérences géométriques (cf. section 5.1). Ensuite, nous nous appuyons sur ce processus pour proposer une nouvelle méthode de reconstruction volumique 3D à partir de données médicales discrètes, produisant un modèle anatomique utilisable directement pour les analyses statistiques des données physiologiques (cf. section 5.2). Enfin, nous comparons nos résultats avec les modèles produits par les logiciels cliniques couramment utilisés (cf. section 5.3).

## 5.1 Correction et reconstruction topologique

Comme nous l'avons présenté en section 3.4, pour représenter virtuellement un objet réel, le modèle 3D correspondant doit être cohérent géométriquement, c'est à dire à un complexe linéaire par morceaux (i.e. sans interpénétration ou sans éléments de taille nulle). De plus, pour qu'il soit utilisable pour des applications de simulation, il doit être structuré avec l'ensemble des relations de voisinage entre les éléments (sommets, arêtes, faces et volumes) et être caractérisé par les données sémantiques nécessaires aux algorithmes de simulation (couleur, propriétés de domaine, etc.). Dans cette section, nous présentons un processus de reconstruction topologique 3D à partir d'une soupe de polygones de manière robuste, précise et efficace. L'objectif est de produire un modèle géométrique et topologique permettant de représenter des objets "réels" tout en conservant les données sémantiques si elles existent. Le modèle généré doit donc satisfaire les propriétés suivantes :

- Géométrique : le modèle ne contient pas d'élément dégénéré (arêtes de longueur nulle, faces d'aire nulle, volumes nuls);
- Topologique : le modèle est orienté et doit être fermé en toute dimension;
- Géométrique et topologique : le modèle est une partition de l'espace (deux cellules de même dimension ne peuvent occuper les mêmes espaces) et l'espace entier est occupé.

L'ensemble de la méthode est basée sur l'utilisation du modèle de représentation des cartes généralisées pour valider formellement les contraintes topologiques à chaque étape du processus et utiliser des opérations topologiques robustes (cf. section 2.2.2). Les informations de voisinages permettent également d'améliorer l'efficacité des traitements. De plus, les informations sémantiques (données spécifiques au domaine) sont conservées lors de la réparation afin de maintenir un modèle directement utilisable dans de nombreux domaines d'application (impression 3D, simulation, visualisation).

Dans la suite, les étapes principales du processus sont décrites : la reconstruction topologique des données (cf. section 5.1.1) et le processus de corafinement 3D qui permet de traiter les intersections géométriques (cf. section 5.1.2). Certains résultats sont donnés et discutés dans la section 5.1.3.

### 5.1.1 Processus de reconstruction topologique

Le processus de reconstruction topologique vise à convertir le modèle géométrique d'entrée en modèle topologique orienté et fermé en éliminant les incohérences telles que les arêtes et faces dégénérées. On distingue deux étapes : la création des faces topologiques et la reconstruction des informations de voisinage des faces (les liens de dimension 2). La figure 5.2 montre toutes les étapes du processus de reconstruction topologique.

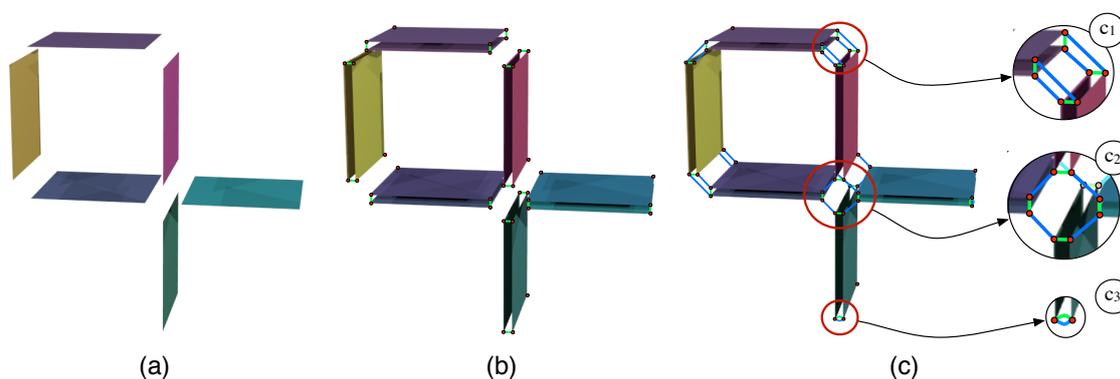


Figure 5.2 – Reconstruction topologique (vue éclatée) : (a) soupe de faces d'origine ; (b) faces topologiques : avec  $\alpha_0, \alpha_1$  et  $\alpha_3$  fermées,  $\alpha_2$  ouvert ; (c) fermeture de la relation  $\alpha_2$  : toutes les faces sont liées selon l'angle autour du bord géométrique ; ( $c_1$ ) deux faces sont directement reliées selon l'orientation ; ( $c_2$ ) plusieurs faces sont cousues selon l'orientation et la disposition des angles ; ( $c_3$ ) face pendante : arête incidente à une seule face unique où pour tous brins  $b$  dans l'arête  $b.\alpha_2 = b.\alpha_3$ .

#### Création des faces topologiques

Lors du chargement des données, la géométrie est contrôlée pour détecter et corriger, quand cela est possible, les éléments dégénérés. Pour chaque face géométrique valide, une face topologique est créée. Chaque face compte  $n * 4$  brins (avec  $n$  nombre d'arêtes), et les arêtes sont créées par les liens  $\alpha_0, \alpha_1$  et  $\alpha_3$  comme illustré dans la figure 5.2.b. Cette étape produit (par construction) des faces topologiques fermées à 0, 1 et 3 ; une marque d'orientation est fixée sur les brins de la face afin de satisfaire la propriété d'orientation. L'étape suivante du processus consiste à fermer les liens  $\alpha_2$  (reliant les faces topologiques) conformément à la propriété d'orientation grâce à un algorithme d'interclassement autour des arêtes.

#### Interclassement 3D

À ce stade, les liaisons  $\alpha_2$  sont ouvertes et ne suivent pas la propriété de fermeture. Le processus d'interclassement connecte autour de chaque arête géométrique, les faces topologiques incidentes (cf. figure 5.2). Pour cela, les faces sont triées en fonction de leur angle autour de l'arête (cf. figure 5.2.c), puis les brins sont successivement liés par  $\alpha_2$  en fonction de l'orientation (cf. figure 5.2.c<sub>1</sub> et c<sub>2</sub>). Cette gestion angulaire permet de traiter les faces pendantes (bord de l'objet) avec le même procédé comme le montre la figure 5.2.c<sub>3</sub>.

Ce processus de reconstruction topologique garantit le respect des propriétés de fermeture et d'orientation avec la production d'un modèle 3D orienté et fermé dépourvu d'éléments dégénérés.

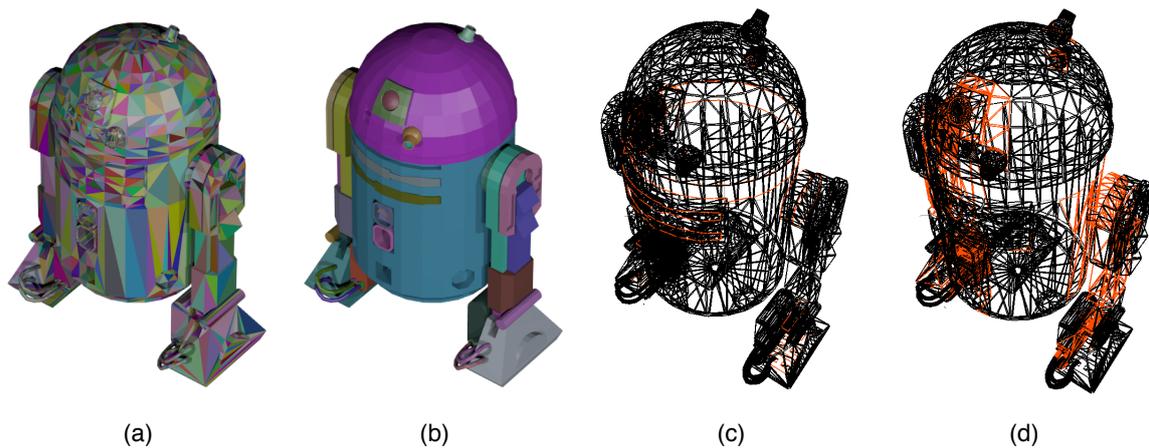


Figure 5.3 – Résultat de la reconstruction topologique : (a) ensemble initial de faces géométriques ; (b) après la reconstruction topologique, plusieurs volumes sont construits (un par couleur) ; (c) détection immédiate d’une face pendante (propriété topologique) ; (d) visualisation des intersections des faces (en rouge).

La figure 5.3 présente le résultat du processus de reconstruction topologique. À partir d’un ensemble de polygones, notre processus de reconstruction topologique produit un ensemble de volumes connectés (cf. figure 5.3.b). Grâce à l’information topologique, les faces pendantes sont directement détectées (cf. figure 5.3.c). Toutefois, des incohérences géométriques correspondant aux intersections peuvent être présentes dans le modèle de résultat et un processus de réparation est nécessaire.

### 5.1.2 Corafinement 3D

Le processus de reconstruction ne permet pas de contrôler et corriger les critères de partition de l’espace (deux éléments de même dimension ne doivent pas occuper le même espace). La deuxième étape consiste donc à résoudre les intersections avec une opération de corafinement 3D. Cet algorithme subdivise les cellules jusqu’à la satisfaction des contraintes de partitionnement. Les intersections de volumes (et les auto-intersections) sont résolues par un corafinement des cellules de dimension inférieure. Ce processus est composé de deux étapes : (1) le 1-co-raffinement induit l’insertion d’un sommet au point d’intersection de deux arêtes (2) le 2-co-raffinement correspond à la division des faces sur leurs intersections. Dans cette opération, le voisinage topologique est utilisé pour réduire le nombre de calculs effectués ce qui induit un temps de traitement plus faible et minimise les divergences dues aux imprécisions numériques. En effet, si une coupe a lieu sur une arête topologique (une cellule de dimension 1, orbite  $\langle \alpha_0, \alpha_2, \alpha_3, \rangle$ ), toutes les faces incidentes sont automatiquement découpées avec un seul traitement ce qui évite les redondances de calcul.

#### Découpe des arêtes (1-co-raffinement)

Le processus d’intersection d’arêtes (1-co-raffinement) repose sur le calcul des intersections entre les arêtes topologiques. Comme une arête topologique regroupe plusieurs arêtes géométriques (une par face incidente), le nombre d’éléments testés est réduit. De plus, l’insertion d’un sommet impacte automatiquement toutes les faces incidentes sans calcul supplémentaire. Cette opération est courante dans les modèles topologiques et est très efficace.

### Découpe des faces (2–corafinement)

Les calculs du processus de 2–corafinement s'appuient sur les plans 3D associés à chaque polygone. En 3D, l'intersection de deux plans non parallèles est une droite. Lorsque l'intersection existe, il est possible de scinder les deux faces le long de cette *ligne de scission* avec certaines contraintes. Concrètement, la décision de découpe devient un problème 1D dépendant de l'intersection entre la ligne de scission et chaque bord des faces. Le positionnement des intersections entre les arêtes des faces et la ligne de scission indique si la découpe est nécessaire ou non. Le figure 5.4 montre différentes configurations.

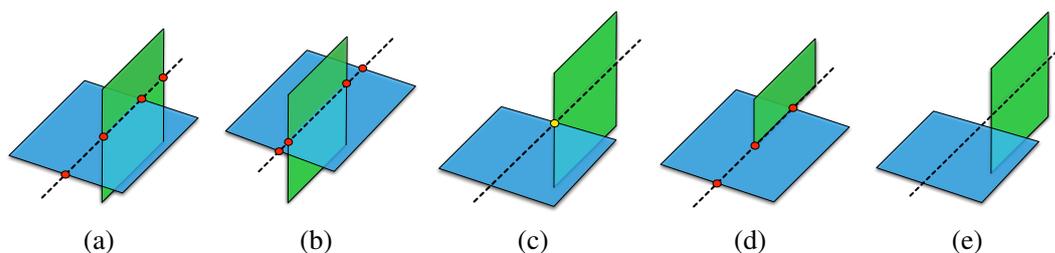


Figure 5.4 – Étude de cas d'intersection dans le 2–corafinement : (a) et (b) opération de découpe avec insertion de points sur les deux faces; (c) lorsque les faces se touchent en un seul point, la découpe est réalisée lors du 1–corafinement; (d) ligne de scission confondue avec une arête d'une face; (e) aucune intersection ne se produit.

Dans un modèle complexe contenant un grand nombre de polygones, une face peut être intersectée par plusieurs autres faces. Pour gérer ces multi-intersections, toutes les lignes de scission sont stockées dans une structure ad hoc sans découpe effective. Les lignes de scission contiennent tous les sommets d'intersection détectés et triés le long de son vecteur directeur à partir de tous les bords de chaque face intersectée. Une analyse de cette ligne délimite pour chaque face un segment avec un ensemble de sommets supplémentaires qui proviennent des calculs d'intersection. Les faces sont finalement découpées le long de ce segment résultant avec tous les sommets d'intersection inclus. La figure 5.5 illustre différents exemples où les intersections d'au moins deux faces génèrent des lignes de scission.

Une fois le processus de corafinement 3D appliqué, le modèle respecte les contraintes de partitionnement, il n'y a plus de superposition d'éléments de même dimension.

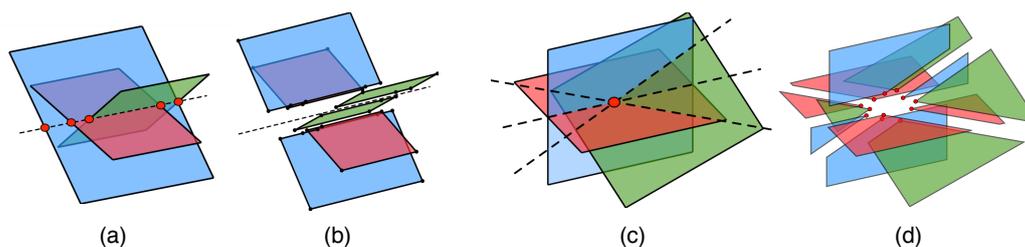


Figure 5.5 – Intersection entre plusieurs faces : (a) 3 faces sont intersectées sur la même ligne de scission; (b) les 3 faces sont coupées, et tous les sommets inclus sont ajoutés sur les nouvelles faces; (c) 3 faces sont intersectées sur des lignes de séparation différentes; (d) les 3 faces sont coupées, et le point d'intersection de la ligne de séparation est ajouté sur chaque nouvelle face.

### 5.1.3 Résultats et discussion

La figure 5.6 illustre l'intégralité du processus de reconstruction. Dans un premier temps, une reconstruction topologique est effectuée après importation des données brutes (cf. figure 5.6.a-b). Ensuite, nous exploitons la topologie pour l'étape de nettoyage/réparation qui garantit le respect des propriétés de notre modèle (cf. figure 5.6.c). Enfin, le processus applique une nouvelle passe de reconstruction topologique afin de créer la partition volumique de l'espace (cf. figure 5.6.d).

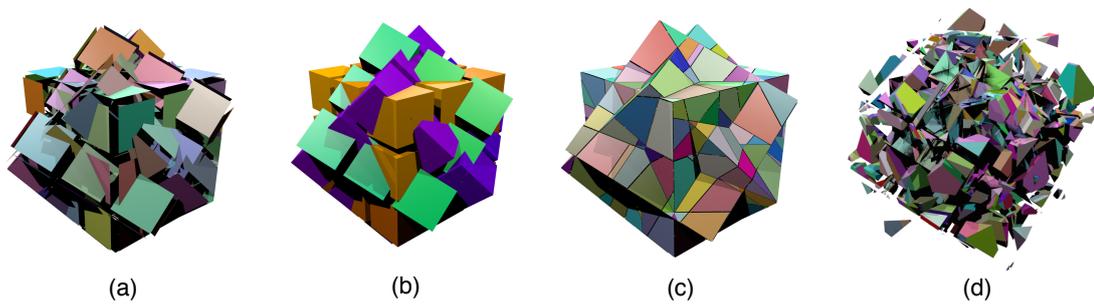


Figure 5.6 – Processus global : (a) 3 Rubik’s cubes superposés (total : 324 faces) ; (b) résultat après la première reconstruction topologique, 81 volumes sont construits ; (c) résultat après le processus de raffinement 3D, toutes les faces et arêtes sont divisées le long des intersections, le modèle est composé de 3998 faces et 618 volumes ; (d) vue éclatée de tous les volumes obtenus.

La figure 5.7 illustre un cas typique où la topologie permet d’améliorer les traitements. Deux faces s’intersectent (la bleue et la rouge) après calcul de la ligne de scission, elles sont découpées. L’insertion des sommets topologiques est automatiquement propagée aux faces adjacentes (la verte et la violette). Cette opération implicite évite un rappel aux processus de 1–corafrinement et d’interclassement pour conserver un modèle cohérent.

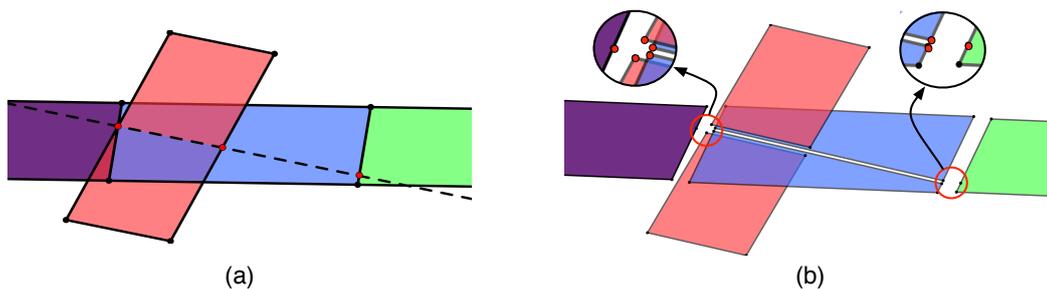


Figure 5.7 – Apport de la topologie pour réduire les traitements d’intersection : (a) 4 faces dont trois cousues par  $\alpha_2$  et la rouge ne croise que la bleue ; (b) Après découpage effectif : le nouveau sommet topologique est inséré sur l’arête topologique (et donc toutes les faces adjacentes).

Notre processus de correction et de reconstruction topologique a été mis en oeuvre avec deux outils, tous deux développés au sein de Xlim ASALI. Le premier est Jerboa [14], une bibliothèque Java dédiée aux cartes généralisées. Cet outil permet de prototyper rapidement des algorithmes pour évaluer leur faisabilité et leur efficacité, mais il n’offre pas de bonnes performances en termes de temps de calcul. Le second est Moka [159], le modéleur géométrique à base topologique basé sur la G- Carte et développé en C++ où de nombreuses opérations topologiques sont implémentées de manière efficace (cf. section 2.2.2).

L’application proposée permet de traiter des modèles complexes avec des temps de traitement raisonnables (quelques secondes à quelques minutes). Les informations de voisinage sont utilisées pour réduire le nombre de tests effectifs ce qui réduit la divergence des calculs et assure un algorithme plus robuste. Toutes ces opérations conduisent à satisfaire un modèle topologique spécifique avec des critères forts qui assurent une haute qualité de l’objet produit. Enfin, les objets obtenus correspondent à un PLC, formés de volumes disjoints, représentant une partition de l’espace orienté où toutes les informations topologiques sont cohérentes. Ces modèles peuvent être directement utilisés pour divers algorithmes de simulation. Nous verrons dans la section 5.2 comment ce processus est utilisé pour la reconstruction topologique de données médicales.

## 5.2 Reconstruction 3D de données médicales à multi label sémantiques

En 2019, la création du laboratoire commun I3M (Imagerie Métabolique Multi-noyaux Multi-organes) a officialisé la collaboration établie entre le CNRS, l'entreprise Siemens Healthineers, le Centre Hospitalier Universitaire et l'Université de Poitiers. Il associe trois équipes de la fédération MIRES (FR 3423) : l'équipe DACTIM-MIS du Laboratoire de Mathématiques et Applications (LMA, UMR CNRS 7348) et les équipes ICONES et IG de l'axe ASALI de XLIM (UMR CNRS 7262) de l'Université de Poitiers. L'objectif commun est de mettre en oeuvre des techniques innovantes d'Intelligence Artificielle pour le traitement et l'analyse automatique des images multi-modalités, pour l'aide au diagnostic et au suivi thérapeutique.

I3M regroupe environ 50 chercheurs et enseignant-chercheurs dont je fais partie. Mon rôle est d'apporter mon expertise en informatique graphique notamment pour les problématiques de reconstruction et de visualisation de modèles 3D à partir des données médicales. Dans ce contexte, je suis le porteur du projet région MEDIA-IRM 3D (MEDical Data Intelligent Analysis - Interpretation, Reconstruction and Manipulation in 3D) accepté en juillet 2020. Ce projet se veut résolument pluridisciplinaire en impliquant étroitement des équipes de mathématiques, traitement d'images et informatique, des équipes du secteur biologie/santé et des médecins avec un budget de 489Keuros (comprenant 4 thèses, 1 post doctorat et du matériel). Le caractère innovant du projet est basé sur la recherche académique de nouvelles méthodes de reconstruction et de visualisation de données médicales. Il exploite des informations générées aux CHU de Poitiers et de Limoges. L'objectif, à terme, est de fournir une aide personnalisée au diagnostic et au suivi thérapeutique à partir d'une analyse non invasive du cerveau, notamment des glioblastomes. La thèse de Sylvain Gerbaud traitant de la reconstruction de modèles 3D topologiques à partir d'images IRM est financée par ce projet et a débutée en octobre 2020. La thèse de Hayssam Obeid traitant de la visualisation multimodale, en cotutelle avec le Liban a débutée en décembre 2021.

La problématique centrale du projet Media IRM 3D réside dans la définition et l'utilisation d'un modèle de représentation globale et cohérente qui répond aux besoins de simulation statistique pour l'étude des données physiologiques cérébrales. La suite de ce chapitre présente le volet du projet traitant de la reconstruction automatique de modèles de cerveaux composés de plusieurs substances anatomiques. L'objectif est de fournir un modèle global, sans incohérences géométriques (un PLC), au plus proche de la réalité anatomique, avec un ensemble d'informations topologiques que nous utiliserons dans les autres activités du projet pour simuler la répartition et la circulation de certaines molécules à l'intérieur du cortex. Cette thématique correspond aux dix huit premiers mois de la thèse de Sylvain Gerbaud. Nous avons mis en place un nouveau pipeline pour la reconstruction de maillage volumique à partir d'acquisitions médicales (IRM). Pour guider le processus de reconstruction et pour répondre aux besoins de simulation statistique clinique, nous proposons un ensemble de contraintes géométriques, topologiques et sémantiques.

À partir d'un ensemble de voxels résultant des méthodes classiques d'acquisition et de segmentation, notre processus produit, sans aucune intervention de l'utilisateur, un maillage 3D volumétrique à partir de plusieurs ROI segmentés, exempt d'espaces vides et de chevauchements volumiques. Cette représentation respecte les contraintes de voisinage des différentes substances en se basant sur une identification sémantique cohérente (chaque valeur d'étiquette de la segmentation correspond à une structure anatomique) et des relations d'adjacence explicites définies par le modèle topologique des cartes généralisées. Les informations topologiques et sémantiques sont utilisées pour détecter et corriger les incohérences structurelles dans le maillage produit. Ainsi, les calculs géométriques sont évités et les corrections sont traitées localement, ce qui réduit considérablement les imprécisions numériques et les temps de calcul. Notre travail se fait en étroite collaboration avec plusieurs partenaires cliniques (radiologue, neuroradiologues et chirurgiens), qui orientent les choix en fonction des connaissances médicales pour l'automatisation de notre processus. Pour adapter le pipeline à une reconstruction d'organe, un opérateur peut donner des invariants anatomiques, exprimés comme un système de contraintes, notamment sémantiques, qui doivent être vérifiés par le modèle reconstruit et corrigé en cas d'incohérence.

Dans la suite de cette section, nous étudions tout d'abord quatre méthodes de reconstruction de modèles 3D utilisées en imagerie médicale (cf. section 5.2.1). Nous présentons ensuite les propriétés géométriques, topologiques et sémantiques attendues pour les modèles reconstruits avec notre méthode en justifiant nos choix (cf. section 5.2.4). Dans la section 5.2.5, nous détaillons les différentes étapes de notre processus. Enfin nous discutons des résultats obtenus pour différents jeux de données IRM et comparons avec les applications habituellement utilisées par les médecins (cf. section 5.3).

### 5.2.1 Méthodes de reconstruction étudiées

Afin de bien cerner la problématique qui concerne le sujet de recherche de la reconstruction de modèles 3D à partir de données IRM, nous avons étudié de manière plus approfondie plusieurs algorithmes de reconstruction existants. Dans la suite de cette section, nous décrivons en détail quatre méthodes de reconstruction qui ont été étudiées par Loubna Lechelek dans le cadre de son post doctorat (financé par le projet région IMPROVED). Parmi ces quatre méthodes, deux sont basées sur l'algorithme du Marching Cubes (Flying Edges [143] et Dual Marching Cubes [69]) et les deux autres sont des algorithmes basés sur les contours (Contour Interpolation [152], Multi level Partition Unit [26]). Ces différents travaux ont été redéveloppés par nos soins, puis nous avons comparé leur comportement et performance sur un panel de modèles synthétiques et un panel de modèles issus de données réelles fournies par le CHU de Poitiers.

#### Flying Edges (FE) – [143]

L'algorithme Flying Edges (FE) est un algorithme conçu à partir de l'algorithme MC et optimisé pour les processeurs multi-coeurs. Il opère sur une grille tridimensionnelle de voxels et utilise une table de consultation à partir de laquelle la configuration du triangle de chaque voxel est déterminée. La particularité du FE réside dans la manière dont les données volumiques sont traitées. Alors que le MC classique parcourt tous les voxels du volume en un seul passage, le FE utilise plusieurs passages de prétraitement afin d'assurer une continuité de surface et produire un ensemble volumique, ou a minima d'assurer la continuité de la surface tant que cela est possible.

#### Dual Marching Cubes (DMC) – [69]

Le Dual Marching Cube (DMC) est une stratégie qui génère un modèle 3D à surface fermée en reconstruisant des mailles quadrangulaires à partir de données discrètes volumiques. Les faces quadrilatères sont supposées produire une meilleure forme globale que les faces triangulaires. Le DMC s'appuie sur la table de consultation introduite dans [122] pour produire un maillage à partir de données volumiques telles qu'une grille discrète 3D. Dans notre étude, nous avons utilisé l'algorithme DMC [69] qui propose la reconstruction du maillage, puis une simplification topologique en utilisant le modèle des *half-edge* pour réduire le nombre de sommets irréguliers. Le résultat de cet algorithme est topologiquement cohérent à travers les frontières des cellules, il produit des volumes fermés / étanches.

#### Contour Interpolation (CI) – [152]

L'algorithme d'interpolation de contours (CI) est mis en oeuvre dans SlicerRT [130]. Cette méthode crée des modèles 3D à surface fermée à partir de contours planaires. L'idée principale de l'algorithme est d'interpoler entre les contours transversaux adjacents à l'aide d'un maillage, en passant par quatre étapes principales. La première étape supprime les trous de l'ensemble des contours pour éviter les erreurs dans le processus de triangulation. Les problèmes de trous apparaissent lorsque les contours intérieurs et extérieurs sont reliés et considérés comme un seul contour. La deuxième étape détermine quels contours

doivent être reliés par des triangles entre chacune des tranches. La troisième étape calcule les modèles de ramification pour les contours qui possèdent une correspondance sur une tranche adjacente. La dernière étape, appelée tuilage, joint les contours transversaux voisins en créant une bande de triangles utilisant les correspondances entre les sommets. Elle est basée sur la variante de l'algorithme PointWalk qui triangule le maillage en minimisant la longueur des arêtes traversant les contours. L'algorithme scelle ensuite les contours externes pour combler les vides situés en haut et en bas du modèle 3D en utilisant la triangulation de Delaunay qui construit la surface fermée finale.

### Multi level Partition Unit (MPU) – [26]

La méthode du Multi level Partition Unit (MPU) interprète les contours comme des points dans  $\mathbb{R}^3$  et utilise des modèles implicites de partition d'unité multi-niveau pour créer une surface qui correspond approximativement aux points 3D. L'idée principale de l'algorithme est de subdiviser l'ensemble des points dans l'espace à l'aide d'un schéma de subdivision récursif basé sur les *octrees*, où l'estimation de la surface de chaque partition est effectuée localement et les fonctions implicites qui se chevauchent sont recalculées pour produire la surface globale. La figure 5.8 rappelle brièvement le principe.

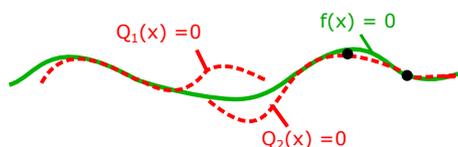


Figure 5.8 – Principe des modèles implicites MPU. Deux approximations locales (courbes rouges en pointillés) recalculées pour former la fonction MPU globale (courbe verte pleine) en utilisant des fonctions quadratiques pour approximer la forme locale.

### 5.2.2 Etude comparative

Nos tests et comparaisons ont été réalisés avec 3D Slicer [129] et Slicer RT (extension de slicer 3D dédiée à la reconstruction de surface 3D à partir d'un ensemble de contours) [130]. Pour cette étude, nous avons d'abord reconstruit les modèles synthétiques (cf. figure 5.9) à partir des jeux de données d'exemple inclus dans 3D Slicer. Nous avons également appliqué les quatre méthodes sur des IRM de cortex cérébraux humains fournis par le CHU de Poitiers dans le cadre du projet Media IRM 3D.

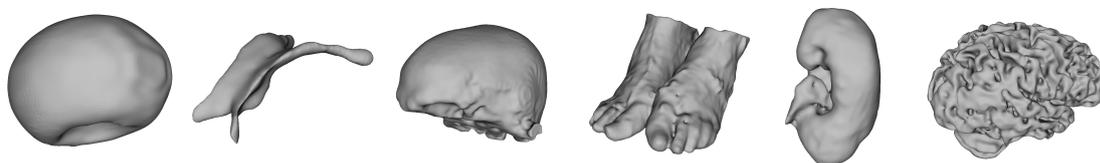


Figure 5.9 – Modèles 3D synthétiques utilisées pour valider nos algorithmes.

L'objectif majeur des méthodes de reconstruction dédiées à la reconstruction médicale est de produire un modèle 3D aussi proche que possible des données fournies par le processus de segmentation. C'est pourquoi, pour évaluer la fidélité anatomique de chaque modèle 3D, nous mesurons via une distance de Hausdorff la distance entre la segmentation et le maillage produit (c'est-à-dire la distance entre chaque point de la segmentation et le point le plus proche du modèle 3D). Nous avons également ajouté une étude quantitative basée sur l'analyse de la courbure des modèles reconstruits. Le comportement de la courbure transmet des informations sur la forme locale de la surface et donne un aperçu du degré de lissage.

La figure 5.10 présente un exemple de résultat de la mesure de distance sur un modèle de cerveau pour chacune des quatre méthodes de reconstruction. Une forte disparité est directement observée. Une couleur verte correspondant à la taille du voxel (1mm) est dominante pour les méthodes FE, DMC et MPU, mais les zones bleues (très proches) et rouges (très éloignées) ne sont pas positionnées aux mêmes endroits. Quant à l'IC, les résultats sont assez différents puisqu'on observe une prédominance de la couleur bleu (distance inférieure à la moitié de la taille du voxel) et rouge (supérieure à la taille du voxel). Les résultats peuvent être approfondis en regardant les courbes qui représentent la répartition cumulée en fonction de la distance. En ce qui concerne FE, DMC et MPU, la majorité des points de la surface se trouve à moins d'un millimètre de la segmentation (entre 80% et 90%). Quant au CI, il produit une répartition équilibrée avec la moitié des points très proches et l'autre moitié très éloignée. Pour mieux appréhender ces disparités, la figure 5.11 montre une coupe transversale permettant de visualiser et différencier les surfaces reconstruites pour chaque méthode de reconstruction par rapport à une même segmentation.

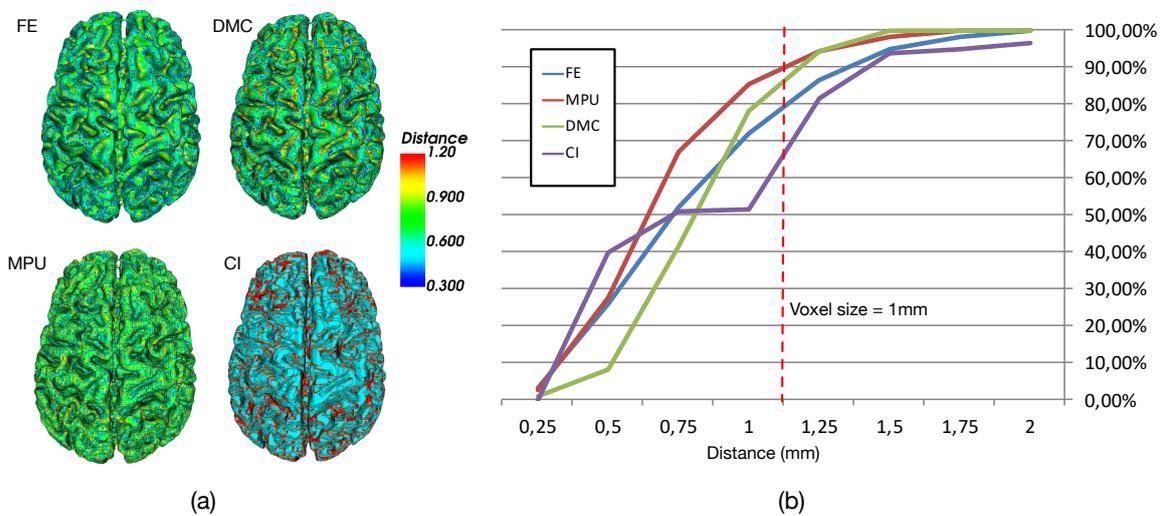


Figure 5.10 – Visualisation de la distance entre la segmentation et le modèle 3D reconstruit. La taille du voxel de la grille de segmentation sur l'axe transversal est de 1mm : (a) distance calculée avec la métrique de hausdorff (en bleu les distances faibles et en rouge les distances élevées). (b) pourcentage cumulé de la proportion d'éléments en fonction de leur distance à la segmentation.

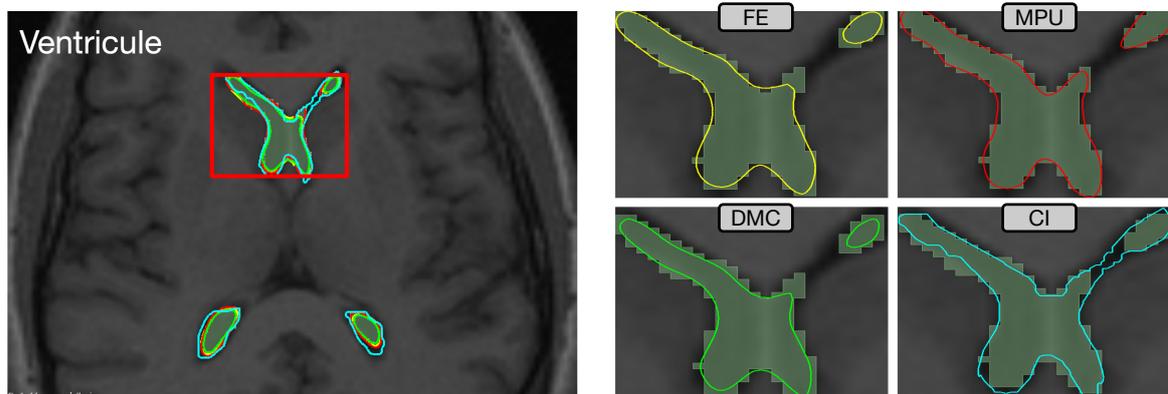


Figure 5.11 – Empreinte 2D de chaque modèle 3D produits pour la reconstruction d'une segmentation correspondant au ventricule cérébral avec FE (jaune), MPU (rouge), DMC (vert) et CI (cyan).

Afin d'élargir l'évaluation des méthodes de reconstruction, nous avons testé chacune d'entre elles sur 40 cortex cérébraux aux géométries complexes (cf. figure 5.12). En moyenne, les points du maillage reconstruit sont majoritairement à une distance inférieure à la taille du voxel (précision de la segmentation) avec : 90,33 % pour le DMC, 80,51% pour le MPU, 78,39 % pour le FE et 45,95% pour le CI. Les statistiques montrent que les écarts-types de DMC, FE et MPU restent relativement faibles par rapport à CI (en moyenne : DMC 0,26, FE 0,35, MPU 0,35, CI 1,91).

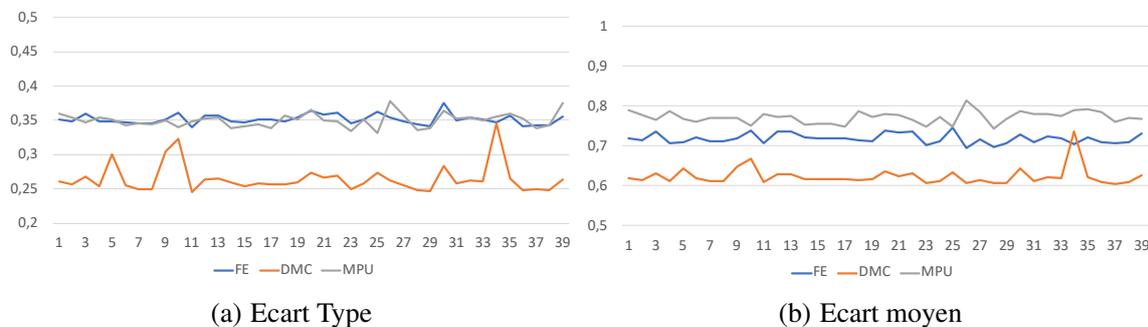


Figure 5.12 – Reconstruction FE, DMC et MPU calculée sur chacun des 40 cortex cérébraux de notre panel clinique (en abscisse); CI n'est pas représenté en raison de son écart-type élevé.

Nous avons également comparé les méthodes de reconstruction en analysant les informations de courbure des surfaces reconstruites. La figure 5.13 montre l'estimation de la courbure en chaque point de maillage avec une grande hétérogénéité des résultats. La couleur verte est prédominante pour les quatre méthodes. La couleur rouge (courbure très élevée) est peu présente pour la FE par rapport à DMC, MPU et CI. Cela signifie que, dans l'ensemble, la surface de courbure de la FE est relativement faible par rapport à DMC, MPU ou CI.

Le côté droit de la figure 5.13 montrent la courbure moyenne en fonction du pourcentage de points. Si l'on considère les courbures faibles (entre -0,25 et +0,25), le FE a le plus grand pourcentage de points. Cela est principalement dû au fait que FE ne produit pas de surfaces lisses, de sorte que les triangles de surface ont tendance à suivre les limites du voxel. Le post-traitement de lissage de maillage appliqué au modèle de sortie donne un effet lisse, mais en contrepartie, il tend à modifier la forme locale de la surface et à lisser considérablement les zones à fortes courbures. Contrairement au FE, le DMC produit des surfaces légèrement lisses grâce à son algorithme qui joue sur la valence des faces; il ne nécessite donc pas un fort lissage. Quant au MPU, puisqu'il est basé sur des fonctions implicites adaptatives qui confinent la reconstruction à un paramètre de tolérance spécifié par l'utilisateur, la forme locale de la surface dépend du choix de la valeur de tolérance. Le processus CI produit plus de courbures élevées (supérieures à +0,5). Cela est dû au fait que le modèle n'a pas subi de lissage de post-traitement du maillage, de sorte que les zones à fortes courbures sont maintenues.

### 5.2.3 Bilan

Les méthodes FE et DMC sont globalement très proches en termes de résultats, elles suivent bien la segmentation, mais produisent des modèles non lissés (forme d'escaliers). Le traitement de lissage permet de présenter l'aspect naturel des structures anatomiques, mais en contrepartie, il peut dégrader la précision de la reconstruction. L'examen de l'écart-type moyen montre que le DMC est plus proche de la segmentation car il est calculé à partir du dual d'un MC (cf. figure 5.12.a), les points de la surface sont donc plus proches du centre des voxels de la segmentation. Malheureusement, le modèle produit étant à l'intérieur de la zone de segmentation, lors de la reconstruction de plusieurs éléments anatomiques, les volumes générés seront disjoints et un post traitement sera nécessaire pour gérer les vides générés par une reconstruction multisegment.

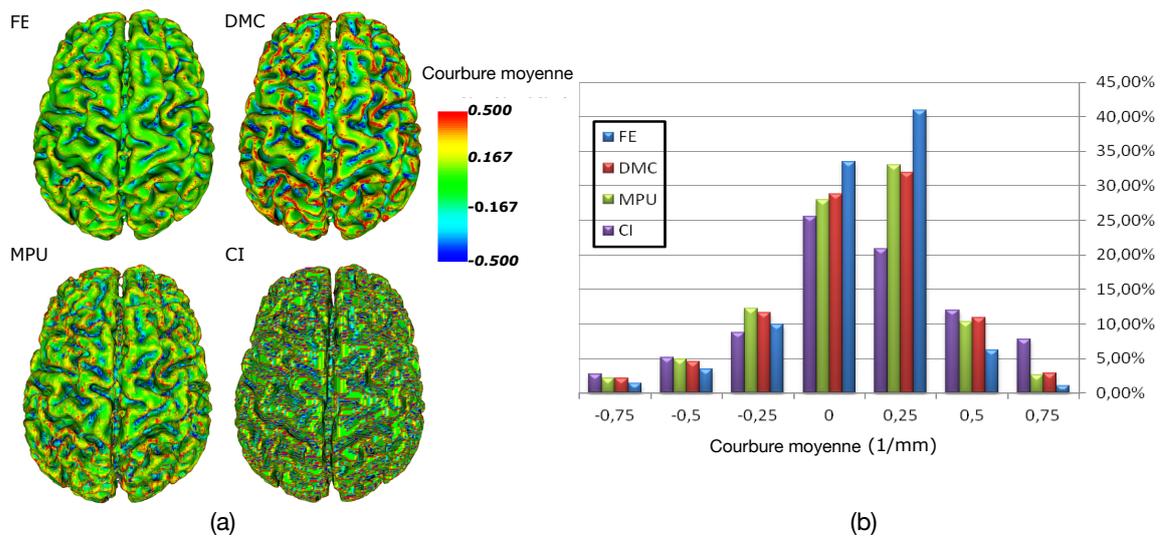


Figure 5.13 – Visualisation de la courbure moyenne sur un modèle de cerveau en 3D : (a) courbure moyenne estimée par VTK (les triangles sont codés par couleur en fonction de la courbure ; le bleu reflète les courbures faibles et le rouge les courbures élevées). (b) pourcentage de points en fonction de leur valeur de courbure moyenne.

Le MPU est globalement plus précis, car il est basée sur des fonctions implicites adaptatives qui se conforment aux caractéristiques et détails locaux. Mais si cette méthode s'avère très précise pour des régions à faible rayon de courbure, elle l'est beaucoup moins sur des éléments saillants (c.f. l'erreur maximale calculée). L'un des principaux inconvénients du MPU est l'intégration dans les fonctions implicites à la fois de la flexibilité de la spécification de la régularité d'une région et de la précision de l'approximation de la forme par rapport aux données fournies par la segmentation. Pour obtenir un niveau élevé de lissage, il est nécessaire d'utiliser une valeur relativement grande de la tolérance, ce qui se traduira ensuite par une mauvaise approximation de la forme originale de la segmentation ; l'utilisateur doit donc ajuster le paramètre de tolérance pour produire des résultats visuellement plausibles avec des statistiques d'erreur raisonnables.

Basés sur ces différentes méthodes, de nombreux pipelines sont utilisés par les médecins pour reconstruire des maillages (surfiques) à partir de données médicales. Le principal problème est qu'ils nécessitent un temps d'exécution très long et utilisent des pipelines étendus qui ne permettent pas de contrôler et corriger les incohérences du maillage final. Une intervention d'un utilisateur expert est souvent nécessaire pour s'assurer que les limites sont bien définies pour chaque substance à l'intérieur d'un cerveau reconstruit. Actuellement, le logiciel *FreeSurfer* largement utilisé est le principal logiciel permettant aux neuroscientifiques de travailler avec un maillage continu en 3D [39]. Il fournit de nombreux outils de calcul d'informations 2D et 3D pour l'analyse des images IRM [115, 148], tels qu'un calcul de l'épaisseur corticale [45], un algorithme de dépouillement de crâne [145], ou encore une segmentation automatisée des structures neuroanatomiques du cerveau entier [46]. Néanmoins, il souffre des inconvénients mentionnés ci-dessus, notamment en cas de données IRM d'entrée bruitées qui demandent un temps de traitement de plusieurs dizaines d'heures pour un résultat incertain. De plus, il ne calcule aucun voisinage topologique entre les composants reconstruits. Dans le meilleur des cas, leurs surfaces sont simplement juxtaposées et peuvent présenter des "zones de contact", mais dans le pire des cas, leurs volumes peuvent se chevaucher. Sans post traitement, ces incohérences rendent le modèle produit inutilisable pour des processus de simulation physiologique (par exemple pour la détection des tumeurs).

### 5.2.4 Modèle topologique pour le médical

Contrairement à une représentation discrète, une approche continue est plus appropriée pour contrôler la topologie car elle représente plus précisément la forme des objets reconstruits. Malheureusement, même si chaque segment est reconstruit correctement d'un point de vue topologique, la fusion de tous les segments ne fournit pas nécessairement une topologie globale correcte. En pratique, des espaces vides ou des chevauchements peuvent apparaître, en raison des processus de reconstruction habituellement utilisés qui ne permettent pas une gestion globale de l'objet (cf. section 2.3.2). Comme notre objectif est de reconstruire un objet multi-segment cohérent, nous proposons d'utiliser un modèle de contraintes basé sur le modèle topologique des cartes généralisées.

Un organe, quel qu'il soit, reste un objet réel. Comme nous l'avons vu dans la section précédente, la représentation topologique d'un objet réel doit correspondre à une partition de l'espace 3D qui respecte des contraintes géométriques et topologiques (cf. section 5.1) :

- Géométrie : le modèle ne contient pas d'élément dégénéré (arêtes de longueurs nulles, faces d'aire nulle).
- Topologique : le modèle est orienté et doit être fermé en toute dimension.
- Géométrie et topologique : le modèle est une partition de l'espace (deux cellules de même dimension ne peuvent occuper les mêmes espaces) et l'espace entier est occupé.

L'ensemble de la méthode est basée sur l'utilisation du modèle des cartes généralisées ce qui permet de valider facilement les contraintes topologiques. Le résultat final attendu de notre processus de reconstruction correspond à une partition continue de l'espace 3D en volumes adjacents représentant les différentes régions d'intérêt de la segmentation (ROI). En effet, deux segments ne doivent pas se chevaucher et l'espace entier doit être rempli. Topologiquement parlant, les segments sont orientables et avec des frontières. Dans la partition, l'adjacence entre les volumes de différentes substances est bien identifiée par leurs faces partagées.

En s'appuyant sur ce modèle, le processus de génération garantit que :

- le modèle correspond à une partition de l'espace 3D (donc exempt de vide, de chevauchement et tout l'espace est représenté)
- la segmentation reconstruite est exempte de bruit ;
- les contraintes anatomiques sont prises en compte (voisinage des substances) ;
- les surfaces du maillage sont aussi proches que possible des données anatomiques grâce à une approximation géométrique fine.

Nous avons choisi de combiner la structure des cartes généralisées (représentation explicite des adjacences) et la méthode récente (et parallélisée) basée sur les Marching Cubes (MC) du Flying Edges (FE) pour la construction de surfaces fermées (cf. section 5.2.1). L'utilisation du processus simple et rapide de FE permet de produire des frontières cohérentes, il est donc adapté à la fusion de plusieurs objets sans approximation géométrique. C'est une bonne base pour notre processus, en rendant plus facile et précise la mise en place de la structure topologique.

### 5.2.5 Mise en oeuvre détaillée

L'ensemble de notre processus est présenté dans la figure 5.14. Il est composé de trois étapes principales : (1) reconstruction topologique et nettoyage du bruit pour chaque ROI (2) réunification des segments, détection et correction des incohérences sémantiques et de voisinage (3) modification géométrique pour que le modèle soit au plus proche des données originales anatomiques.

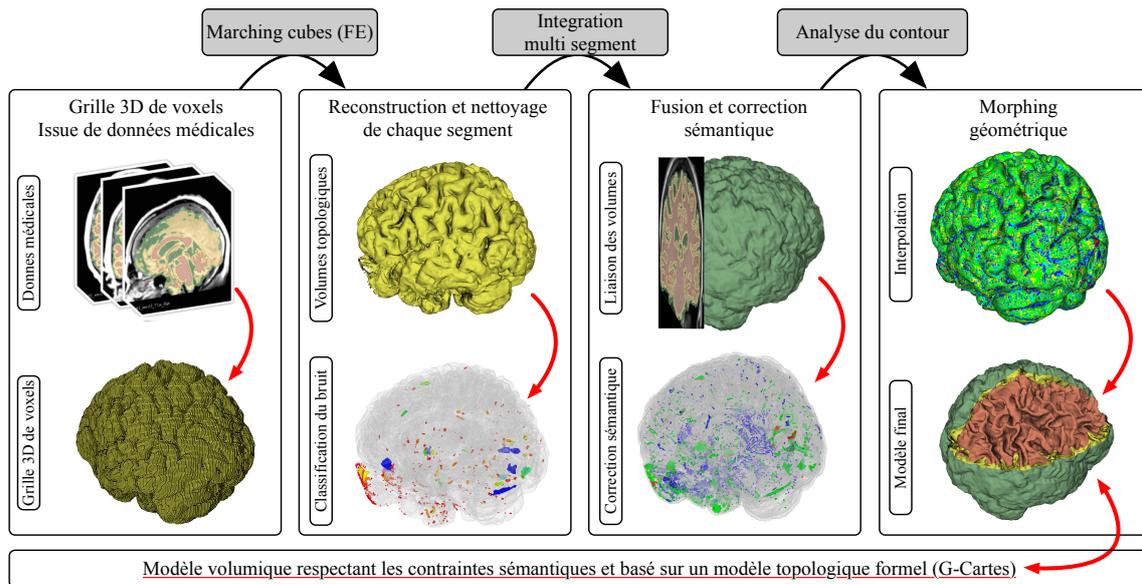


Figure 5.14 – Notre pipeline est décrit comme suit : (1) Reconstruction topologique basée sur *Marching Cubes* et nettoyage du bruit pour chaque ROI. (2) Fusion de tous les segments reconstruits, détection et correction des incohérences sémantiques et de voisinage. (3) Transformation géométrique pour déformer le modèle le plus proche possible des données originales et préserver les contraintes topologiques et sémantiques.

La première étape du pipeline consiste en une reconstruction basée sur une méthode de type *Marching Cubes* qui produit l'ensemble des surfaces correspondant aux contours de chaque ROI. Nous utilisons ici la méthode du *Flying Edges* décrite dans la section 5.2.1. Les surfaces sont ensuite associées à une carte généralisée pour construire topologiquement l'ensemble des volumes fermés. Les informations sémantiques sont stockées pour préserver la nature des substances et un processus de nettoyage est appliqué pour éliminer les éléments indésirables correspondant au bruit d'acquisition.

Lors de la deuxième étape, les segments anatomiques sont intégrés dans un même espace (regroupés dans une même carte généralisée). Puis, les volumes sont reliés topologiquement par leurs faces communes lorsqu'ils sont adjacents. Cette étape fournit toutes les informations d'adjacence entre les volumes, ce qui conduit à un voisinage détaillé et contrôlé du maillage. Les propriétés topologiques sont utilisées pour détecter et corriger les incohérences de voisinage, c'est-à-dire (i) les espaces vides (ii) les voisinages anatomiques incorrects. Ces opérations automatiques sont locales et résolues uniquement au moyen d'informations topologiques et sémantiques, aucun calcul géométrique n'est nécessaire.

Dans une dernière étape, une transformation qui préserve la topologie (homéomorphisme) et les contraintes sémantiques est appliquée pour calculer une forme aussi proche que possible des données originales. À cette fin, des informations sur les voxels, telles que l'intensité ou la PVE [11], sont utilisées. Lors du pré-traitement de la segmentation, pour chaque voxel, la PVE est calculée à partir d'une interpolation entre son intensité et celles qui l'entourent. Cette valeur définit le segment dans lequel le voxel sera assigné, représenté par une valeur probabiliste. En déformant le maillage en fonction de ce ratio de couverture, nous assurons une meilleure adaptation de la surface.

Ce pipeline est développé dans le logiciel open-source 3D Slicer [43], qui offre de nombreux outils pour analyser et manipuler les données médicales. Le noyau topologique *Moka Library* [159] a été intégré à 3D Slicer et est utilisé pour tous les processus topologiques.

### Etape 1 : Reconstruction de chaque substance

#### Reconstruction topologique des volumes

À partir d'une grille de voxels segmentée, la méthode de reconstruction du Flying Edges (FE) est appliquée sur chaque segment pour reconstruire un maillage triangulaire à surface fermée pour chaque substance (cf. figure 5.15). Dans cette étape, nous supposons que l'isosurface à reconstruire coupe toujours les bords de la grille de segmentation exactement en son point médian. Ainsi les sommets de l'isosurface sont placés au milieu du bord correspondant de la grille de voxels.

Le choix de ne pas traiter les aspects géométriques à ce stade du pipeline nous permet d'assurer la complémentarité des volumes les uns par rapport aux autres. Cela nous permet également d'appliquer toutes nos opérations topologiques (par exemple, la fusion des volumes) sans avoir à considérer les cas géométriquement dégénérés tels que les triangles plats. La géométrie sera raffinée à la fin de notre pipeline pour garantir une correspondance correcte avec la forme réelle de la substance. Le modèle résultant est garanti sans auto-intersection ni chevauchement grâce au partitionnement de l'espace produit par le FE qui garantit la continuité des surfaces générées par l'algorithme de MC.

Une fois les mailles géométriques produites, nous souhaitons reconstruire la structure topologique à partir de l'ensemble des faces produites par le MC. Pour cela, la carte généralisée associée est construite en utilisant le workflow décrit dans section 5.1. Ce processus nous permet de construire l'ensemble des liens topologiques. Lors du calcul, bien que le FE ne produise pas d'incohérence géométrique, nous vérifions l'ensemble des contraintes géométriques d'un PLC, c'est à dire : absence d'intersections, de faces non planes et d'arêtes de longueur nulle. Ce workflow appliqué au maillage surfacique fourni par le FE génère des volumes fermés et étanches. Dans ces volumes, les faces adjacentes sont explicitement liées à leur bord commun par des liens topologiques de dimension 2 (involution  $\alpha_2$  - figure 5.16).

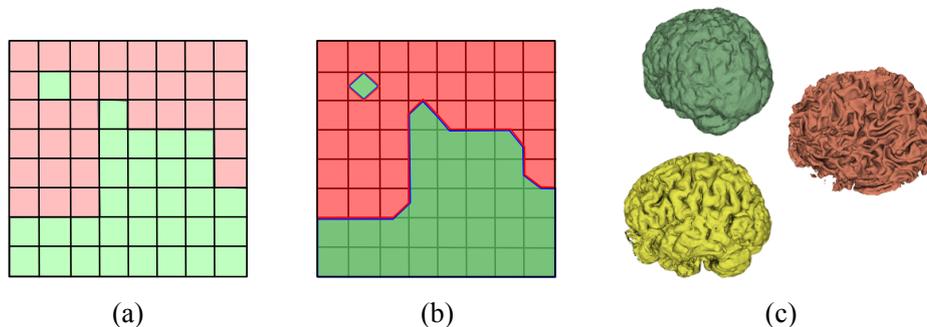


Figure 5.15 – De manière classique, la méthode FE est appliquée aux différentes régions (a), et produit (en 3D) un ensemble de faces (b). Ce processus est appliqué indépendamment à chaque région, produisant un modèle 3D de chaque substance (c).

#### Classification et nettoyage

La deuxième partie de cette étape consiste à éliminer les volumes correspondant au bruit produit pendant la phase d'acquisition pour chaque segment. À ce stade, chaque volume est identifié et ses caractéristiques géométriques et topologiques (taille, nombre de faces) sont calculées localement grâce à la structure des cartes généralisées qui permet de réaliser ces traitements sur les cellules de dimension 3.

La classification se fait en prenant en compte les informations de segmentation pour associer chaque volume à une substance. Dans notre cas, le terme classification est utilisé pour séparer les volumes qui sont considérés comme du bruit, et ceux qui doivent être conservés. Parmi ces derniers sont étiquetés comme *composants principaux* sur la base de l'analyse de caractéristiques spécifiques (ce sont les plus grands objets en terme de volume et de nombre de faces). Les objets plus petits qui sont éloignés des com-

posantes principales sont étiquetés soit des volumes de bruit, soit comme des volumes à conserver. Pour cela, la distance à la composante principale est calculée et la taille des volumes est mesurée au moyen d'une décomposition tétraédrique (en utilisant la bibliothèque Tetgen [77]). La figure 5.17 présente le résultat de la classification multicritères où les volumes considérés comme inutiles peuvent être supprimés. Ces traitements habituellement fastidieux et nécessitant un réglage fin de nombreux paramètres sont traités automatiquement tout en conservant l'ensemble des propriétés topologiques et sémantiques.

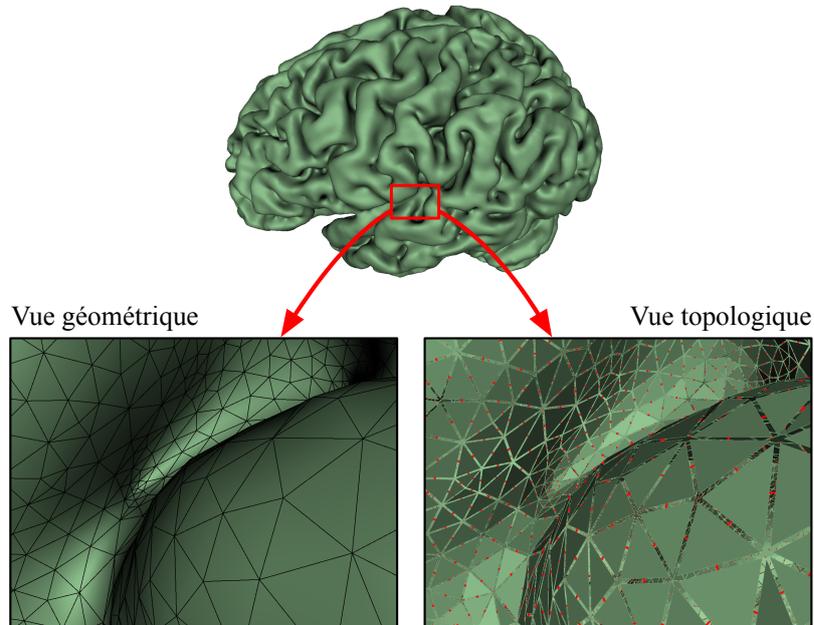


Figure 5.16 – Le maillage géométrique est associé à la 3G-Map correspondante. La vue topologique montre les liens de dimension 2 entre les faces.

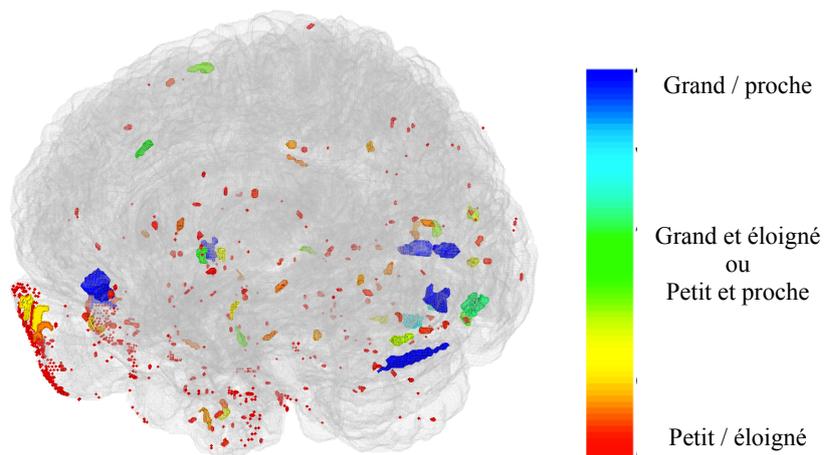


Figure 5.17 – Multi-critères pour la détection du bruit : les volumes de petite taille et éloignés de la composante principale sont classés comme bruit et peuvent être supprimés.

## Etape 2 : Intégration multi-segments

Pour réunifier l'ensemble des données construit dans l'étape précédente, tous les volumes correspondant à chaque segment sont intégrés dans un même espace. Une première opération topologique est appliquée : elle permet de joindre les volumes adjacents par leurs faces identiques, via leur involution  $\alpha_3$ . Ensuite, les défauts tels que les espaces vides ou les incohérences de voisinage sont détectés et corrigés.

### Liaison volumique entre les volumes adjacents

Tous les segments sont reconstruits par le FE à partir de la même segmentation et avec une même isovaleur de 0,5 (cf. section 5.2.5). Cette propriété du FE garantit que les faces communes entre les segments ont un ajustement exact (cf. figure 5.18.a). Le processus d'identification consiste donc à relier les faces identiques de différents segments adjacents (par un lien de dimension 3, liaison  $\alpha_3$ ). La figure 5.18.b montre le processus de couture topologique correspondant à la segmentation présente dans la figure 5.15. Par ailleurs, les faces sans correspondance sont laissées libres (c'est-à-dire sans volume adjacent). Elles sont traitées dans l'étape suivante.

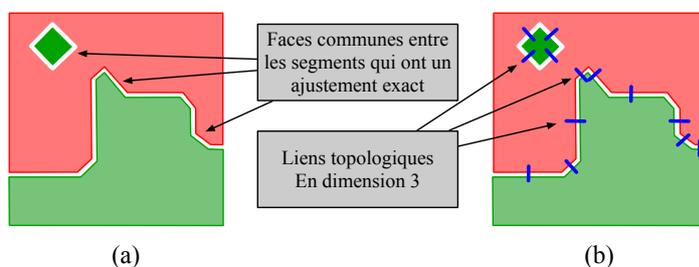


Figure 5.18 – Le lien topologique en dimension 3 correspond à (a) identifier et (b) relier les faces identiques entre les différents segments. Dans ce qui suit, les liens topologiques ne sont pas représentés pour une meilleure lisibilité.

### Contrôle et correction sémantique

La connaissance des relations de voisinage entre les volumes et les informations sémantiques relatives aux substances permettent de contrôler les modèles par rapport aux contraintes anatomiques. Grâce aux voisinages topologiques des volumes, les espaces de vide sont directement localisés (ils sont définis par des liens libres  $\alpha_3$ ) et une opération de fermeture topologique permet de construire directement le volume qui remplit cet espace vide. Ces volumes sont alors étiquetés *unnamed* et peuvent être traités localement pour reconstruire correctement l'espace en assurant une cohérence de voisinage sémantique valide (le voisinage entre les substances). On distingue trois cas (cf. figure 5.19) :

- **Cas 1** : le voisinage d'un volume *unnamed* ne correspond qu'à un seul segment. Cet artefact est de taille très faible et correspond à une erreur de segmentation. Ce type de volume est automatiquement supprimé (figure 5.20 - cas 1), ou conservé en fonction des contraintes anatomiques ;
- **Cas 2** : le voisinage d'un volume *unnamed* correspond à plusieurs segments, il pourrait reconnecter deux volumes qui doivent, par connaissance anatomique, être liés (figure 5.21 - cas 2) ;
- **Cas 3** : le voisinage d'un volume *unnamed* correspond à plusieurs segments sans connaissance anatomique explicite à exploiter. La correction est effectuée en utilisant une information probabiliste d'appartenance issue de la segmentation (figure 5.21 - cas 3).

En résumé, dans cette étape de résolution des contraintes sémantiques, si les contraintes sémantiques et anatomiques ne nous permettent pas de résoudre une incohérence, une étude locale est effectuée pour combler le vide. Une fois l'appartenance sémantique définie, une opération topologique (locale) de fusion est appliquée sur les volumes traités pour supprimer les faces inutiles (qui séparent une même substance).

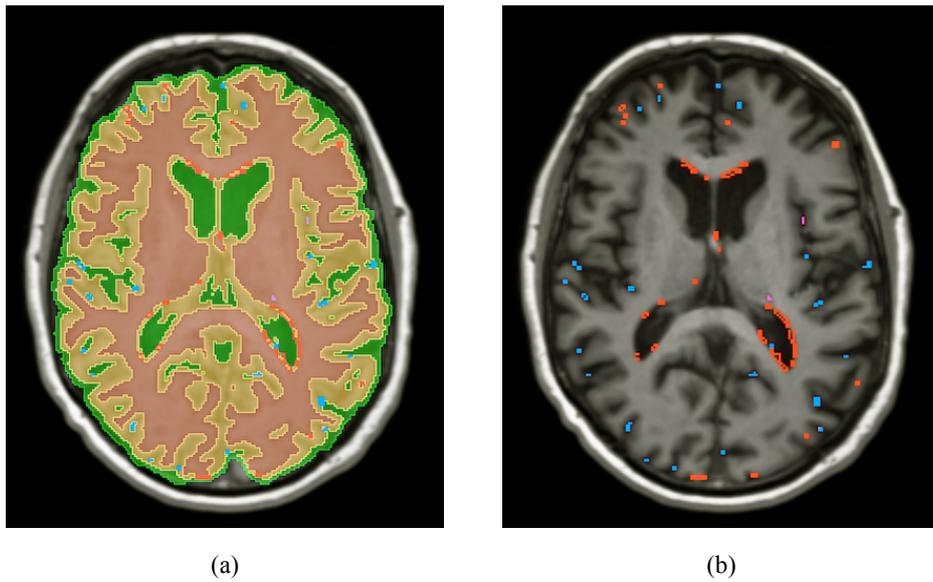


Figure 5.19 – Volumes vides identifiés lors de la deuxième étape (a) image IRM, ROI segmenté et voxels marqués en bleu (cas 1), rouge (cas 2) et violet (cas 3) (b) image IRM et voxels marqués en bleu (cas 1), rouge (cas 2) et violet (cas 3).

À la fin de cette étape, nous obtenons un modèle 3D topologique avec des relations d'adjacence explicites entre les segments reconstruits. Toutes les opérations sont calculées localement au cours de ces étapes, en utilisant uniquement la topologie, ainsi le modèle est préservé de toute approximation géométrique. Il respecte les données sémantiques et est exempt de défauts topologiques et géométriques. En outre, tout au long du processus, les informations sémantiques sont mises à jour et restent cohérentes.

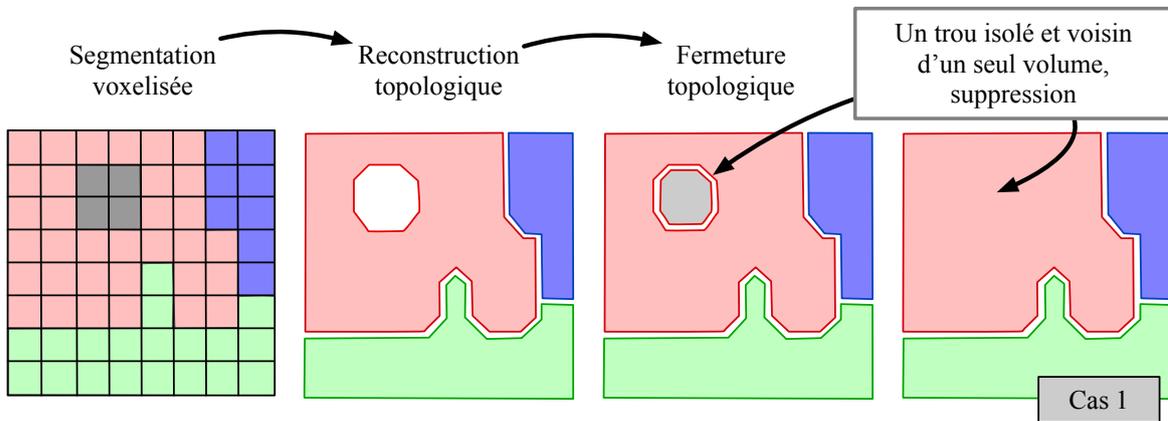


Figure 5.20 – Trou isolé : les voisins d'un seul segment peuvent être directement supprimés.

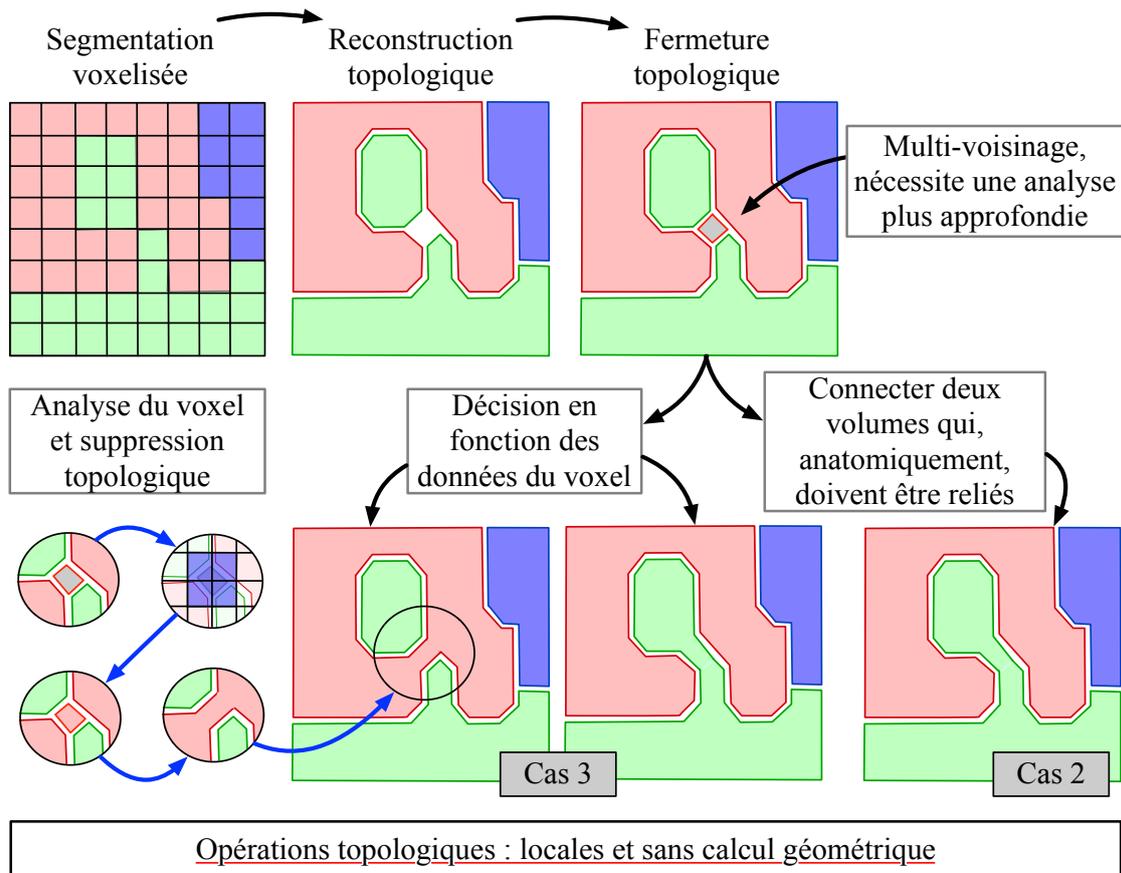


Figure 5.21 – Trou multi-connecté : après fermeture topologique, si le volume généré est amené à connecter deux volumes qui, anatomiquement, doivent être liés, la sémantique correspondante est directement affectée (cas 2). Sinon, une analyse des voxels permet de définir la bonne sémantique (analyse en bas à gauche et cas 3). Enfin, les faces inutiles sont détectées (face entre la même substance) et supprimées par une opération topologique.

### Etape 3 : Interpolation géométrique

Dans l'étape de reconstruction de surface basée sur le FE, les sommets des mailles générées sont arbitrairement calculés au point médian des bords de la grille de segmentation (étape 1). Cette procédure nous permet d'obtenir une correspondance exacte entre les segments adjacents lors de l'étape de fusion (étape 2). Cependant, ce positionnement géométrique est arbitraire et n'approche que grossièrement la forme anatomique réelle des segments (l'erreur est cependant limitée à la moitié de la taille des voxels).

Pour calculer une position plus précise des sommets, il est possible d'utiliser des informations statistiques extraites des voxels de la grille d'origine (par exemple, l'intensité du voxel en niveau de gris). Mais, en imagerie cérébrale, nous disposons des informations probabilistes d'appartenance à chaque ROI calculées lors du prétraitement de segmentation. Nous utilisons ces données, appelé PVE, pour définir une nouvelle isovaleur et modifier la position des sommets comme le montre la figure 5.22,

Les relations topologiques entre les segments établies lors du processus de reconstruction assurent un déplacement des sommets topologiques. Ainsi, aucun espace vide et aucun chevauchement ne peut apparaître. En effet, si un sommet appartient à une ou plusieurs faces partagées de volumes adjacents, la forme des deux volumes est modifiée lorsque ce sommet (partagé) est déplacé. La figure 5.23 montre une visualisation du déplacement du morphing géométrique, la variation moyenne est de l'ordre d'un quart de longueur de voxel, selon la direction.

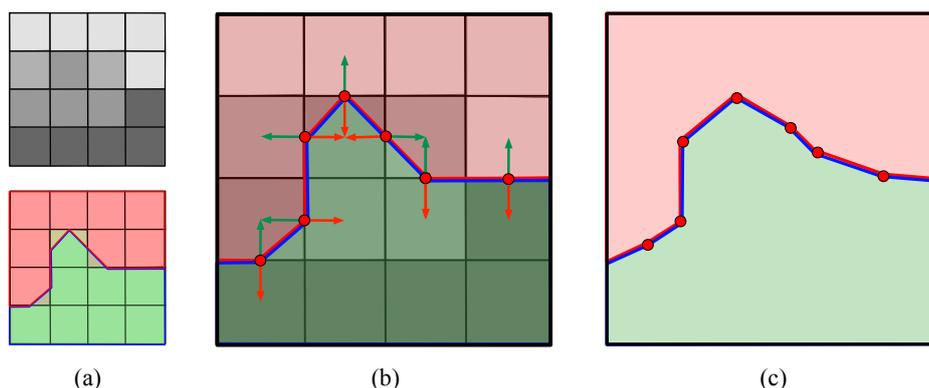


Figure 5.22 – Morphing géométrique : (a) voxels originaux et segmentation correspondante et reconstruction topologique 3D. (b) les flèches illustre le déplacement potentiel de chaque vertex par estimation de sa position basée sur une interpolation des informations statistiques du voxel. (c) nouvelle position géométrique de ce sommet, au plus proche de l’anatomie réelle.

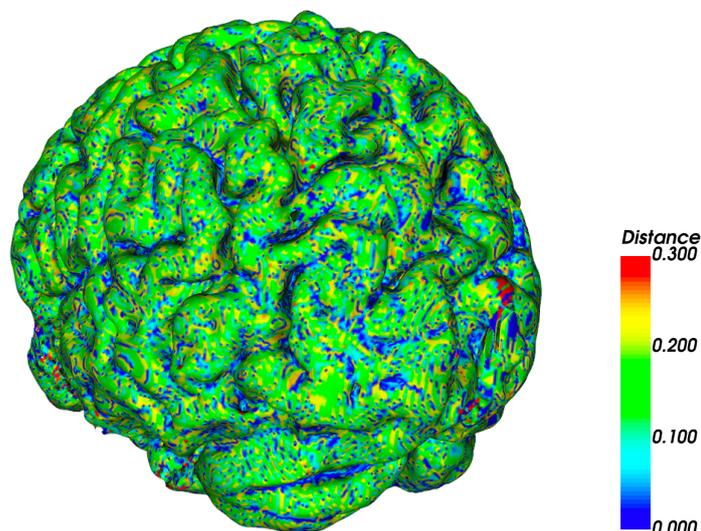


Figure 5.23 – Modification du positionnement géométrique des sommets : afin que le modèle soit aussi proche que possible de l’anatomie réelle. La distance de déplacement ne peut pas dépasser un demi-voxel, c’est-à-dire 0,5 mm. La plupart des variations se situent autour de 0,2/0,3mm.

### 5.3 Résultats et comparaisons

Nous présentons dans cette section les statistiques et le temps de traitement de notre pipeline pour différents modèles reconstruits. Puis, nous comparons les modèles 3D produits par notre processus avec les méthodes de reconstruction présentées en section 5.2.1 (FE [143], DMC [122] et MPU [26]). Enfin, nous comparons notre méthode avec le logiciel *FreeSurfer* [39] montrant des différences notables en termes de qualité du modèle 3D reconstruit.

Nos jeux de tests correspondent à des modèles de données IRM synthétiques (présents dans le logiciel Slicer 3D [129]) et des modèles d’IRM réelles de cerveaux humains sains ( $brain_1, \dots, brain_{40}$ ), fournis par notre partenaire clinique. Le tableau 5.1 présente un échantillon représentatif des modèles utilisés, avec le nombre de faces correspondant et le temps total du pipeline. Toutes les images réelles sont acquises sur un Magnetom Skyra 3 Tesla (Siemens Healthineers, Erlangen, Allemagne). Les don-

nées originales sont enregistrées au format DICOM (Digital imaging and communications in medicine). En post-traitement, FSL-BET [150] est appliqué pour éliminer la graisse sous-cutanée afin de créer un masque du cerveau uniquement. Ensuite, FSL-FAST [171] est lancé pour obtenir les 3 classes de segmentation : (i) le cortex également connu sous le nom de matière grise, (ii) la matière blanche, (iii) le liquide céphalo-rachidien.

Les résultats présentés ont été produits avec un ordinateur équipé d'un processeur Intel core(TM) I9-9900 3.10 GHz, avec 8 threads et 96 Go de RAM. Le code a été complètement intégré dans *3D Slicer* [43] avec la bibliothèque *Moka* [159] comme noyau topologique.

Modèles	Nombre de faces $seg_1 / \dots / seg_n$	Temps total d'exécution
<i>Eye</i>	41 292 / 57 100	36s
<i>Kidney</i>	82 500 / 162 892	43s
<i>Brain</i>	504 068 / 71 800	1min 18s
<i>Brain<sub>1</sub></i>	1 023 696 / 1 508 240 / 884 424	11min 20s
<i>Brain<sub>2</sub></i>	1 004 228 / 1 497 588 / 869 552	10min 25s
<i>Brain<sub>3</sub></i>	899 480 / 1 390 964 / 852 804	8min 30s
<i>Brain<sub>4</sub></i>	959 608 / 1 563 308 / 974 416	11min 30s

TABLE 5.1 – Caractéristiques de notre ensemble de modèles de test et temps de calcul total du pipeline correspondant. La deuxième colonne indique la répartition des faces après l'étape de segmentation.

Model	# volumes reconstruits	Temps de reconstruction	# et pourcentage de volumes supprimés	Temps de classification et nettoyage	Total Time
<i>Eye</i>	62 / 328	2s / 5s	6 (10%) / 75 (23%)	2s / 1s	12s
<i>Kidney</i>	189 / 225	3s / 4s	74 (39%) / 102 (45%)	5s / 7s	25s
<i>Brain</i>	43 / 17	19s / 5s	4 (10%) / 2 (10%)	12s / 6s	1min 02s
<i>Brain<sub>1</sub></i>	399 / 623 / 579	48s / 1min 3s / 46s	0 / 141 (23%) / 94 (16%)	0s / 1min 22s / 41s	6min 32s
<i>Brain<sub>2</sub></i>	261 / 564 / 514	54s / 1min 24s / 50s	0 / 114 (20%) / 94 (18%)	0s / 1min 14s / 1min	6min 36s
<i>Brain<sub>3</sub></i>	365 / 704 / 386	47s / 1min 4s / 38s	0 / 211 (30%) / 89 (23%)	0s / 1min 32s / 46s	4min 47s
<i>Brain<sub>4</sub></i>	408 / 901 / 334	52s / 1min 26s / 46s	0 / 141 (15%) / 116 (35%)	0s / 1min 48s / 1min 26s	7min 21s

TABLE 5.2 – Informations quantitatives et temps d'exécution de chaque phase de l'étape 1.

Modèles	Processus Semantique/Topologique			# Voxels modifiés	Temps 5.2.5 total
	$Cas_1$	$Cas_2$	$Cas_3$		
<i>Eye</i>	31	0	569	717	17s
<i>Kidney</i>	22	0	569	441	18s
<i>Brain</i>	39	0	166	626	17s
<i>Brain<sub>1</sub></i>	142	1922	2286	1896	4min 34s
<i>Brain<sub>2</sub></i>	110	1959	1925	1993	3min 34s
<i>Brain<sub>3</sub></i>	144	1688	1840	1329	3min 27s
<i>Brain<sub>4</sub></i>	160	2765	2165	3599	3min 50s

TABLE 5.3 – Informations quantitatives et temps d'exécution de chaque phase de l'étape 2. Les voxels correspondant à chaque volume *unnamed* sont modifiés en fonction de leur cas sémantique/topologique.

### 5.3.1 Statistiques et performances

Le tableau 5.2 présente les résultats statistiques de la première étape de notre méthode : la reconstruction et le nettoyage d'un segment. On constate que les différentes étapes restent du même ordre de grandeur en terme de temps de calcul en fonction du nombre de faces composant le segment à reconstruire. Les variations lors de la phase de nettoyage sont dues à la qualité des données segmentées. Un modèle avec beaucoup de bruit est plus long à traiter, principalement à cause du calcul précis des volumes opéré avec la bibliothèque de tétraédricisation *tetgen* [77].

Le tableau 5.3 présente les résultats statistiques de la deuxième étape de la méthode : la fusion de tous les segments. La méthode détecte et traite plusieurs milliers d'incohérences en quelques minutes. La figure 5.24 illustre les différents cas de volumes vides détectés dans un modèle complet de cerveau reconstruit composé de plusieurs substances.

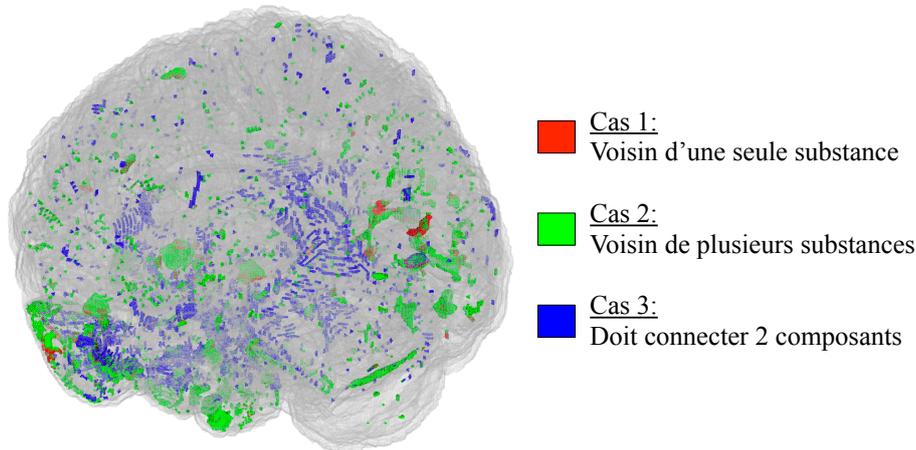


Figure 5.24 – Visualisation de la classification des volumes vides telle que décrite dans l'étape 2 dans un modèle complet de cerveau reconstruit composé de plusieurs substances.

### 5.3.2 Comparaison avec des méthodes de reconstruction classiques

Afin d'évaluer la performance de notre interpolation géométrique utilisant l'information probabiliste des voxels (PVE), nous comparons les modèles reconstruits pour un segment donné avec notre méthode et trois méthodes de reconstruction 3D différentes qui sont les suivantes :

- Flying Edges (FE) qui est utilisé dans notre pipeline. Il est intéressant d'étudier les différences entre FE et notre méthode une fois que l'ensemble du pipeline sera terminé ;
- Dual Marching Cubes (DMC) qui utilise un décideur asymptotique et remplace le maillage triangulaire par un maillage quadruple qui tend à éliminer les triangles mal formés présents occasionnellement dans les surfaces MC [69] ;
- Multi-level Partition of Unity (MPU) qui utilise des modèles implicites [26] et interprète les contours (c'est-à-dire la limite de segmentation à l'intérieur du volume).

La figure 5.25 présente une évaluation de la distance entre les centres des voxels de la segmentation initiale et les modèles de cerveau reconstruits pour les différentes méthodes. Ces mesures sont mesurées à l'aide de la distance de Hausdorff. Pour chaque modèle, une couleur verte est prédominante (distance équivalente à un demi voxel). En raison de sa méthode de construction qui tranche avec un facteur de 0,5 entre les voxels, le FE présente naturellement une carte de distance très homogène autour de 0,5. DMC produit une carte de distance moins homogène et la présence de quelques couleurs rouges et jaunes peut être remarquée, contrairement à FE. MPU présente également une dominance de couleurs rouges (grandes distances), indiquant une augmentation de la distance à la segmentation due à sa construction par une méthode implicite qui s'écarte plus significativement de la segmentation malgré un lissage de la surface 3D. Quant à notre méthode, elle montre une carte de distance légèrement plus proche de la segmentation, caractérisée par davantage de petites distances (cyan et bleu). Le morphing géométrique appliqué à la fin de notre pipeline permet à un ensemble non négligeable de points de se rapprocher de la surface segmentée, ce qui donne des résultats plus fidèles à l'anatomie.

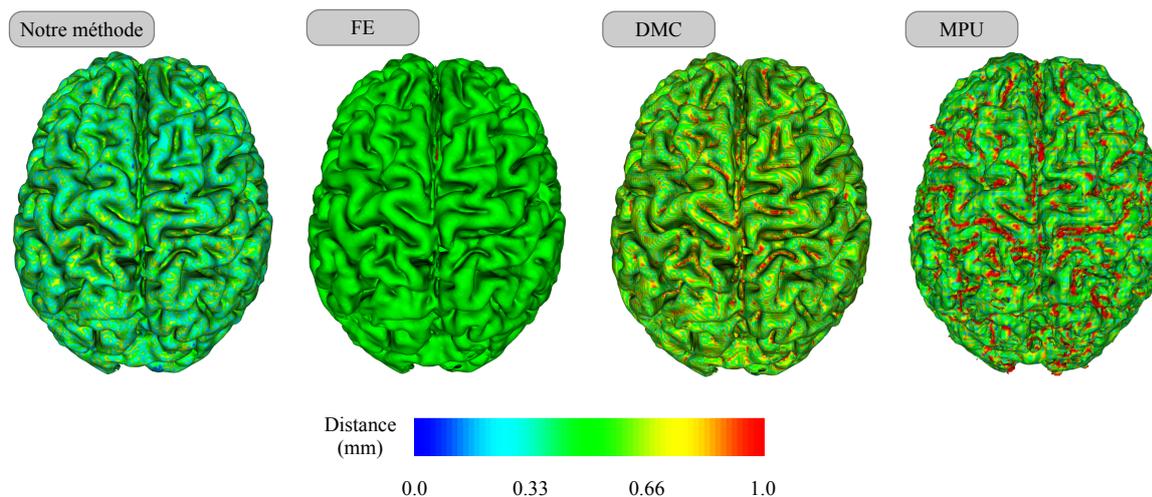


Figure 5.25 – Visualisation des différences entre la segmentation et plusieurs modèles 3D. Les différences sont calculées en utilisant la métrique de Hausdorff (le bleu reflète les distances faibles et le rouge les distances élevées).

Les résultats peuvent être approfondis en regardant les courbes de la figure 5.26 qui montre le pourcentage cumulé d'éléments en fonction de la distance (c'est-à-dire le ratio d'éléments qui sont, au maximum, à cette distance de la segmentation). Grâce à son mode de construction qui suit les voxels un par un, FE produit plus de 90 % des points de maillage à une distance de 0,5 mm, c'est-à-dire la moitié d'un voxel. En raison du lissage obtenu directement lors de sa construction, MPU produit des points légèrement plus éloignés. Grâce à sa méthodologie, DMC produit un maillage cohérent mais il rencontre des difficultés dans la gestion du bruit. Avant le morphing géométrique de notre pipeline (étape 3), notre méthode produit un résultat très similaire à FE en raison des corrections sémantiques effectuées lors de la deuxième étape. La courbe montrant le pourcentage cumulé des éléments en fonction de leur distance après morphing géométrique (en rouge) montre un cumul moins satisfaisant. En effet, puisque la troisième étape de notre méthode opère via des données voxel, le déplacement des points 3D n'est pas nécessairement effectué vers le centre de leur voxel associé. Le but de ces opérations est d'obtenir des résultats aussi proches que possible des données réelles, à une précision plus fine que celle utilisée par la segmentation.

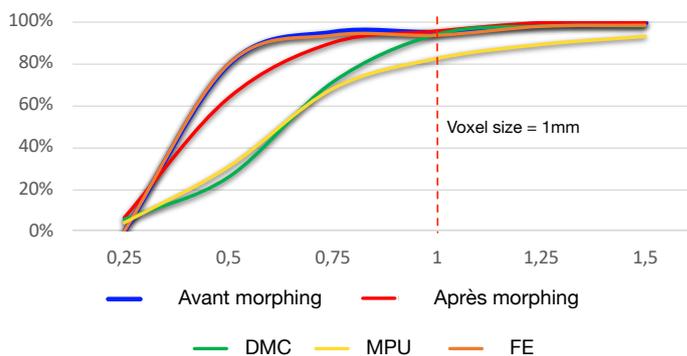


Figure 5.26 – Pourcentage cumulé des éléments en fonction de leur distance à la segmentation.

Au-delà de ces considérations géométriques, il faut noter que, contrairement à notre méthode, MPU et DMC ne sont pas directement utilisables pour regrouper et fusionner plusieurs volumes entre eux, en raison de leur système de construction qui ne produit pas de faces permettant une liaison topologique directe des volumes adjacents.

La figure 5.27 montre l'écart-type et la moyenne arithmétique du calcul de la distance entre l'objet et la segmentation effectuée sur notre jeu de données médicales réelles (quarante cerveaux). Notre méthode présente de meilleurs résultats indépendamment de la complexité du modèle ou du niveau de bruit.

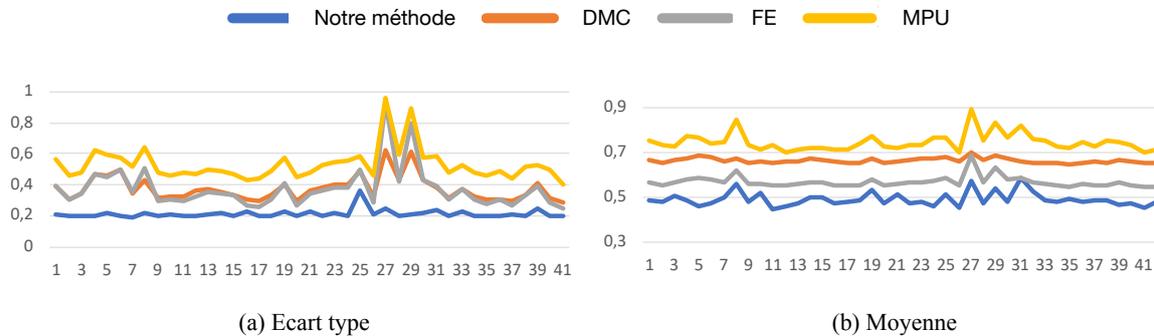


Figure 5.27 – L'écart-type (a) et la moyenne arithmétique (b) des valeurs d'erreur pour les quarante cortex cérébraux reconstruits avec notre méthode, FE, DMC et MPU.

### 5.3.3 Comparaison avec FreeSurfer

*FreeSurfer* (FS) est un logiciel pour l'étude de l'anatomie corticale et sous-corticale [39]. Actuellement, il constitue le principal logiciel permettant aux neuroscientifiques de tirer parti de la 3D [66, 67]. Son processus de reconstruction génère la surface limite entre la matière blanche et la matière grise corticale, ainsi que de la surface pial [45]. Les volumes sont créés en post-traitement pour étiqueter les classes de tissus sous-corticaux. Globalement, le modèle est reconstruit par un alignement volumétrique non linéaire de haute dimension, effectué sur un atlas donné [47].

Cette méthode de reconstruction ne produit qu'un seul modèle représentant le mélange d'interfaces corticales et sous-corticales contrairement à notre méthode qui discrimine tous les segments du cerveau. Un processus de nettoyage automatique du bruit est disponible en post traitement, mais il reste peu efficace et en cas de données trop bruitées, une intervention manuelle est nécessaire. En outre, notre méthode est plus performante en terme de temps de calcul, puisque plusieurs heures sont nécessaires à la reconstruction d'un modèle.

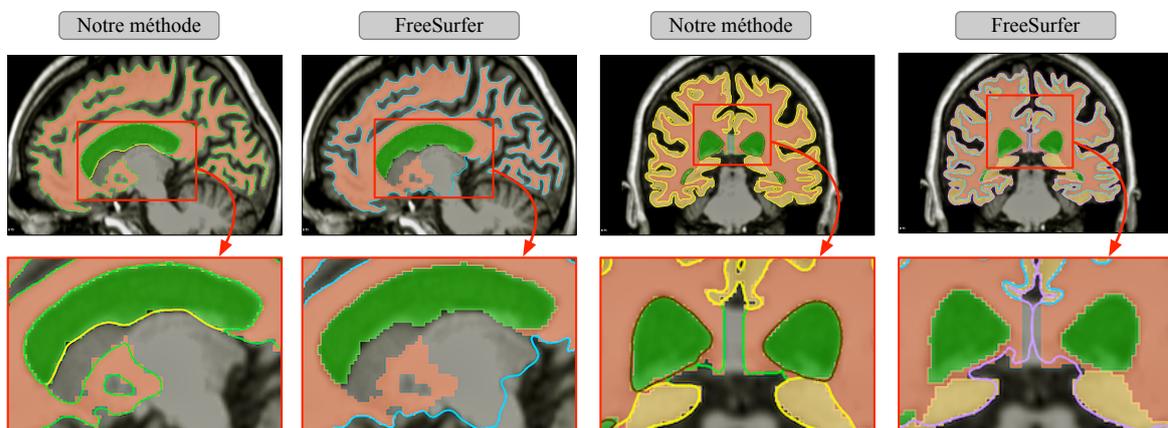


Figure 5.28 – Exemples de reconstruction de la matière blanche avec notre méthode en vert et FreeSurfer en bleu.

La reconstruction de la zone corticale est précise grâce à l'utilisation d'atlas de données. Cependant, la méthode de FreeSurfer étant basée sur une reconstruction indépendante de chaque hémisphère, une forte imprécision est présente entre les deux lobes (cf. figures 5.28 et 5.29). De plus, chaque substance étant construite indépendamment, il n'est pas possible de calculer facilement le voisinage topologique entre les mailles des différentes substances.

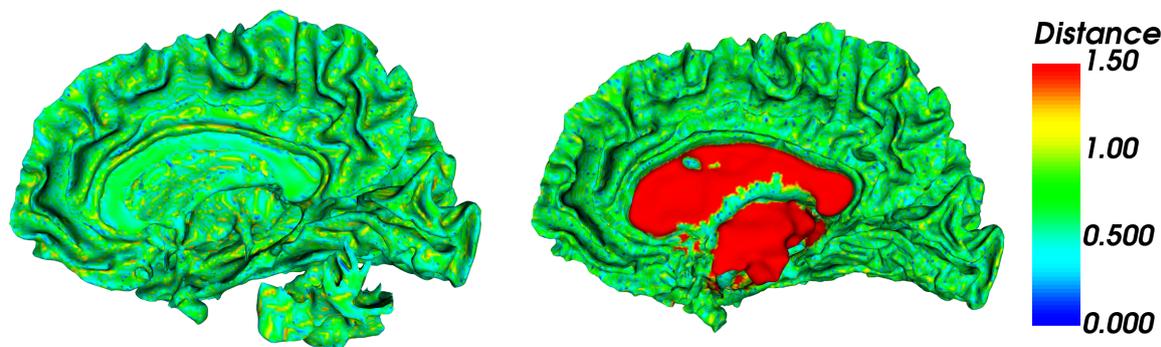


Figure 5.29 – Visualisation des surfaces de matière blanche reconstruites avec notre pipeline sur le côté gauche et avec FreeSurfer sur le côté droit. Pour chaque méthode de reconstruction, les surfaces sont codées par couleur en fonction de la distance absolue par rapport à la segmentation, du bleu (petite distance) au rouge (grande distance), en utilisant la métrique de Hausdorff.

## 5.4 Conclusion

Dans ce chapitre, nous avons tout d'abord présenté une méthode de reconstruction topologique de volumes à partir d'une soupe de polygones quelconque. Ce processus s'appuie sur la structure topologique des cartes généralisées. Il est composé de deux étapes majeures : la reconstruction des informations topologiques qui permettent de produire une subdivision cellulaire de l'espace et une opération de nettoyage/correction pour détecter et corriger les incohérences géométriques du modèle (éléments nuls, intersections). Les informations topologiques sont mises à contribution afin de réduire le nombre de tests effectifs et d'éviter la réapplication multiple des calculs d'intersections. Toutes ces opérations conduisent à satisfaire un modèle topologique spécifique avec des contraintes de cohérences qui assurent la qualité de l'objet produit. Enfin, les objets obtenus sont formés de volumes disjoints où toutes les informations topologiques et d'encastrement sont cohérentes. Ils peuvent ainsi être directement utilisés pour divers algorithmes de simulation.

Nous avons ensuite utilisé ce travail comme base pour proposer un nouveau pipeline dédié à la reconstruction de maillage volumétrique à partir d'acquisitions médicales 3D discrètes telle que des IRM. Ce processus permet de produire automatiquement, à partir de plusieurs régions d'intérêt segmentées, un modèle 3D cohérent et exempt d'espaces vides, en d'autres termes, une partition rigoureuse de l'espace. À cette fin, un processus topologiquement cohérent est appliqué pour fusionner efficacement chaque segment nettoyé en un seul modèle volumétrique global. Notre système inclut une correction topologique en utilisant des données sémantiques, suivies d'un morphing géométrique qui modifie la forme des segments reconstruits tout en respectant la topologie globale. De cette façon, il fournit aux praticiens un modèle 3D cohérent, aussi proche que possible des données réelles.

Nous avons mené un ensemble d'expérimentation à partir de données IRM de cerveaux humains sains, fournis par notre partenaire clinique. Les modèles 3D obtenus respectent les contraintes topologiques des différentes substances sur la base d'une identification sémantique cohérente et d'un modèle topologique formel, ce que les autres méthodes de reconstruction ne sont pas en mesure d'accomplir. De plus, notre méthode produit des modèles géométriquement plus proches des données réelles par rapport aux processus habituellement employés (FE, MPU ou DMC). Nous avons également réalisé une comparaison avec le logiciel *FreeSurfer*, le principal framework utilisé par les neuroscientifiques, et montré que notre reconstruction est plus précise et moins chronophage. De plus, notre méthode est en capacité de traiter des reconstructions de cortex cérébral fragmenté (par exemple en cas de tumeur avancée). De par son processus de reconstruction qui génère un volume unique par substance, *FreeSurfer* n'est pas en capacité de traiter ce genre de modèle.

La mise à disposition d'un modèle 3D représentant la globalité d'un cerveau, pourvu des informations sémantiques et topologiques ouvre de nombreuses perspectives dans le domaine transverse de la neuroscience et de l'informatique graphique. Ces différents points seront détaillés dans le chapitre 6.

## Publications relatives au chapitre

### Revue internationale

*Topological 3D Reconstruction Of Multiple Semantic Objects From Discrete Volumetric Medical Data* .  
S. Gerbaud, S. Horna, R. Zrour, P. Meseure .  
Computer Graphics Forum, 2022. (Soumission en avril 2022)

*A Hybrid Method for 3D Reconstruction of MR Images.*  
L. Lecheleck, S. Horna, R. Zrour.,  
Journal of imaging, 2022.Vol. 8

*Comparative Study of 3D Reconstruction Methods for Medical Imaging.*  
L. Lecheleck, S. Gerbaud, R. Zrour, S. Horna.  
Computer Aided Design And Application, 2021.Vol. 19

### Conférences internationales avec comité de lecture

*Reconstruction and topological cleaning of brain MR Images.*  
S. Gerbaud, S. Horna, R. Zrour, M. Naudin, C. Guillevin, P. Meseure.  
International Conference on Advances in Biomedical Engineering (ICABME), 2021.

*Comparative Study of 3D Reconstruction Methods for Medical Imaging.*  
L. Lecheleck, S. Gerbaud, E. Zrour, S. Horna.  
Computer Aided Design Conference, 2021.

### Conférences nationales sans comité de lecture

*Reconstruction topologique 3D de plusieurs objets sémantiques à partir de données médicales volumétriques discrètes* .  
S. Gerbaud, S. Horna, R. Zrour, P. Meseure .  
Journées Françaises de l'informatique Graphique, 2021.Nice - Sophia Antipolis

### Projets et collaboration

- 2019 : Porteur du projet Région MEDIA IRM 3D avec les CHU de Poitiers et Limoges, Siemens Healthineers, les universités de Poitiers, Limoges et La Rochelle, ainsi que les partenaires internationaux de l'université de Balamand (Liban) et de Heriot Watt University (Ecosse).
- 2018 : Projet région IMPROVED avec le CHU de Poitiers, les universités de Poitiers et La Rochelle.
- Collaboration avec le CHU de Poitiers dans le cadre de la thèse de Sylvain Gerbaud.

---

# CONCLUSION

---

Ce mémoire présente plusieurs travaux qui démontrent l'intérêt d'utiliser un modèle formel basé sur des propriétés géométriques, topologiques et sémantiques pour améliorer les opérations de modélisation et optimiser les méthodes de simulation. La structuration de l'espace et les relations de voisinage sont mises fortement à contribution pour contrôler l'environnement, localiser les traitements et mieux propager les données dans l'espace lors des calculs de simulation.

## Bilan

Le premier chapitre présente un modèle géométrique, topologique et sémantique dédié à la modélisation architecturale. Il peut être utilisé directement comme structure accélératrice pour une application de lancer de rayons interactif. Le parcours de la structure est simple et optimisé par une stratégie d'ordonnement des traitements guidée par les relations topologiques entre les primitives géométriques de la scène. Ce premier travail a mis en évidence l'intérêt de structurer et hiérarchiser les primitives géométriques en fonction de leur répartition spatiale. En effet, le mobilier est représenté par des modèles 3D composés de beaucoup de petites faces. Ces objets sont positionnés dans les bâtiments composés quand à eux de grandes faces. C'est pourquoi nous avons proposé de hiérarchiser notre modèle avec l'inclusion de volumes englobants dans une structure topologique globale.

Dans le deuxième chapitre, nous avons étendu ce sujet en proposant une nouvelle structure topologique et hiérarchique conçue pour améliorer la visualisation des bords arrondis avec une application de lancer de rayons : SREC-RT. La gestion de détails fins arrondis reste un problème difficile, car la visualisation des effets lumineux sur les chanfreins peut entraîner des incohérences visuelles : reflets lumineux, silhouette des objets, calcul d'ombre. Notre structure SREC-RT est construite automatiquement à partir de la géométrie d'une scène 3D. Sa mise en place n'est pas invasive (la scène originale reste conservée). SREC-RT repose sur une structure de détection hiérarchique et topologique qui permet d'une part de localiser efficacement les zones concernées par les chanfreins des objets, indépendamment de leur convexité, d'autre part de localiser le calcul du parcours des rayons au travers de la structure de détection. Notre structure offre une souplesse dans le choix de représentation géométrique des arrondis, nous avons proposé une solution basée sur des patches triangulaires quadratiques de Bézier pour les coins et des courbes de Bézier extrudées pour les bords. Cette représentation ne nécessite qu'une faible empreinte mémoire par rapport aux maillages de subdivision. De plus, le coût en temps de calcul reste équivalent aux méthodes usuelles qui utilisent des subdivisions pour modéliser les arrondis avec une qualité visuelle similaire.

L'utilisation de la topologie nous permet de proposer des processus plus élaborés pour la simulation d'éclairage avec une gestion fine de la géométrie. Toutefois, les informations topologiques ne sont pas systématiquement disponibles dans les objets 3D. Il est souvent nécessaire d'appliquer un pré-traitement qui définit les relations de voisinage. C'est pourquoi nous avons proposé dans le troisième chapitre une méthode de reconstruction topologique à partir d'une soupe de polygones. Cette application permet également de détecter et corriger les incohérences géométriques en s'appuyant sur le modèle topologique reconstruit. Nous avons ensuite mis à contribution ce processus dans le contexte de la reconstruction de modèles 3D à partir d'imagerie médicale. Nous avons ajouté une extension au logiciel Slicer 3D qui permet de produire automatiquement, à partir de plusieurs régions d'intérêt segmentées, un modèle 3D cohérent et exempt d'espaces vides, en d'autres termes, une partition rigoureuse de l'espace constituée de volumes fermés et liés topologiquement. Basé sur un ensemble de contraintes géométriques, topologiques et sémantiques, le modèle est suffisamment riche pour : réaliser des calculs statistiques de simulation de propagation de données métabolites (physiologiques); étudier la croissance de tumeurs cérébrales; visualiser les données multimodales.

## Projet scientifique

L'ensemble de mes travaux m'ont permis d'établir clairement un lien (topologique) entre les modèles de représentation issus de la modélisation et les structurations des données utiles en simulation d'éclairage. Même si les performances en termes de temps de traitement ne sont pas toujours compétitives par rapport aux méthodes classiques, l'utilisation des relations de voisinage et de la hiérarchisation de l'espace permet d'appliquer des traitements de manière beaucoup plus pertinente (gestion de détail, parcours de la lumière, localisation des calculs). Dans la suite de mes travaux, je souhaite globalement approfondir ce thème de recherche, d'une part en poursuivant l'utilisation des liens topologiques dans une structure hiérarchique pour des méthodes de visualisation, d'autres parts en étendant ce procédé à d'autres systèmes de simulation, par exemple dans le traitement de données médicales multimodales.

### Visualisation hiérarchique et topologique

Nos travaux concernant la visualisation de détails fins ont mis en évidence l'intérêt d'utiliser une structure hiérarchique et topologique pour réduire les traitements nécessaires au calcul de simulation d'éclairage. Notre méthode permet notamment d'appliquer des traitements spécifiques pour les rayons concernés par les arrondis. Les perspectives directes de ces travaux concernent l'utilisation de notre structure pour d'autres éléments de détail, par exemple du grillage, des particules en suspension, des fils fins (fils de pêche, toiles d'araignée, etc. ). Nous pouvons également viser à approfondir l'utilisation des informations topologiques et hiérarchiques de manière encore plus judicieuse pour guider le processus de lancé de rayons, notamment en étudiant la mise en corrélation entre la construction des chemins de lumière et de la répartition spatiale des détails fins.

Une première piste concerne l'utilisation de la structure hiérarchique pour adapter la géométrie en fonction de la localisation des petits éléments. Par exemple, il est possible de mettre en place un système de décimation de la géométrie en fonction de l'empreinte pixellaire des primitives correspondant aux détails fins. Par ailleurs, il est sans doute pertinent de mettre en place un système adaptatif dans le calcul des rebonds des chemins de lumière. En effet, la contribution des éléments fins est sans doute marginale dans les calculs des rayons secondaires. Pour l'ensemble de ces approches, la structure de donnée hiérarchique et topologique sera mis à contribution pour guider la gestion des niveaux de détails.

La répartition spatiale peut également être utilisée pour adapter l'échantillonnage des rayons primaires (le nombre de rayons par pixel). Une projection de notre structure sur l'espace-écran permet de localiser les zones de l'image nécessitant un nombre d'échantillons plus important. Il est par exemple utile d'envoyer plus de rayons dans les zones comportant des petits éléments (poussière en suspension, toile d'araignée). Par ailleurs, connaître la répartition spatiale et les relations de voisinage peut permettre d'estimer l'orientation des effets de lumière le long des surfaces. À la manière des méthodes de *manifold exploration* [86, 101], ce procédé guide la concentration des rayons pour améliorer la qualité visuelle des effets de réflexion.

Au delà de son utilisation pour des procédés de lancé de rayons, une structure hiérarchique peut également permettre d'adapter les calculs d'apparence des surfaces. Le travail de thèse d'Elsa Tamisier s'inscrit dans la problématique de la gestion des niveaux de détail et en particulier dans la prise en compte de la microgéométrie des surfaces en fonction du point de vue et son impact sur l'apparence des surfaces. L'analyse du masquage réalisée a permis de comprendre les modèles de visibilité de Smith et d'Ashikhmin [7] et d'identifier leurs limites.

Ces modèles ne permettent pas de calculer la visibilité exacte pour toutes les microgéométries. Nous souhaitons créer un nouveau modèle prédisant avec efficacité la visibilité de certaines microsursaces réelles. Pour cela, nous avons analysé les modèles de Smith et d'Ashikhmin dans un contexte de surfaces réelles pour établir des caractéristiques communes et effectuer une classification. Nous définissons ainsi une métrique d'erreur d'une microsursace vis-à-vis de chaque modèle. Notre objectif final est d'utiliser cette métrique dans un système de niveau de détail hiérarchique pour minimiser l'erreur de visualisation des surfaces rugueuses. Ce système s'insérera dans notre modèle de structuration de l'espace hiérarchique et topologique.

#### Traitement de données multimodales médicales

Le diagnostic préopératoire des tumeurs cérébrales est difficile, car l'anatomie humaine est complexe et la mine d'information mise à disposition par les méthodes d'acquisition n'est pas toujours facilement interprétable. Pour visualiser et explorer de telles données, il est essentiel de disposer de techniques pour distinguer les structures importantes et contrôler leur visibilité. L'IRM, grâce à ses séquences clés (diffusion, perfusion, spectroscopie), permet de prédire le grade tumoral<sup>1</sup>. La visualisation multi-modale de l'ensemble de ces données est un enjeu majeur car les médecins ont besoin de synthétiser de manière visuelle des informations provenant de diverses acquisitions, de faire des recoupements plus faciles entre les différentes données et ainsi de poser des diagnostics plus fiables pour prédire ou diagnostiquer l'apparition de certaines maladies comme Alzheimer, de certaines pathologies tumorales ou de troubles épileptiques.

Actuellement, les logiciels dont disposent les médecins proposent des visualisations 3D restreintes à une séquence d'acquisition sélectionnée, c'est-à-dire des visualisations 3D surfaciques de la partie osseuse ou des faisceaux de fibres. Des visualisations 3D volumiques sont également disponibles, (i) par principe de projection du maximum d'intensité des voxels rencontrés dans la direction du regard (l'algorithme MIP est essentiellement utilisé pour visualiser les vaisseaux sanguins en imagerie vasculaire), (ii) par principe de modulation de l'intensité du voxel affiché en fonction de sa distance (dans le depth culling [136], il est difficile de visualiser ce qui est loin de la caméra virtuelle, cela donne naissance à de fausses impressions d'espaces vides et par conséquent à de fausses interprétations), (iii) par application d'un seuil de transparence relatif à la densité des voxels rencontrés dans la direction du regard (utilisation de fausses couleurs par tissu afin de pouvoir les distinguer plus facilement lors de la visualisation par lancé de rayons [5]).

---

1. Le grade tumoral est la description d'une tumeur basée sur l'aspect anormal des cellules tumorales et du tissu tumoral. Il s'agit d'un indicateur de la vitesse à laquelle une tumeur est susceptible de se développer et de se propager.

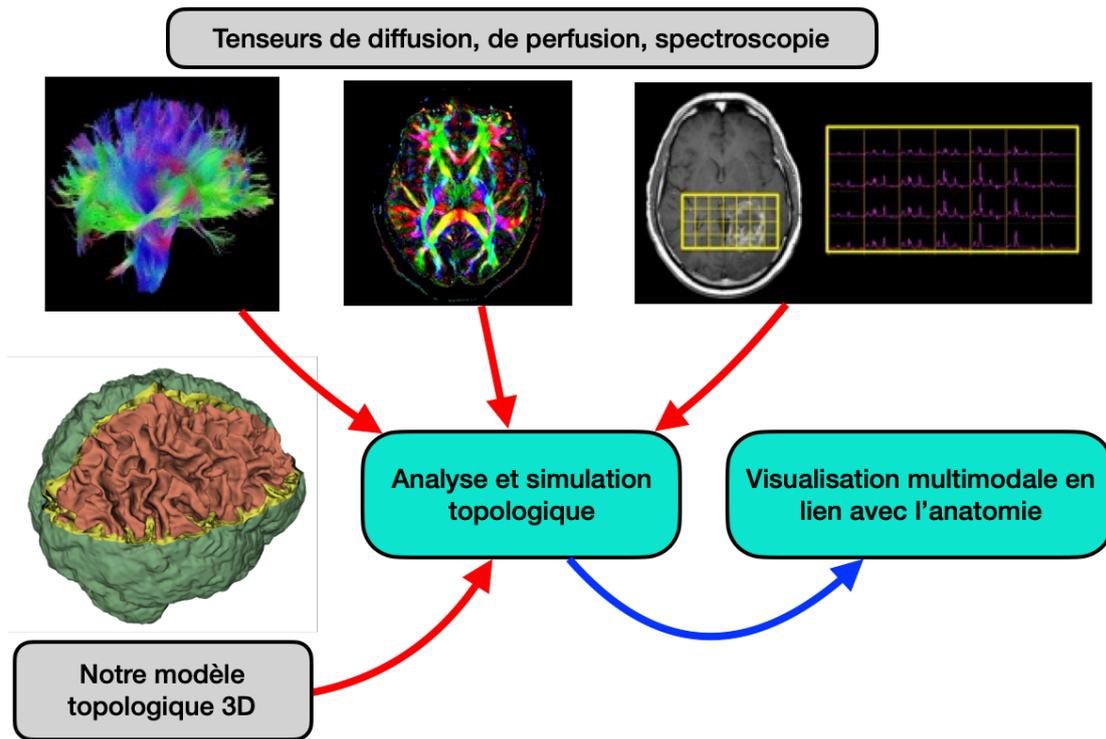


Figure 6.1 – Nous souhaitons mettre à contribution notre modèle 3D topologique pour recalculer la répartition des données dans le modèle continu, puis combiner et synthétiser les informations des différentes acquisitions.

Toutes ces méthodes de visualisation volumique n'utilisent pas de modèles géométriques en amont, elles travaillent directement sur les données IRM qui sont visualisées en utilisant l'accélération matérielle des cartes graphiques [125]. Malheureusement, aucun de ces logiciels n'est capable de proposer une vraie visualisation multi-modale combinant différentes séquences comme, par exemple, l'affichage des faisceaux de fibres qui traversent une tumeur. Les approches de type volumique permettent une bonne visualisation des structures mais posent néanmoins de nombreux problèmes pour la reconstruction de celles-ci, notamment en ce qui concerne la reconnaissance d'un organe particulier et non pas d'une structure générale (os, muscle, tissus aqueux, etc.) dont la caractérisation n'a été réalisée que selon des critères basiques de seuillage. De ce fait, il est important de proposer aux médecins des méthodes de reconstructions combinées, efficaces et précises leur permettant d'améliorer les diagnostics au quotidien grâce à un système de visualisation 3D plus pertinent.

À partir du modèle de données topologiques que nous avons proposé dans ce manuscrit, notre objectif principal est de développer un système permettant de recalculer la répartition des données dans le modèle continu, puis combiner et synthétiser les informations des différentes acquisitions (cf. figure 6.1). Nous souhaitons ainsi pouvoir proposer une représentation plus fine visualisable à partir d'indicateurs d'importance au choix du médecin. Une première étude expérimentale de ce processus s'inscrit pleinement dans le sujet de thèse de Hayssam Obeid, débutée en 2022. À terme nous souhaitons ajouter une dimension temporelle à ce système de visualisation pour étudier l'évolution des pathologies comme les glioblastomes. L'ensemble des informations recueillies par ce système permettra la mise en place d'un processus de simulation d'évolution des tumeurs cérébrales.

---

# COLLABORATIONS ET ENCADREMENTS

---

## Collaborations Nationales et Internationales

- 2021 : Projet région ARTERIA  
Partenaires : CHU de Poitiers, Siemens Healthineers, l'université de Poitiers.
- 2019 : Porteur du projet Région MEDIA IRM 3D  
Partenaires : CHU de Poitiers et Limoges, Siemens Healthineers, les universités de Poitiers, Limoges et La Rochelle, ainsi que les partenaires internationaux de l'université de Balamand (Liban) et de Heriot Watt University (Ecosse).  
Cotutelle de thèse : université de Balamand au Liban (Dr. Mohammad Khalil)
- 2018 : Projet région IMPROVED  
Partenaires : CHU de Poitiers, les universités de Poitiers et La Rochelle.
- 2018 : Projet région SIMOREVA  
Partenaires : universités de Poitiers (XLim), Montréal (LIGUM), l'INRIA, le L2PN de l'Institut d'Optique et le NTNU colorlab en Norvège  
Cotutelle de thèse : université de Montréal, laboratoire LIGUM (Dr : Pierre Poulin)

Pour des questions liées à la rapidité de production, les problématiques de construction de maquettes virtuelles 3D et simulation sont primordiales dans le milieu professionnel. Les thématiques de mes travaux de recherche se placent au coeur de plusieurs collaborations existantes notamment dans les contextes métiers de la géo-modélisation (Géosiris et Total), la visualisation 3D et le rendu réaliste.

- Geosiris (contrat équipe conseil de 40 Keuros, 2014 - 2016) :  
Geosiris offre des services et des nouvelles technologies dans le domaine de géomodélisation (à partir de l'interprétation de données sismiques). L'objectif principal de la collaboration avec XLim-ASALI est de proposer un modéleur qui permet de produire un modèle structurel de données en gardant toutes les informations topologiques entre les surfaces géologiques.

- Savane Vision (contrat équipe conseil de 15 Keuros, du 1er janvier 2014 au 30 juin 2015) :  
L'objectif du projet est de mettre en place une chaîne de traitements permettant à un utilisateur non spécialiste en 3D de concevoir des maquettes virtuelles de plusieurs pièces d'un logement pour leur rénovation. Les maquettes virtuelles doivent être construites de manière intuitive en 2D, puis portées en 3D automatiquement, et visualisées de façon réaliste sur le terminal mobile de l'utilisateur. Le laboratoire XLim-Sic (représenté par D. Meneveaux et moi-même) a pour objectif de guider les développements en apportant son expertise dans les domaines de la modélisation et le rendu réaliste. Notre contribution a permis à l'entreprise Savane vision de remporter plusieurs appels de levé de fonds : concours crea'Vienne, BpiFrance, jeune pépite (environ 120Keuros)
- Equilibre Games (contrat équipe conseil de 15 Keuros, de mars à septembre 2014) :  
Equilibre Games est un studio de développement et d'expertise technique (jeux vidéo et outils) soutenu par l'incubateur image d'Angoulême. A. Arnould, H. Ferrier-Belhaouari et moi-même intervenons comme encadrants et conseillers scientifiques pour le développement de l'outil intitulé « a Maze Kit » créé en nom propre par Equilibre Games. Outil de création et d'édition de parcours et cheminement 3D, « a Maze Kit » a été développé dans le but d'automatiser la conception de niveaux de jeux vidéo.
- Financements Action - MIRES  
La fédération MIRES - Mathématiques & leurs Interactions, Images & information numérique, Réseaux et Sécurité - est une fédération regroupant les laboratoires en mathématiques et sciences du numérique de Poitiers, Limoges et la Rochelle. Elle est labellisée par le CNRS depuis janvier 2012. MIRES permet le financement des collaborations entre les chercheurs des différents laboratoires de la fédération. Ces financements sont obtenus par appel à projet pour une durée de 1 an renouvelable. Entre 2012 et 2017, j'ai participé aux actions MESA (modélisation géométrique et topologique) et AGIME (calculs de visibilité et algèbres géométriques). Depuis 2018, j'ai été porteur des actions Mart et VisuMed qui ont permis la mise en relation des partenaires du projet Media IRM 3D.

### Encadrements de thèses et post doctorats

- Maxime Maria (thèse soutenue en 2016).  
Utilisation d'une Partition spatiale contrainte et convexe pour la simulation basée rayons,  
Directeur : L. Aveneau (directeur), co-directeur : **S. Horna (ACT)**.
- Simon Courtin (thèse débutée en 2018 - soutenance prévue été 2022).  
Prise en compte des effets haute fréquence en simulation d'éclairage,  
Directeur : D. Meneveaux, co-directeur : **S. Horna (ACT)**, encadrant : M. Ribardièrre.
- Elsa Tamisier (thèse débutée en 2019 en cotutelle avec l'université de Montréal).  
Représentations géométriques de détails fins pour la simulation d'éclairage  
Directeurs : D. Meneveaux et P. Poulin, encadrants : M. Ribardièrre et **S. Horna**.
- Loubna Lecheleck (post doctorat de 18 mois 2020 - 2021).  
Reconstruction de modèles 3D à partir de données médicales.  
Encadrants : **S. Horna** et R. Zrour
- Sylvain Gerbaud (thèse débutée en 2020)  
Développement de nouvelles méthodes de reconstructions tissulaires basées sur le métabolisme sous-jacent exploré en spectroscopie par résonance magnétique multi-noyaux,  
Directeur : P. Meseure, co-directeur : **S. Horna (ACT)**, encadrant : R. Zrour.
- Hayssam Obeid (thèse débutée en 2021 en cotutelle avec l'université Libanaise).  
Visualisation et interaction avec les informations anatomiques et métaboliques  
Directeurs : P. Meseure et M. Kalil, encadrants : R. Zrour et **S. Horna**.

### Encadrements de stages et projets de licences et Masters

L'objectif de ce type d'exercice est de mettre en situation un ou plusieurs étudiants sur un sujet en relation avec les travaux réalisés au laboratoire. Il permet de mettre en place en situation réelle les bonnes pratiques en vigueur dans le domaine de la gestion de projet, via la réalisation d'un projet d'envergure (environ 13 semaines à temps complet en M2). Il s'agit de choisir et de décliner, en fonction de contraintes opérationnelles, une méthodologie de planification, d'évaluation des risques et des coûts afin de respecter l'ensemble des attendus d'un projet réel d'informatique.

J'ai l'opportunité de proposer des sujets en lien direct avec mes travaux de recherche : modélisation géométrique à base topologique, rendu réaliste, visualisation interactive. C'est aussi pour moi l'occasion de travailler avec les étudiants en fin de cursus informatique et de participer à leur formation en prenant le rôle de tuteur et/ou client sur cet exercice de gestion de projet et de développement.

Depuis 2010, j'ai suivi 17 projets ou stages de la L3 au M2 informatique. Ces sujets s'articulent autour des projets de l'équipe d'Informatique Graphique de XLim - ASALI. Notamment avec l'ajout de nouveaux modules au modèleur géométrique à base topologique MOKA

Stage M2 Recherche	
2020	Sylvain Gerbaud - 6 mois - XLim ASALI - 50% encadrement
2019	Elsa Tamisier - 6 mois - XLim ASALI - 50% encadrement
2018	Simon Courtin - 6 mois - XLim ASALI - 50% encadrement
2014	Viviane Jaraczewski - 6 mois - XLim ASALI et Equilibres Games - 40% encadrement
2012	Maxime Maria - 6 mois - XLim ASALI - 50% encadrement

Projets de fin d'étude en lien avec la recherche	
2021	Stage de L3 - 2 mois : Mise en oeuvre et analyse à partir du logiciel freeSurfer
2021	Stage de L3 - 2 mois : Création de site web
2020	Stage de L3 - 2 mois : Analyse comparative des méthodes de reconstruction 3D
2019	Projet de M2 - 6 mois : Reconstruction 3D d'images IRM
2019	Stage de L3 - 2 mois : Analyse de données médicales/Fichier DICOM
2018	Projet de M2 - 6 mois : IHM pour une application de synthèse d'images réalistes
2017	Projet de M2 - 6 mois : Construction procédurale topologique de cathédrales
2017	Projet de M2 - 6 mois : Moteur de rendu pour le modèleur Moka
2016	Stage de M1 - 2 mois : Construction procédurale d'objets complexes
2016	Stage de M1 - 2 mois : Lancé de rayons topologiques
2015	Stage de L3 - 2 mois : Construction d'objets complexes
2015	Projet de M1 - 2 mois : Étude des structures accélératrices en synthèse d'images
2014	Projet de M2 - 6 mois : Modèleur Architectural pour les Archéologues (RUBBLE)
2014	Projet de M1 - 2 mois : Lancé de rayons topologiques
2013	Projet de M2 - 6 mois : Modèleur topologique pour la géologie (GeoTopoModeler)
2013	Projet de M2 - 6 mois : Modèleur Architectural pour les Archéologues (RUBBLE)
2012	Stage de L3 - 2 mois : Modèleur Architectural à base de règles (RUBBLE)
2012	Projet de M2 - 6 mois : Modèleur topologique pour la géologie (GeoTopoModeler)
2011	Projet de M2 - 6 mois : Modèleur topologique pour la géologie (GeoTopoModeler)
2011	Projet de M1 - 2 mois : Visualisation de carte de hauteur



---

## PUBLICATIONS PERSONNELLES

---

### Revues internationales [8]

*Topological 3D Reconstruction Of Multiple Semantic Objects From Discrete Volumetric Medical Data .*  
S. Gerbaud, S. Horna, R. Zrour, P. Meseure .

Computer Graphics Forum, 2022. (Soumission en avril 2022)

*A Hybrid Method for 3D Reconstruction of MR Images.*

L. Lecheleck, S. Horna, R. Zrour, .

Journal of imaging, 2022. Vol. 8

*Comparative Study of 3D Reconstruction Methods for Medical Imaging.*

L. Lecheleck, S. Gerbaud, R. Zrour, S. Horna.

Computer Aided Design And Application, 2021. Vol. 19

*SREC-RT : A Structure for Ray Tracing Rounded Edges and Corners.*

S. Courtin, S. Horna, M. Ribardière, P. Poulin, D. Meneveaux. .

Computer Graphics Forum , 2021, pp. Volume 40, pages 200-214.

*Reconstruction of Volumes From Soup of Faces with formal Topological Approach.*

H. Belhaouari, S. Horna.

Computer Aided Design And Application , 2019, pp. Volume 16, pages 972-984.

*Efficient Rendering of Rounded Corners and Edges for Convex Objects.*

S. Courtin, S. Horna, M. Ribardière, P. Poulin, D. Meneveaux.

Lecture Notes in Computer Science - Advances in Computer Graphics. Proceedings of CGI 2019, 2019, pp. Volume 11542.

*Constrained Convex Space Partition for ray-tracing.*

M. Maria, S. Horna, L. Aveneau.

Computer Graphics Forum, 2015, pp. Volume 36, pages 288-300.

*Consistency constraints and 3D building reconstruction.*

S. Horna, D. Meneveaux, G. Damiand, Y. Bertrand.

Computer Aided Design (CAD), 2009, pp. Volume 41, pages 13-27.

## Brevet [1]

*Isometric Unfolding of Stratigraphic Grid Units for Accurate Property Populating.*  
S.Horna, C. Bennis, J. F. Rainaud.  
Brevet industriel déposé à IFP Energie Nouvelles , 2011.

## Conférences internationales avec comité de lecture [12]

*Reconstruction and topological cleaning of brain MR Images.*  
S. Gerbaud, S. Horna, R. Zrour, M. Naudin, C. Guillevin, P. Meseure.  
International Conference on Advances in Biomedical Engineering (ICABME), 2021.

*Comparative Study of 3D Reconstruction Methods for Medical Imaging.*  
L. Lecheleck, S. Gerbaud, E. Zrour, S. Horna.  
Computer Aided Design Conference, 2021.

*Efficient Rendering of Rounded Corners and Edges for Convex Objects.*  
S. Courtin, S. Horna, M. Ribardière, P. Poulin, D. Meneveaux.  
Computer Graphics International, (CGI), 2019.

*Reconstruction of Volumes From Soup of Faces with formal Topological Approach.*  
H. Belhaouari, S.Horna.,  
Computer Aided Design Conference, 2018.

*Efficient Ray Traversal of Constrained Delaunay Tetrahedralisation.*  
M. Maria, S.Horna, L. Aveneau.  
Conference on Computer Graphics Theory and Applications (GRAPP), 2017.

*A topological approach for automated unstructured meshing of complex reservoir.*  
V. Gauthier, A. Arnould, H. Belhaouari, S. Horna, M. Perrin, M. Poudret, J. F.Rainaud.  
European Conference on the Mathematics of Oil Recovery (ECMOR), 2015.

*Topological Space Partition for Fast Ray Tracing in Architectural Models.*  
S. Horna, G. Damiand, A. Diakité, D. Meneveaux.  
Eurographics Workshop on Urban Data Modelling and Visualisation (UDMV), 2014, pp. 13–18.

*Topological Space Partition for Fast Ray Tracing in Architectural Models.*  
M. Maria, S.Horna, L. Aveneau.  
Conference on Computer Graphics Theory and Applications (GRAPP), 2014.

*Isometric Unfolding of Stratigraphic Grid Units for Accurate Property Populating.*  
S. Horna, C. Bennis, H. Bourouchaki, C. Delage, J.F. Raynaud.,  
European Conference on the Mathematics of Oil Recovery (ECMOR), 2010.

*Extracting and Unfolding a Stratigraphic Unit to Update Property Population.*  
S. Horna, C. Bennis, T. Crabie, S. Peltier, J.F. Raynaud, .  
European Association of Geoscientists and Engineers Conference (EAGE), 2010.

*Building 3D indoor scenes topology from 2D architectural plans.*  
S. Horna, G. Damiand, D. Meneveaux, Y. Bertrand.  
Conference on Computer Graphics Theory and Applications (GRAPP), 2007, pp. 37–44.

*Out-of-core photon-mapping for large buildings.*  
D. Fradin, D. Meneveaux, S. Horna.  
Proceedings of Eurographics Symposium on Rendering (EGSR), 2005, pp. 65–72.

## Conférences nationales sans comité de lecture [9]

*Reconstruction topologique 3D de plusieurs objets sémantiques à partir de données médicales volumétriques discrètes .*

S. Gerbaud, S. Horna, R. Zrour, P. Meseure .

Journées Françaises de l'informatique Graphique, 2021.Nice - Sophia Antipolis

*Arrondis et simulation d'éclairage basée rayons : une structure dédiée aux surfaces implicites.*

S. Courtin, S. Horna, M. Ribardière, P. Poulin, D. Meneveaux .

Journées Françaises de l'informatique Graphique, 2019.Marseille - 2<sup>eme</sup> meilleur papier

*Visualisation interactive de grands bâtiments.*

M. Maria, S.Horna, L. Aveneau.

Journées Françaises de l'informatique Graphique, 2013.Limoges - Meilleur papier

*Visualisation interactive de grands bâtiments.*

M. Maria, S.Horna, L. Aveneau.

Groupe de Travail Rendu, 2013.Paris

*Reconstruction d'environnements architecturaux 3D.*

S. Horna.

Exposé invité, Ecole thématique MoDys, 2012.Marseille

*S.Horna, G.Damiand, D.Meneveaux, Y.Bertrand.*

Reconstruction topologique 3D de bâtiments.

Journées Françaises de l'informatique Graphique, 2006.Bordeaux

*Reconstruction de complexes architecturaux à partir de données numériques.*

S.Horna, D.Meneveaux, G.Damiand.

École thématique de modélisation informationnelle pour l'architecture patrimoniale, 2006.

*Reconstruction topologique de complexes architecturaux à partir de données numériques.*

S.Horna, G.Damiand, D.Meneveaux, Y.Bertrand.

Journée du GTMG, 2006.

*Simulation d'éclairage par tracé de photons pour de grands bâtiments.*

D. Fradin, D. Meneveaux, S. Horna.

Groupe de Travail CNRS, Bavay-Valenciennes, 2005.



---

## RESPONSABILITÉS

---

En parallèle de mes activités de recherche et d'enseignement, je me suis investi dans différentes responsabilités collectives tant au sein de mon équipe de recherche (XLIM - ASALI), qu'au sein de ma composante de rattachement (la faculté de Droit).

### **Responsable financier de l'équipe IG du laboratoire XLim-ASALI**

Pendant 8 ans, j'ai assuré la gestion financière de l'équipe IG du laboratoire XLim-ASALI (17 enseignants-chercheurs permanents – budget dédié de 40Keuros/an auquel s'ajoutent les divers projets industriels et nationaux). Cette responsabilité m'a amené à suppléer le responsable (D. Meneveau de 2012-2015, L. Aveneau de 2016-2017, D. Meneveau 2018-2019) dans la supervision de l'équipe notamment dans l'organisation des missions, des achats de matériel et du financement de personnels (stages, ingénieurs, etc.). Cette responsabilité m'a permis de m'impliquer très fortement dans le montage financier des projets (ANR, Région, etc.) et dans la rédaction de contrats de collaboration entre l'équipe et les industriels.

Je suis également très investi dans la gestion globale de XLim-SIC, puis XLim-ASALI pour des questions liées aux finances du département (budget de 80Keuros/an), ainsi que pour des questions d'organisation scientifique plus générales en raison de ma position de représentant du corps MC au conseil de département (de 2013 à 2016), puis membre du conseil scientifique de l'axe ASALI depuis 2019.

### **Responsabilités scientifiques locales et nationales**

- Depuis janvier 2021 : Chargé de la communication de la fédération MIRES Mathématiques & leurs Interactions, Images & information numérique, Réseaux et Sécurité.
- Depuis janvier 2021 : Chargé de la communication du Laboratoire Commun I3M Imagerie Métabolique Multi-noyaux Multi-organes.
- Depuis 2019 : Membre élu au conseil de l'axe de recherche XLim-ASALI
- Depuis 2018 : Membre élu au CA de l'Association Française d'Informatique Graphique
- 2012-2019 : Responsable financier de l'équipe IG du laboratoire XLim-ASALI
- 2013-2022 : Membre élu à la CES 27 de l'université de Poitiers (V. Président corps MC)
- 2013-2016 : Membre élu au conseil de département Xlim-SIC (représentant corps MC)

## Responsabilités pédagogiques

Depuis 2010, j'effectue des enseignements en Licence et Master au département informatique de la faculté de Droit et Sciences Sociales de l'université de Poitiers. En première et deuxième année de Licence de Droit, les objectifs sont d'apprendre et comprendre l'utilisation des outils informatiques : gestion de l'environnement, utilisation d'Internet, licences logicielles, bons usages en termes de sécurité, outils bureautiques. En troisième année de Licence et Masters Droit du numérique, les connaissances sont approfondies notamment concernant le codage, le matériel informatique et le traitement des données.

### Responsabilité au sein de la faculté de Droit

Au fil des années, j'ai progressivement pris en charge la responsabilité complète des unités d'enseignements transverses (S2, S3 et S6 de licence Droit et AES). Les effectifs importants de ces formations (900 L1, 600 L2), impliquent une logistique conséquente : gestion des examens, des corrections, des absences, coordination des intervenants (entre 2 et 6 chargés de TD par UE), organisation des plannings de TD répartis sur 2 sites soit 30 groupes de TD en 1<sup>re</sup> année entre Poitiers et Angoulême. De plus, depuis la mise en place de l'offre de formation de 2018, cette unité inclut le numérique, la recherche documentaire et le Projet Professionnel de l'Étudiant (PPPE). La responsabilité de ces unités d'enseignement comprend donc également la coordination entre le numérique et les autres enseignements dispensés par des bibliothécaires et des juristes (académiques ou intervenants extérieurs).

Depuis le début de ma carrière d'enseignant-chercheur, je prête une attention particulière à m'investir dans l'organisation pédagogique. En effet, je pris part peu à peu à l'organisation logistique des 3 niveaux de licence. Ainsi, depuis 2016, j'épaule pleinement les responsables de formation dans la répartition des groupes de TD, la gestion des options, la construction de l'emploi du temps. C'est une organisation complexe en raison du grand nombre de parcours et d'options disponibles, du faible nombre des salles (notamment en numérique) et des effectifs conséquents concernés.

### Responsable de la commission pédagogique de numérique de l'université de Poitiers. (Depuis 2019)

Dans la mise en place de l'offre de formation 2018-2021, l'université de Poitiers a souhaité donner une impulsion forte à l'acquisition des compétences transversales et préprofessionnelles en donnant un cadre Établissement aux UE transversales de la licence générale. La maquette mise en place propose ainsi une unité d'enseignement transverse (UE6) commune pour toutes les licences générales au semestre (S1, S2 et S3). Le contenu de formation a été réfléchi afin de permettre à tous les étudiants d'acquérir les compétences leur permettant de passer la certification nationale PIX.

Ayant été un acteur majeur dans la mise en place de ce dispositif, et possédant une expérience reconnue dans l'enseignement des enseignements transversaux à la faculté de droit, il m'a été proposé de prendre en charge le pilotage de la commission pédagogique de numérique de l'université de Poitiers. Cette commission a pour rôle d'organiser et d'harmoniser la mise en place, de favoriser les échanges et de créer des contenus dans le cadre des enseignements transverses (numérique et recherche documentaire). L'objectif est de proposer un socle d'enseignement en numérique commun pour l'ensemble des composantes universitaires concernées par le cursus de licence générale (6 composantes, soit 8000 étudiants concernés). Cette commission est composée de représentants de chaque composante (Sciences Fondamentales, Lettre et langues, Droit, Sciences de l'Homme, Sciences économiques et Sciences du sport), ainsi que des représentants des services d'informatique et des services communs de documentation. Le

rôle de la commission est de travailler sur l'organisation et le contenu pédagogique des matières de numérique et recherche documentaire pour les licences générales. Elle met en place l'articulation entre les enseignements dispensés et la certification PIX nationale.

Mon rôle à la tête de cette commission consiste à organiser et à piloter l'ensemble des réunions, coordonner les différents services concernés (informatique, département d'enseignement, etc.), assurer l'application des décisions dans les diverses composantes, aider au recrutement des intervenants extérieurs, planifier les certifications, assurer le lien avec l'équipe présidentielle de l'université de Poitiers. C'est une responsabilité lourde (plus de 50 h de référentiel).



---

---

## BIBLIOGRAPHIE

---

- [1] Mira Geoscience Ltd 2017.
- [2] Timo AILA et Samuli LAINE : Understanding the efficiency of ray traversal on gpus. In Proceedings of the conference on high performance graphics 2009, pages 145–149, 2009.
- [3] Timo AILA, Samuli LAINE et Tero KARRAS : Understanding the efficiency of ray traversal on GPUs – Kepler and Fermi addendum. NVIDIA Technical Report NVR-2012-02, NVIDIA Corporation, juin 2012.
- [4] John M AIREY, John H ROHLF et Frederick P BROOKS JR : Towards image realism with interactive update rates in complex virtual building environments. ACM SIGGRAPH computer graphics, 24(2):41–50, 1990.
- [5] Anod ALHAZMI et Sudhanshu SEMWAL : 3d volume visualization in medical application. In Proceedings of the International Conference on Modeling, Simulation and Visualization Methods (MSV), pages 58–64. The Steering Committee of The World Congress in Computer Science, 2018.
- [6] Aytek AMAN, Serkan DEMIRCI et Uğur GÜDÜKBAY : Compact tetrahedralization-based acceleration structure for ray tracing. arXiv preprint arXiv :2103.02309, 2021.
- [7] Michael ASHIKMIN, Simon PREMOŽE et Peter SHIRLEY : A microfacet-based brdf generator. In Proceedings of the 27th annual conference on Computer graphics and interactive techniques, pages 65–74, 2000.
- [8] Marco ATTENE, Marcel CAMPEN et Leif KOBELT : Polygon mesh repairing : An application perspective. ACM Computing Surveys (CSUR), 45(2):1–33, 2013.
- [9] M. BABA-ALI, G. DAMIAND, X. SKAPIN et D. MARCHEIX : Insertion and expansion operations for n -dimensional generalized maps. In Proceedings of 14th IAPR International Conference on Discrete Geometry for Computer Imagery, volume 4992 de LNCS, pages 141–152. Springer-Verlag Berlin Heidelberg, April 2008.
- [10] Chandrajit L BAJAJ, Edward J COYLE et Kwun-Nan LIN : Arbitrary topology shape reconstruction from planar cross sections. Graphical models and image processing, 58(6):524–543, 1996. <https://doi.org/10.1006/gmip.1996.0044>.
- [11] Miguel Angel Gonzalez BALLESTER, Andrew P ZISSERMAN et Michael BRADY : Estimation of the partial volume effect in mri. Medical image analysis, 6(4):389–405, 2002.
- [12] Bruce G BAUMGART : Winged edge polyhedron representation. Rapport technique, STANFORD UNIV CA DEPT OF COMPUTER SCIENCE, 1972.
- [13] Bruce G BAUMGART : A polyhedron representation for computer vision. In Proceedings of the May 19-22, 1975, national computer conference and exposition, pages 589–596, 1975.

- [14] H. BELHAOUARI, A. ARNOULD, P. LE GALL et T. BELLET : JERBOA : A graph transformation library for topology-based geometric modeling. In 7th International Conference on Graph Transformation (ICGT 2014), volume 8571 de LNCS, York, UK, juillet 2014. Springer.
- [15] Jon Louis BENTLEY : Multidimensional binary search trees used for associative searching. Communications of the ACM, 18(9):509–517, 1975.
- [16] Stephan BISCHOFF et Leif KOBBELT : Structure preserving cad model repair. Computer Graphics Forum, 24(3):527–536, Sep 2005.
- [17] Stephan BISCHOFF, Darko PAVIC et Leif KOBBELT : Automatic restoration of polygon models. ACM Transactions on Graphics, 24(4):1332–1352, Oct 2005.
- [18] BLENDER FOUNDATION : Blender shader for round edges. <https://wiki.blender.org/wiki>.
- [19] BLENDER FOUNDATION : Subdivision tool in Blender. [www.docs.blender.org/manual](http://www.docs.blender.org/manual).
- [20] BLENDER FOUNDATION : Blender. [www.blender.org](http://www.blender.org), 2018.
- [21] BLENDER FOUNDATION : Cycles. [www.cycles-renderer.org](http://www.cycles-renderer.org), 2018.
- [22] Jean-Daniel BOISSONNAT : Shape reconstruction from planar cross sections. Computer vision, graphics, and image processing, 44(1):1–29, 1988. [https://doi.org/10.1016/S0734-189X\(88\)80028-8](https://doi.org/10.1016/S0734-189X(88)80028-8).
- [23] Mario BOTSCH, Leif KOBBELT, Mark PAULY, Pierre ALLIEZ et Bruno LÉVY : Polygon mesh processing. CRC press, 2010.
- [24] Sylvain BRANDEL, Sebastien SCHNEIDER, Michel PERRIN, Nicolas GUIARD, Jean-François RAINAUD, Pascal LIENHARD et Yves BERTRAND : Automatic building of structured geological models. Journal of Computing and Information Science in Engineering, 5(2):138–148, Feb 2005.
- [25] Sylvain BRANDEL, Sebastien SCHNEIDER, Michel PERRIN, Nicolas GUIARD, Pascal RAINAUD, Jean Francois Lienhardt et Yves BERTRAND : Automatic building of structured geological models. 2005. Journal of Computing and Information Science in Ingeneering, Volume 5.
- [26] Ilya BRAUDE, Jeffrey MARKER, Ken MUSETH, Jonathan NISSANOV et David BREEN : Contour-based surface reconstruction using mpu implicit models. Graphical models, 69(2):2007, 139–157. <http://doi.org/10.1016/j.gmod.2006.09.007>.
- [27] Erik BRISSE : Representing geometric structures ind dimensions : Topology and order. Discrete & Computational Geometry, 9(4):387–426, 1993.
- [28] Rickard BRÜEL-GABRIELSSON, Vignesh GANAPATHI-SUBRAMANIAN, Primoz SKRABA et Leonidas J GUIBAS : Topology-aware surface reconstruction for point clouds. In Computer Graphics Forum, volume 39, pages 197–207. Wiley Online Library, 2020.
- [29] Marcel CAMPEN : Partitioning surfaces into quadrilateral patches : A survey. In Computer graphics forum, volume 36, pages 567–588. Wiley Online Library, 2017.
- [30] Edwin CATMULL et James CLARK : Recursively generated b-spline surfaces on arbitrary topological meshes. Computer-aided design, 10(6):350–355, 1978.
- [31] CHAOS GROUP : V-RAY round-edges shader. [www.docs.chaosgroup.com](http://www.docs.chaosgroup.com).
- [32] Zhiqin CHEN et Hao ZHANG : Neural marching cubes. ACM Transactions on Graphics (TOG), 40(6):1–15, 2021.
- [33] Qiangqiang CHENG, Pengyu SUN, Chunsheng YANG, Yubin YANG et Peter Xiaoping LIU : A morphing-based 3d point cloud reconstruction framework for medical image processing. Computer methods and programs in biomedicine, 193, 2020.
- [34] Hiroaki CHIYOKURA : An extended rounding operation for modeling solids with free-form surfaces. In Computer Graphics 1987, pages 249–268. Springer, 1987.
- [35] Robert L COOK et Kenneth E. TORRANCE : A reflectance model for computer graphics. ACM Transactions on Graphics (ToG), 1(1):7–24, 1982.

- [36] Simon COURTIN, Mickael RIBARDIÈRE, Sebastien HORNA, Pierre POULIN et Daniel MENEVEAUX : Srec-rt : A structure for ray tracing rounded edges and corners. In Computer Graphics Forum, volume 40, pages 200–214. Wiley Online Library, 2021.
- [37] R. CRUZ, L. LEBRAT, P. BOURGEAT, C. FOOKES, J. FRIPP et O. SALVADO : Deepcsr : A 3d deep learning approach for cortical surface reconstruction. In WACV, pages 755–768, octobre 2020.
- [38] L. CUSTODIO, T. ETIENE, S. PESCO et C. SILVA : Practical considerations on marching cubes 33 topological correctness. Computers and Graphics, 37:840–850, novembre 2013.
- [39] A.M. DALE, B. FISCHL et M.I. SERENO : Cortical surface-based analysis. i. segmentation and surface reconstruction. NeuroImage, 9:179–194, 1999.
- [40] Guillaume DAMIAND et Pascal LIENHARDT : Removal and contraction for n-dimensional generalized maps. In International Conference on Discrete Geometry for Computer Imagery, pages 408–419. Springer, 2003.
- [41] Jonathan DUPUY, Eric HEITZ, Jean-Claude IEHL, Pierre POULIN, Fabrice NEYRET et Victor OSTROMOUKHOV : Linear efficient antialiased displacement and reflectance mapping. ACM Trans. on Graphics, 32(6):Art. 211, 2013.
- [42] John Robert EDMONDS JR : A combinatorial representation for oriented polyhedral surfaces. Thèse de doctorat, 1960.
- [43] A. FEDOROV, R. BEICHEL, J. KALPATHY-CRAMER, J. FINET, J.-C. FILLION-ROBIN, S. PUJOL, C. BAUER, D. JENNINGS, F. FENNESSY, M. SONKA, J. BUATTI, S. AYLWARD, J.V MILLER, S. PIEPER et R. KIKINIS : 3d slicer as an image computing platform for the quantitative imaging network. Magn. Reson. Imaging, 30:1323–1341, novembre 2012.
- [44] José FERNÁNDEZ, Boglárka TÓTH, Lázaro CÁNOVAS et Blas PELEGRÍN : A practical algorithm for decomposing polygonal domains into convex polygons by diagonals. Top, 16(2):367–387, 2008.
- [45] B. FISCHL et A. DALE : Measuring the thickness of the human cerebral cortex from magnetic resonance images. Proceedings of the National Academy of Sciences of the United States of America, 97:11050–11055, octobre 2000.
- [46] Bruce FISCHL, David H SALAT, Evelina BUSA, Marilyn ALBERT, Megan DIETERICH, Christian HASELGROVE, Andre VAN DER KOUWE, Ron KILLIANY, David KENNEDY, Shuna KLAVENESS et al. : Whole brain segmentation : automated labeling of neuroanatomical structures in the human brain. Neuron, 33(3):341–355, 2002.
- [47] Bruce FISCHL, Andr  van der KOUWE, Christophe DESTRIEUX, Eric HALGREN, Florent S GONNE, David H. SALAT, Evelina BUSA, Larry J. SEIDMAN, Jill GOLDSTEIN, David KENNEDY, Verne CAVINESS, Nikos MAKRIS, Bruce ROSEN et Anders M. DALE : Automatically Parcellating the Human Cerebral Cortex. Cerebral Cortex, 14(1):11–22, 01 2004.
- [48] Tim FOLEY et Jeremy SUGERMAN : Kd-tree acceleration structures for a gpu raytracer. In Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware, pages 15–22, 2005.
- [49] Steven FORTUNE : Topological beam tracing. In Proceedings of the fifteenth annual symposium on Computational geometry, pages 59–68, 1999.
- [50] David FRADIN : Modélisation et simulation d’éclairage  $\tilde{A}$  base topologique : application aux environnements architecturaux complexes. Thèse de doctorat, University of Poitiers, France, 2004.
- [51] David FRADIN, Daniel MENEVEAUX et Sebastien HORNA : Out of core photon-mapping for large buildings. In Rendering Techniques, pages 65–72, 2005.
- [52] David FRADIN, Daniel MENEVEAUX et Pascal LIENHARDT : A hierarchical topology-based model for handling complex indoor scenes. In Computer Graphics Forum, volume 25, pages 149–162. Wiley Online Library, 2006.

- [53] Jean FRANÇON et Yves BERTRAND : Topological 3d-manifolds : a statistical study of the cells. Theoretical Computer Science, 234(1-2):233–254, 2000.
- [54] Sarah F FRISKEN : Surfacenets for multi-label segmentations with preservation of sharp boundaries. Journal of Computer Graphics Techniques Vol, 11(1), 2022.
- [55] Henry FUCHS, Zvi M KEDEM et Bruce F NAYLOR : On visible surface generation by a priori tree structures. In Proceedings of the 7th annual conference on Computer graphics and interactive techniques, pages 124–133, 1980.
- [56] Akira FUJIMOTO, Takayuki TANAKA et Kansei IWATA : Arts : Accelerated ray-tracing system. IEEE Computer Graphics and Applications, 6(4):16–26, 1986.
- [57] Kikuo FUJIMURA et Eddy KUO : Shape reconstruction from contours using isotopic deformation. Graphical Models and Image Processing, 61(3):127–147, 1999. <https://doi.org/10.1006/gmp.1999.0494>.
- [58] Eric GALIN et Samir AKKOUICHE : Fast surface reconstruction from contours using implicit surfaces. In Implicit Surfaces, volume 98, pages 139–144, 1998.
- [59] S GANAPATHY et Thomas G DENNEHY : A new general triangulation method for planar contours. ACM Siggraph Computer Graphics, 16(3):69–75, 1982. <https://doi.org/10.1145/965145.801264>.
- [60] Michael P GARRITY : Raytracing irregular volume data. In Proceedings of the 1990 workshop on Volume visualization, pages 35–40, 1990.
- [61] A. S. GLASSNER : Space subdivision for fast ray tracing. IEEE Computer Graphics and Applications, 4(10):15–22, octobre 1984.
- [62] Jeffrey GOLDSMITH et John SALMON : Automatic creation of object hierarchies for ray tracing. IEEE Computer Graphics and Applications, 7(5):14–20, 1987.
- [63] Cindy M GORAL, Kenneth E TORRANCE, Donald P GREENBERG et Bennett BATTAILE : Modeling the interaction of light between diffuse surfaces. ACM SIGGRAPH computer graphics, 18(3):213–222, 1984.
- [64] Marvin J GREENBERG et John R HARPER : Algebraic topology : a first course. CRC Press, 2018.
- [65] John A GREGORY : Smooth interpolation without twist constraints. In Computer aided geometric design, pages 71–87. Elsevier, 1974.
- [66] Emily GROSSNER, Rachel BERNIER, Einat BRENNER, Kathy CHIOU et Frank HILLARY : Pre-frontal gray matter volume predicts metacognitive accuracy following traumatic brain injury. Neuropsychology, 32:484–494, mai 2018.
- [67] Emily GROSSNER, Einat BRENNER, Rachel BERNIER, Amanda RABINOWITZ et Frank HILLARY : Metacognition, aging, and tbi metacognition in individuals aging with traumatic brain injury. (preprint), avril 2021.
- [68] R. GROSSO : Construction of topologically correct and manifold isosurfaces. Computers and Graphics, 35:187–196, août 2016.
- [69] R. GROSSO et D. ZINT : Parallel reconstruction of quad only meshes from volume data. In Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, pages 102–112. SciTePress, 2020.
- [70] A. GUEZIEC, G. TAUBIN, F. LAZARUS et W. HORN : Converting sets of polygons to manifold surfaces by cutting and stitching, pages 383–390. Visualization 1998. IEEE Proceedings, Oct 1998.
- [71] Nicolas GUIARD : Construction de modeles geologique structuraux. Ph.d. thesis, Ecole des Mines, FR, 2006.
- [72] Leonidas GUIBAS et Jorge STOLFI : Primitives for the manipulation of general subdivisions and the computation of voronoi. ACM transactions on graphics (TOG), 4(2):74–123, 1985.

- [73] Leonidas GUIBAS et Jorge STOLFI : Primitives for the manipulation of general subdivisions and the computation of voronoi. ACM transactions on graphics (TOG), 4(2):74–123, 1985.
- [74] Johannes GUNTHER, Stefan POPOV, Hans-Peter SEIDEL et Philipp SLUSALLEK : Realtime ray tracing on gpu with bvh-based packet traversal. In 2007 IEEE Symposium on Interactive Ray Tracing, pages 113–118. IEEE, 2007.
- [75] Peter HACHENBERGER, Lutz KETTNER et Kurt MEHLHORN : Boolean operations on 3d selective nef complexes : Data structure, algorithms, optimized implementation and experiments. Computational Geometry, 38(1):64–99, 2007.
- [76] X. HAN, DL. PHAM, D. TOSUN, ME. RETTMANN, C. XU et JL. PRINCE : Cruise : cortical reconstruction using implicit surface evolution. NeuroImage, 23:997–1012, novembre 2004.
- [77] Si HANG : Tetgen, a delaunay-based quality tetrahedral mesh generator. ACM Trans. Math. Softw., 41(2):11, 2015.
- [78] Pat HANRAHAN : A survey of ray-surface. An introduction to ray tracing, page 79, 1989.
- [79] V. HAVRAN et F. SIXTA : Comparison of hierarchical grids. Ray Tracing News, 12(1), Jun 1999.
- [80] Vlastimil HAVRAN : Heuristic Ray Shooting Algorithms. Ph.D. Thesis, Department of Computer Science and Engineering, Faculty of Electrical Engineering, Czech Technical University in Prague, November 2000.
- [81] Vlastimil HAVRAN et Jiří BITTNER : Ray tracing with sparse boxes. In Proceedings of the 23rd Spring Conference on Computer Graphics, pages 49–54, 2007.
- [82] L. HENSCHER, S. CONJETI, S. ESTRADA, K. DIERS, B. FISCHL et M. REUTER : Fastsurfer - a fast and accurate deep learning based neuroimaging pipeline. NeuroImage, 219:117012, juin 2020.
- [83] Sebastien HORNA, Guillaume DAMIAND, Daniel MENEVEAUX et Yves BERTRAND : Building 3d indoor scenes topology from 2d architectural plans. In GRAPP, pages 37–44, 2007.
- [84] Sébastien HORNA, Daniel MENEVEAUX, Guillaume DAMIAND et Yves BERTRAND : Consistency constraints and 3d building reconstruction. Computer-Aided Design, 41(1):13–27, 2009.
- [85] K.H. HÄHNLE, M. BOMANS, A. POMMERT, M. RIEMER, C. SCHIERS, U. TIEDE et G. WIEBECKE : Voxel-based visualization of medical images in three dimensions. Medical Images : Formation, Handling and Evaluation, 98:145–158, 1992.
- [86] Wenzel JAKOB et Steve MARSCHNER : Manifold exploration : A markov chain monte carlo technique for rendering scenes with difficult specular transport. ACM Transactions on Graphics (TOG), 31(4):1–13, 2012.
- [87] Mark W JONES et Min CHEN : A new approach to the construction of surfaces from contour data. In Computer Graphics Forum, volume 13, pages 75–84. Wiley Online Library, 1994. <https://doi.org/10.1111/1467-8659.1330075>.
- [88] James T. KAJIYA : Ray tracing parametric patches. In ACM SIGGRAPH Computer Graphics, volume 16, pages 245–254, 1982.
- [89] James T KAJIYA : The rendering equation. In Proceedings of the 13th annual conference on Computer graphics and interactive techniques, pages 143–150, 1986.
- [90] Javor KALOJANOV, Markus BILLETER et Philipp SLUSALLEK : Two-level grids for ray tracing on gpus. In Computer Graphics Forum, volume 30, pages 307–314. Wiley Online Library, 2011.
- [91] Timothy L KAY et James T KAJIYA : Ray tracing complex scenes. ACM SIGGRAPH computer graphics, 20(4):269–278, 1986.
- [92] Hassan A KHALIL, Saad DARWISH, Yasmine M IBRAHIM et Osama F HASSAN : 3d-mri brain tumor detection model using modified version of level set segmentation based on dragonfly algorithm. Symmetry, 12(8), 2020.

- [93] Asieh KHOSRAVANI, Mohammad RAHMANIMANESH, Parviz KESHAVARZI et Saeed MOZAFFARI : Fast level set method for glioma brain tumor segmentation based on superpixel fuzzy clustering and lattice boltzmann method. Computer Methods and Programs in Biomedicine, 198, 2021.
- [94] Vassiliki I. KIGKA, George RIGAS, Antonis SAKELLARIOS, Panagiotis SIOGKAS, Ioannis O. ANDRIKOS, Themis P. EXARCHOS, Dimitra LOGGITSI, Constantinos D. ANAGNOSTOPOULOS, Lampros K. MICHALIS, Danilo NEGLIA, Gualtriero PELOSI, Oberdan PARODI et Dimitrios I. FOTIADIS : 3d reconstruction of coronary arteries and atherosclerotic plaques based on computed tomography angiography images. Biomedical Signal Processing and Control, 40:286–294, 02 2018.
- [95] B. KIMIA, A. TANNENBAUM et S. ZUCKER : Shapes, shocks, and deformations i : The components of two-dimensional shape and the reaction-diffusion space. International journal of computer vision, 15(3), 1995.
- [96] Ares LAGAE et Philip DUTRÉ : Accelerating ray tracing using constrained tetrahedralizations. In Computer Graphics Forum, volume 27, pages 1303–1312. Wiley Online Library, 2008.
- [97] Roeel LAZAR, Nadav DYM, Yam KUSHINSKY, Zhiyang HUANG, Tao JU et Yaron LIPMAN : Robust optimization for topological surface reconstruction. ACM Transactions on Graphics (TOG), 37(4):1–10, 2018.
- [98] Leo LEBRAT, Rodrigo SANTA CRUZ, Frederic de GOURNAY, Darren FU, Pierrick BOURGEAT, Jurgen FRIPP, Clinton FOOKES et Olivier SALVADO : Corticalflow : A diffeomorphic mesh deformation module for cortical surface reconstruction. In Advances in Neural Information Processing Systems, volume 34, 12 2021.
- [99] Pierre-François LÉON, Xavier SKAPIN et Philippe MESEURE : Topologically-based animation for describing geological evolution. Journal on Machine Graphics and Vision, 15(3-4):http-www, 2006.
- [100] David LEVIN : Multidimensional reconstruction by set-valued approximations. IMA Journal of Numerical Analysis, 6(2):173–184, 1986. <https://doi.org/10.1093/imanum/6.2.173>.
- [101] Tzu-Mao LI, Jaakko LEHTINEN, Ravi RAMAMOORTHY, Wenzel JAKOB et Frédo DURAND : Anisotropic gaussian mutations for metropolis light transport through hessian-hamiltonian dynamics. ACM Transactions on Graphics (TOG), 34(6):1–13, 2015.
- [102] Yiyi LIAO, Simon DONNE et Andreas GEIGER : Deep marching cubes : Learning explicit surface representations. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2916–2925, 2018.
- [103] Pascal LIENHARDT : Subdivisions of n-dimensional spaces and n-dimensional generalized maps. In Proceedings of the fifth annual symposium on Computational geometry, pages 228–236, 1989.
- [104] Pascal LIENHARDT : Topological models for boundary representation : a comparison with n-dimensional generalized maps. Computer-aided design, 23(1):59–82, 1991.
- [105] Pascal LIENHARDT : N-dimensional generalized combinatorial maps and cellular quasi-manifolds. International Journal of Computational Geometry & Applications, 4(3):275–324, 1994.
- [106] William E LORENSEN et Harvey E CLINE : Marching cubes : A high resolution 3d surface construction algorithm. ACM siggraph computer graphics, 21(4):1987, 163–169, 1987. <http://doi.org/10.1145/37401.37422>.
- [107] Guillaume LOUBET et Fabrice NEYRET : Hybrid mesh-volume LoDs for all-scale pre-filtering of complex 3D assets. Computer Graphics Forum, 36(2):431–442, 2017.
- [108] Shengyu LU : A novel method for 3d reconstruction of blood vessels. Journal of Physics : Conference Series, 1732, 01 2021.

- [109] Qiang MA, Emma C ROBINSON, Bernhard KAINZ, Daniel RUECKERT et Amir ALANSARY : Pialnn : A fast deep learning framework for cortical pial surface reconstruction. In International Workshop on Machine Learning in Clinical Neuroimaging, pages 73–81. Springer, 2021.
- [110] D. MACDONALD, N. KABANI, D. AVIS et A. EVANS : Automated 3-d extraction of inner and outer surfaces of cerebral cortex from mri. NeuroImage, 12:340–356, octobre 2000.
- [111] J David MACDONALD et Kellogg S BOOTH : Heuristics for ray tracing using space subdivision. The Visual Computer, 6(3):153–166, 1990.
- [112] Jeffrey MAHOVSKY et Brian WYVILL : Fast ray-axis aligned bounding box overlap tests with plucker coordinates. Journal of Graphics Tools, 9(1):35–46, 2004.
- [113] Jeffrey MAHOVSKY et Brian WYVILL : Memory-conserving bounding volume hierarchies with coherent raytracing. In Computer Graphics Forum, volume 25, pages 173–182. Wiley Online Library, 2006.
- [114] Martti MÄNTYLÄ : An introduction to solid modeling. Computer Science Press, Inc., 1987.
- [115] D. MARET, N. TELMON, O. PETERS, B. LEPAGE, J. TREIL, J.M. INGLÄNDE, A. PEYRE, J.L. KAHN et M. SIXOU : Effect of voxel size on accuracy of 3d reconstructions with cone beam ct. Dento maxillo facial radiology, 41:649–55, 12 2012.
- [116] Jeffrey MARKER, Ilya BRAUDE, Ken MUSETH et David E BREEN : Contour-based surface reconstruction using implicit curve fitting, and distance field filtering and interpolation. In Volume Graphics, volume 2006, pages 1–9, 2006. <http://dx.doi.org/10.2312/VG/VG06/095-102>.
- [117] Gerd MARMITT et Philipp SLUSALLEK : Fast ray traversal of tetrahedral and hexahedral meshes for direct volume rendering. In Proceedings of the Eighth Joint Eurographics/IEEE VGTC conference on Visualization, pages 235–242, 2006.
- [118] Daniel MENEVEAUX, Kadi BOUATOUCH, Eric MAISEL et Romuald DELMONT : A new partitioning method for architectural environments. The Journal of Visualization and Computer Animation, 9(4):195–213, 1998.
- [119] Gary L. MILLER, Dafna TALMOR, Shang-Hua TENG, Noel WALKINGTON et Han WANG : Control Volume Meshes using Sphere Packing : Generation, Refinement and Coarsening. In Proceedings of the 5th International Meshing Roundtable, IMR '96, pages 47–62, 1996.
- [120] T. M. MURALI et Thomas A. FUNKHOUSER : Consistent solid and boundary representations from arbitrary polygonal data. In Proceedings of the 1997 Symposium on Interactive 3D Graphics, pages 155–163, 1997.
- [121] G. NIELSON : On marching cubes. IEEE Transactions on Visualization and Computer Graphics, 9:283–297, août 2003.
- [122] G. NIELSON : On marching cubes. Dual Marching Cubes, pages 489–496, novembre 2004.
- [123] Hyunwook PARK, Min Jeong KWON et Yeji HAN : Techniques in image segmentation and 3d visualization in brain mri and their applications. Medical Imaging Systems Technology : Methods in cardiovascular and brain systems, 5:2005, 207, 2005. [http://doi.org/10.1142/9789812701046\\_0007](http://doi.org/10.1142/9789812701046_0007).
- [124] Jeong PARK, Peter FLORENCE, Julian STRAUB, Richard NEWCOMBE et Steven LOVEGROVE : DeepSDF : Learning continuous signed distance functions for shape representation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 165–174, 06 2019.
- [125] Steven G PARKER, James BIGLER, Andreas DIETRICH, Heiko FRIEDRICH, Jared HOBEROCK, David LUEBKE, David MCALLISTER, Morgan MCGUIRE, Keith MORLEY, Austin ROBISON et al. : Optix : a general purpose ray tracing engine. Acm transactions on graphics (tog), 29(4):1–13, 2010.
- [126] Michel PERRIN et Jean-François RAINAUD : Shared earth modeling : knowledge driven solutions for building and managing subsurface 3D geological models. Editions Technip, 2013.

- [127] Matt PHARR, Wenzel JAKOB et Greg HUMPHREYS : Physically based rendering : From theory to implementation. Morgan Kaufmann, 2016.
- [128] Les A. PIEGL et Wayne TILLER : Filling n-sided regions with NURBS patches. The Visual Computer, 15(2):77–89, 1999.
- [129] Steve PIEPER, Michael HALLE et Ron KIKINIS : 3d slicer. volume 1, pages 632–635. IEEE International Symposium on Biomedical Imaging : Nano to Macro (ISBI), Arlington, VA, USA, 2004, 2004. <http://doi.org/10.1109/ISBI.2004.1398617>.
- [130] C. PINTER, A. LASSO, A. WANG, D. JAFFRAY et G. FICHTINGER : Slicerrt, radiation therapy research toolkit for 3d slicer. Med. Phys., 39(10):6332–6338, 2012. <https://doi.org/10.1118/1.4754659>.
- [131] Nikos PLATIS et Theoharis THEOHARIS : Fast ray-tetrahedron intersection using plucker coordinates. Journal of graphics tools, 8(4):37–48, 2003.
- [132] Julius PLUCKER : Xvii. on a new geometry of space. Philosophical Transactions of the Royal Society of London, (155):725–791, 1865.
- [133] Stefan POPOV, Johannes GÜNTHER, Hans-Peter SEIDEL et Philipp SLUSALLEK : Stackless kd-tree traversal for high performance gpu ray tracing. In Computer Graphics Forum, volume 26, pages 415–424. Wiley Online Library, 2007.
- [134] Timothy J PURCELL, Ian BUCK, William R MARK et Pat HANRAHAN : Ray tracing on programmable graphics hardware. In ACM SIGGRAPH 2005 Courses, pages 268–es. 2005.
- [135] Boris RAYMOND, Gaël GUENNEBAUD et Pascal BARLA : Multi-scale rendering of scratched materials using a structured SV-BRDF model. ACM Trans. on Graphics, 35(4):Art. 57, 2016.
- [136] William F REINHART : Depth cueing for visual search and cursor positioning. In Stereoscopic Displays and Applications II, volume 1457, pages 221–232. International Society for Optics and Photonics, 1991.
- [137] Alexander RESHETOV, Alexei SOUPIKOV et Jim HURLEY : Multi-level ray tracing algorithm. ACM Transactions on Graphics (TOG), 24(3):1176–1185, 2005.
- [138] Alexander RESHETOV, Alexei SOUPIKOV et William R MARK : Consistent normal interpolation. In ACM SIGGRAPH Asia 2010 papers, pages 1–8. 2010.
- [139] Jaroslaw R ROSSIGNAC et Michael A O’CONNOR : SGC : A dimension-independent model for pointsets with internal structures and incomplete boundaries. IBM TJ Watson Research Center Yorktown Heights, 1989.
- [140] Steven M RUBIN et Turner WHITTED : A 3-dimensional representation for fast rendering of complex scenes. In Proceedings of the 7th annual conference on Computer graphics and interactive techniques, pages 110–116, 1980.
- [141] Alper ŞAHISTAN, Serkan DEMIRCI, Nathan MORRICAL, Stefan ZELLMANN, Aytek AMAN, Ingo WALD et Uğur GÜDÜKBAY : Ray-traced shell traversal of tetrahedral meshes for direct volume visualization. In 2021 IEEE Visualization Conference (VIS), pages 91–95. IEEE, 2021.
- [142] T SAITO, M SHINYA et T TAKAHASHI : Highlighting rounded edges. In New Advances in Computer Graphics, pages 613–629. Springer, 1989.
- [143] W. SCHROEDER, R. MAYNARD et B. GEVECI : Flying edges : A high-performance scalable isocontouring algorithm. pages 33–40. IEEE 5th Symposium on Large Data Analysis and Visualization (LDAV), Chicago, IL UNITED STATES, 2015, 2015. <http://doi.org/10.13140/RG.2.1.3415.9609>.
- [144] Vincent SCHÜSSLER, Eric HEITZ, Johannes HANIKA et Carsten DACHSBACHER : Microfacet-based normal mapping for robust Monte Carlo path tracing. ACM Trans. on Graphics, 36(6):205, 2017.

- [145] Florent SÉGONNE, Anders M DALE, Evelina BUSA, Maureen GLESSNER, David SALAT, Horst K HAHN et Bruce FISCHL : A hybrid approach to the skull stripping problem in mri. Neuroimage, 22(3):1060–1075, 2004.
- [146] Kan-Le SHI, Jun-Hai YONG, Jia-Guang SUN, Jean-Claude PAUL et He-Jin GU : Filling n-sided regions with G1 triangular Coons B-spline patches. The Visual Computer, 2010.
- [147] K. SHOEMAKE : Plücker coordinate tutorial. Ray Tracing News 11, 1998.
- [148] V. SINGH, H. CHERTKOW, J. LERCH, A. EVANS, A. DORR et N. KABANI : Spatial patterns of cortical thinning in mild cognitive impairment and alzheimer’s disease. Brain : a journal of neurology, 129:2885–2893, novembre 2006.
- [149] Vera SKORKOVSKÁ, Ivana KOLINGEROVÁ et Bedrich BENES : A simple and robust approach to computation of meshes intersection. In Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, pages 175–182. INSTICC, SciTePress, 2018.
- [150] S.M. SMITH : Fast robust automated brain extraction. Human Brain Mapping, 17:143–155, novembre 2002.
- [151] Jos STAM et Charles LOOP : Quad/triangle subdivision. Computer Graphics Forum, 22(1):79–85, 2003.
- [152] Kyle SUNDERLAND, Boyeong WOO, Csaba PINTER et Gabor FICHTINGER : Reconstruction of surfaces from planar contours through contour interpolation. volume 9415, pages 435–442. International Society for Optics and Photonics, Orlando, FLORIDA, UNITED STATES, 2015, 2015. <http://doi.org/10.1117/12.2081436>.
- [153] Márta SZILVASI-NAGY : Flexible rounding operation for polyhedra. Computer-Aided Design, 23(9):629–633, 1991.
- [154] Toshimitsu TANAKA et Tokiichiro TAKAHASHI : Precise rendering method for exact anti-aliasing and highlighting. The Visual Computer, 8(5):315–326, 1992.
- [155] Seth TELLER, Celeste FOWLER, Thomas FUNKHOUSER et Pat HANRAHAN : Partitioning and ordering large radiosity computations. In Proceedings of the 21st annual conference on Computer graphics and interactive techniques, pages 443–450, 1994.
- [156] Seth J. TELLER et Michael E. HOHMEYER : Stabbing oriented convex polygons in randomized  $O(n^2)$  time. 1992.
- [157] Seth J TELLER et Carlo H SÉQUIN : Visibility preprocessing for interactive walkthroughs. ACM SIGGRAPH Computer Graphics, 25(4):61–70, 1991.
- [158] Daniel L. TOTH : On ray tracing parametric surfaces. In ACM SIGGRAPH Computer Graphics, volume 19, pages 171–179, 1985.
- [159] F. VIDIL et G. DAMIAND : Moka. <http://moka-modeller.sourceforge.net/>, 2003.
- [160] Ingo WALD : Realtime ray tracing and interactive global illumination. Ausgezeichnete Informatikdissertationen 2004, 2005.
- [161] Ingo WALD et Vlastimil HAVRAN : On building fast kd-trees for ray tracing, and on doing that in  $O(n \log n)$ . In 2006 IEEE Symposium on Interactive Ray Tracing, pages 61–69. IEEE, 2006.
- [162] Ingo WALD, Thiago IZE, Andrew KENSLER, Aaron KNOLL et Steven G PARKER : Ray tracing animated scenes using coherent grid traversal. In ACM SIGGRAPH 2006 Papers, pages 485–493. 2006.
- [163] Ingo WALD, Philipp SLUSALLEK, Carsten BENTHIN et Markus WAGNER : Interactive rendering with coherent ray tracing. In Computer graphics forum, volume 20, pages 153–165. Wiley Online Library, 2001.
- [164] Ling-Yu WEI, Kan-Le SHI et Jun-Hai YONG : Rendering chamfering structures of sharp edges. The Visual Computer, 31(11):1511–1519, 2015.

- [165] Kevin WEILER : Boundary graph operators for non-manifold geometric modeling topology representations. Geometric modeling for CAD applications, pages 37–66, 1988.
- [166] Kevin WEILER : The radial-edge structure : A topological representation for non-manifold geometric boundary representations. Geometric modeling for CAD applications, 1988.
- [167] Turner WHITTED : An improved illumination model for shaded display. In Proceedings of the 6th annual conference on Computer graphics and interactive techniques, page 14, 1979.
- [168] Sven WOOP, Louis FENG, Ingo WALD et Carsten BENTHIN : Embree ray tracing kernels for cpus and the xeon phi architecture. In ACM SIGGRAPH 2013 Talks, SIGGRAPH '13, pages 44 :1–44 :1, New York, NY, USA, 2013. ACM.
- [169] Lifan WU, Shuang ZHAO, Ling-Qi YAN et Ravi RAMAMOORTHY : Accurate appearance preserving prefiltering for rendering displacement-mapped surfaces. ACM Trans. on Graphics, 38(4), 2019.
- [170] C. Xu X. HAN et J. PRINCE : A topology preserving level set method for geometric deformable models. IEEE Trans. Pattern Anal. MachineIntell, 25:755–768, juin 2003.
- [171] M. Brady Y. ZHANG et S.M. SMITH : Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. IEEE Trans Med Imag, 20:45–57, janvier 2001.
- [172] Y.L. YAMINA, M. LILA, T. ZOUAGUI et R. JENNANE : A topology constrained geometric deformable model for medical image segmentation. Biomedical Signal Processing and Control, 64:997–1012, octobre 2020.
- [173] Dong-Jin YOO : Three-dimensional surface reconstruction of human bone using a b-spline based interpolation approach. Computer-Aided Design, 43(8):934–947, 2011. <https://doi.org/10.1016/j.cad.2011.03.002>.
- [174] Qingnan ZHOU, Eitan GRINSPUN, Denis ZORIN et Alec JACOBSON : Mesh arrangements for solid geometry. ACM Transactions on Graphics (TOG), 35(4):1–15, 2016.



# Structuration topologique pour le rendu et la visualisation 3D

Présenté par :

**Sébastien Horna**

**Résumé :** Les travaux présentés dans cette habilitation à diriger des recherches visent à mettre à contribution les structures topologiques et hiérarchiques pour améliorer les opérations de reconstruction 3D et optimiser les algorithmes de simulation d'éclairage par lancé de rayons.

Les structures de données utilisées en modélisation géométrique à base topologique étant plus riches que les structures habituellement employés dans d'autres type d'application, nous proposons dans un premier axe de recherche de mettre à contribution les relations de voisinage issues de l'étape de modélisation pour améliorer la rapidité des calculs en synthèse d'images. A la manière des méthodes utilisant une structure accélératrice basée sur une partition de l'espace contraint, nous nous appuyons sur notre modèle topologique dédié à l'architecture afin de calculer le parcours d'une onde électromagnétique (un rayon) à travers une scène 3D. L'application résultante permet une visualisation interactive de scènes d'intérieur de bâtiment composé de plusieurs centaines de millions de polygones.

Dans le contexte du rendu réaliste des bords arrondis des objets, nous proposons ensuite dans un deuxième axe de recherche une méthode non invasive (c'est-à-dire qui ne modifie pas la géométrie d'origine) pour la modélisation et le rendu de chanfrein. Contrairement aux surfaces maillées, notre objectif est de garantir une apparence arrondie et lisse, quels que soient le point de vue, la distance de l'observateur et l'éclairage incident. Au coeur de notre contribution se trouve une structure hiérarchique et topologique qui définit automatiquement et avec précision la géométrie des zones arrondies. Cette structure, nommée SREC-RT, est construite automatiquement et est intégrée dans une structure d'accélération dédiée algorithmes de lancé de rayons. Elle permet de déterminer la région d'intérêt de chaque chanfrein et d'y appliquer le traitement correspondant, cela sans augmenter la complexité géométrique de la scène 3D. L'intersection des rayons avec les surfaces arrondies est accélérée grâce à un graphe topologique défini par les relations de voisinage entre les arêtes et les sommets.

Enfin, dans un troisième axe de recherche, nous proposons un système pour la reconstruction de maillage volumétrique à partir d'acquisitions médicales 3D discrètes telles que des IRM. Ce processus permet la reconstruction automatique d'objets composés de plusieurs substances, afin d'assurer une visualisation globale, unique et cohérente d'un cortex cérébral. Notre processus donne, sans aucune intervention de l'utilisateur, un modèle 3D topologique robuste et propre avec des relations d'adjacence explicites entre les segments reconstruits. Le modèle résultant respecte les données sémantiques, sans les défauts topologiques et géométriques habituellement rencontrés dans les autres approches de reconstruction 3D. Il est ainsi directement utilisable pour des calculs d'estimation des données métabolites/physiologiques au sein des différentes parties du cerveau dans le cadre d'outils d'aide à la détection des tumeurs.

---

**Discipline :** Informatique Graphique.

---

**Mots-clés :** Synthèse d'image, Modélisation géométrique à base topologiques, Reconstruction 3D, Rendu réaliste, Lancé de rayons.

---